International Journal of Software Engineering and Knowledge Engineering

Effective Social Productivity Measurements During Software Development: An Empirical Study

Murat Yilmaz
Department of Computer Engineering, Çankaya University, Ankara, Turkey
e-mail: myilmaz@cankaya.edu.tr

Rory O'Connor
School of Computing, Dublin City University, Dublin, Ireland
e-mail: roconnor@computing.dcu.ie

Paul Clarke
Regulated Software Research Center, Dundalk Institute of Technology, Dundalk, Ireland
e-mail: paul.clarke@dkit.ie

Received (Day Month Year)
Revised (Day Month Year)
Accepted (Day Month Year)

Much of contemporary scientific discussion regarding factors that influence software development productivity is undertaken in various domains where there is an insufficient empirical basis for exploring socio-technical factors of productivity that are specific to a software development organization. The purpose of the study is to characterize the multidimensional nature of software development productivity and its social aspects as a set of latent constructs (i.e. variables that are not directly observed) for a medium-sized software company. To this end, we designed an exploratory in-depth field study based on the hypothesized productivity constructs, which were modeled by a set of factors identified from literature reviews, and later refined by industrial focus groups. In order to demonstrate the applicability of our approach, we conducted confirmatory factor analysis with the data attained from a questionnaire with 216 participants.

To investigate factors of influence further, we analyzed the impact of selected team-based variables over the latent constructs of productivity. Taken together, our findings confirm that such an approach can be used to explore the quantifiable influence of socio-technical factors that would affect productivity of a particular software development organization. Ultimately, the resulting model provides guidance to explore the comparative importance of a set of firm-specific factors that may help to improve the productivity of the organization.

Keywords: Structure Equation Modeling, Productivity Factors of Software Development, Socio-Technical Aspects of Software Development.

1. Introduction

One of the main considerations of software development organization is to manage the social aspects of software development by investigating the relationship between social and technical factors that are encountered in the software development lifecycle [1]. Recently, researchers have shown an increased interest in organizational social structures particularly in a software development process where it was suggested that such structures should be tailored regarding the actual development problem [2]. Indeed, over the last decade a significant amount of software engineering researchers have considered software development as a *social activity* and have conducted research on the implications of socio-technical factors for the software development process [3]. Additionally, it is important to recognize that the productivity of software development teams is also dependent on the factors that are affecting software development [4], and their implications on software development roles and team related variables [5]. Being able to understand such quantifiable factors from the planned software activities not only accomplishes project goals but also is an axiomatic step towards improving the productivity of software development organizations as a whole [6].

Productivity is one of the most important concepts that governs the economic production. The past thirty years have seen increasingly rapid advances in the field of software engineering, which shows that software development productivity is likely to have not only from the technical but also from a social dimension [3]. Therefore, recent evidence suggests that software productivity has a multivariate structure, which should be measured from multiple perspectives. In fact, it can be considered as a *construct*[a], which should be investigated from several disciplines, as conducted in productivity research in behavioral and social sciences [7]. Ultimately, such an approach identifies a source of competitive advantage for a software organization, which leads to industrial success.

As a consequence of a lack of understanding of the socio-economic factors of software quality, there exist a variety of definitions of productivity in software engineering literature, which is an issue that inhibits a thorough understanding or measurement of software development productivity [8]. In fact, the present exploratory field study contends that there is a need for techniques to deal with the factors that hinder the productivity of software development. Therefore, it is the primary goal of this research that understanding the socio-technical problems of software development organization requires exploring the relationships among several productivity factors and their associations as identified from the literature.

The fundamental assumption here is that software development productivity is a latent construct, which can be explained by a set of certain factors. In addition, we presume that productivity is a composite variable, which is composed of independent and correlated attributes. Consequently, proposed measurement model is formative [9], which specifies the relationship between productivity and factors where the direction of the causal flow is from factors to the construct. Secondly, software productivity is considered as a multi-dimensional construct where its social aspects are formalized as social productivity and social capital. Based on such

[a]A conceptual variable that cannot be either observed or measured directly.

an assumption, a factor-based productivity model can be constructed using both latent constructs and relevant factors identified from the literature review. The goal is to propose a suitable way to explore how well these indicators are related and to explain or measure the intended constructs by theorizing relationships between actual data and hypothetical variables particularly for a software development organization.

This research proposes an empirical approach to investigating the relationships among the hypothetical latent constructs based on the factors[b] that are affecting software development productivity - a technique that can be used to evaluate the conceptual propositions with respect to the accuracy of data collected. First, we hypothesize the relationships between several social factors (e.g. social debt, communication, team cohesion) and economic factors (e.g. software size, management quality, process) identified in the literature potentially affecting the productivity of software development. Based on the identified factors, we build a tripartite structure equation model, and as a secondary objective we evaluated the model with data collected from a software development organization. In the second part of this study, we analyze the impact of software roles and team-based variables on the latent factors affecting software development productivity using the collected data.

The objectives of the research are as follows:

> **Objective 1:** Investigate the relationships among several productivity factors and their associations with the latent constructs (i.e. productivity, social productivity and social capital) as identified in the literature through a confirmatory factor analysis model.
>
> **Objective 2:** Explore the impact of team-based variables and software development roles on productivity, social productivity, and social capital.

The remainder of this paper is organized as follows: In section two, we define software productivity, and survey the underlying factors affecting the social aspects of software development productivity. Next, we introduce the notion of social productivity and social capital for a software development organization. Then, we briefly survey some important topics of structural equation modeling (SEM). The third and the fourth sections identify the research hypotheses and approach adopted in this research, respectively. The methodological approach taken in this study is a mixed methodology based on a rigorous process tailored to conduct this work. Built on a systematic process, the fifth and the following sections, represent the results, and discuss a tripartite SEM model from our field study based on the views of 216 personnel that participated to our survey. Based on the factors surveyed from the literature and the survey data collected from a middle-size software company, the relationships between software productivity, social productivity and social cap-

[b]To measure and present a latent construct a set of observable indicators are captured.

ital are empirically investigated. Therefore, this study makes a major contribution to research on software productivity by exploring connections between the social aspects of productivity and aims to measure the factors using advance statistical techniques. Next, we analyze the impact of teams and roles to previously identified latent constructs. Therefore, it is considered as a systematic analysis the social aspect of software development productivity where we found interesting results regarding to social capital of software development. Finally, the paper concludes with research implications and a summary of results and further discussions.

2. Background of the Research

Despite the fact that the factors that affect the productivity of software development has been researched from both academic and industrial viewpoints, software development organizations still cannot adequately measure the impact of factors that are affecting the software development activities [10, 11]. One question that needs to be asked here is whether there is a way to explain software development productivity in terms of the factors affecting it. To this end, first we hypothesized a productivity model in terms of the factors that are systematically identified [10]. Secondly, to understand social factors from the software practitioners' perspective, we enhanced our model with the factors based on the social aspects that are affecting productivity by a series of industrial focus group studies. Thirdly, we focused on the literature of social capital, where we particularly integrated a social capital model (based on Narayan and Cassidy's [12] work) to a software development organization. After identifying the indicators[c] to build a tripartite model of productivity specific for a software development organization, our ultimate goal was to validate such a model by using the data collected from an industrial setting.

2.1. *Productivity*

Although productivity can be considered as the amount of production over a limited period of time, software development productivity is quite a challenge to measure [13]. From a traditional perspective, software development productivity is equal to size of the software output to the cost needed for production. However, the broad use of the term productivity is sometimes measured from different viewpoints such as the skill set of software practitioners, their techniques and the instruments they used in the software development processes [14]. From an industrial point of view, productivity is generally understood as a key issue for software development organizations when creating a competitive advantage. Trendowicz and Münch [15] suggest that the factors affecting productivity of software development should be selected to measure the software development productivity (based on the importance of their role), which may alternate in different domains of software development. In addi-

[c]To measure and present a latent construct a set of observable indicators were captured.

tion, they claimed that a productivity model should only include the factors, which are found as the most important ones by the literature.

Factors of Productivity

One of the first systematic studies of the software development productivity issues was reported by Scacchi [16], who reviewed the entire software development productivity literature while analyzing potential productivity problems. Most importantly, he suggested that a multivariate analysis for identifying the factors affecting the software development process might be essential. In addition, Nagappan et al [17] investigated how software quality in Microsoft Windows Vista development is affected from organizational structure where they utilized a set of organizational measures and methods to improve the productivity of software development. Other than the traditional metrics of software quality, they confirmed that organizational factors can be considered as the key predictors of software productivity.

A large and growing body of literature has investigated the factors affecting the productivity of software development organizations. By following the software development productivity literature that was summarized by a number of systematic reviews [10, 18, 11], we select the factors that potentially affect the productivity of software development. These are (i) the software development process [19, 20, 21, 8], (ii) the level of individual's motivation [19, 22, 8] and its influence on software engineers [23, 24, 25], (iii) the ability of an organization to stabilize the customer requirements [26, 27], (iv) software project management quality [1, 28], (v) team issues such as aligning skills of the software teams [19, 20, 21], (vi) reuse [19, 22, 29, 30, 31], (vii) tools that are used in software development [32, 33], (viii) the effect of programming language on software development productivity [19, 22, 34], (ix) software size [35, 18], (x) team size [32, 33, 35], and finally (xi) software complexity [21, 19, 22, 17, 8].

2.2. *Social Capital*

In the field of social sciences, a social network is an organized form of people that comprises the individuals and the connections among them. In general, individuals are considered to be connected in a fabric of social network, and expect some benefits from the social formations and the way the network operates [36]. Consequently, social capital may be broadly defined as an intangible resource accumulated by the social connections. Therefore, individuals should have to be linked to one other to improve their social capital.

> *"**Social capital** is a **potential** form of intangible resources based on patterns of social connections and social abilities of individuals, teams or social groups that has a potential to contribute to the economic progress of an organization."* [37]

Based on its qualitative attributes, social capital is a network of elements consisting of nodes and links of connection. Hence, this form of capital can be increased by improving social interactions. At a social level, it is not surprising to discover that *social capital* can be transformed to measure the productivity of a team or an individual of a software development organization. In light of this information, it should be easier to create compatible and productive team formations. The value of social capital is mostly hidden in a network of interactions or connections where it could be observable in the social activities of a software development organization. Hence, building and improving both professional and individual social bonds would likely to enhance the productivity with the notion of *boundaryless* development landscape [38].

Coleman [39] suggests that all kinds of social configurations may create some amount of social capital. However, to gain a benefit from their existing social capital, its relationship with social productivity should be investigated.

2.3. *Social Productivity*

Barnett [40] describes *social productivity* as an outcome, which can be provided from a social activity. As previously mentioned, software development is considered as a social activity where people should be working in close proximity. Therefore, the notion of social productivity should be measured by the factors that refer to social aspects of productivity. To understand the impacts of social issues over a software organization, we investigate the level of importance for several social factors such as trust, communication, social life, and information awareness. We hypothesized that social productivity should be materialized by several social factors where its relationship with the social capital should also be investigated. Here, we define the social productivity of software development as follows;

> ***"Social productivity*** *is an intangible asset [latent construct] as we have termed here to measure the effects of social factors on the socioeconomic landscape of a software organization. "* [37]

From a software development organizations perspective, social productivity is an attempt to explain the social factors that are hindering the software development productivity. Therefore, we select several potential factors affecting the social productivity from the literature and build our hypothetical model (see Figure 4) based on these; (i) Stober and Hansmann [41] for reputation of a team leader on conflicts and his or her skills, (ii) Dittrich et al. [3] for identification of communication issues with respect to level of individuals interactions, (iii) Koh and Maguire [42] for awareness of information in turbulent business landscapes, (iv) Tamburri et al [43] for social debt, which was based on the unpredictable or a hidden cost that can be caused by a flaw in social relations, which can be connected with the notion of trust in a software development organization [2], and identification of trust (i.e.

level of loyalty) in the software teams [44, 45] (v) Kelly [46] for socialization or social life in the work environments, (vi) Hazzan and Dubinsky [45] for fairness, e.g. fair allocation of work, and finally Churchville [47] for frequent meetings i.e. how team members are informed about each others progress.

2.4. *Structural Equation Modeling*

A family of flexible interrelated statistical techniques (i.e. multivariate, multiple regression analysis, factor analysis) frequently used in social science studies to analyze empirical data and test variables and evaluate their network of hypothesized relationships is called structural equation modeling (SEM) [48]. Based on the patterns of statistical expectation, it is a confirmatory multivariate analysis technique used to estimate the structural or casual relationship between two variable types (i.e. observed and latent). SEM models use a collection of simultaneous equations based on a combination of observed and latent variables (hypothetical constructs or factors), which are frequently used by sociology, psychology research and econometric research [49]. The main component of a structural equation model is an initial hypothesis, which also includes the components that may be connected that are assessed by several statistical tests and if necessary adjusted through modification indexes.

SEM allows the researcher to explore the multivariate relationships that can be used to test an actual hypothesis, which may theoretically be justifiable by empirical observations. A typical SEM model usually encompasses the graphical depiction of the correlation patterns based on a set of variables, and is frequently used for validation of the relationships among the latent constructs. Although it is a quantitative approach, SEM offers a start from a qualitative viewpoint; it has the ability to show how the chosen factors or variables are not only correlated but also interrelated to one other. Therefore, it can be helpful for observing the relationship among several coefficients. It enables the researcher to assess the effectiveness of a hypothetical model for the sampled data. In particular, a model based on the combination of regression, path, and confirmatory factor analysis should be useful for analyzing social factors and their interdependencies.

In addition, it is sometimes used as an instrument to form a measurement scale. A typical SEM includes the direct and the indirect associations of variables that are statistically assessed to identify a relationship between data and the proposed or hypothetical model. Consequently, the notion of correlation and covariance is important for a SEM analysis because they signify the pairs of relationships for a group of variables [50]. Correlation is a tool that defines the discovered linear relationship between two variables (coefficient of correlation measured in a range of -1 to +1). A positive value indicates that there is a positive correlation among the variables, where negative values state the opposite [9]. In fact, SEM is considered as a set of equations used to compute a multiple linear regression model where several factors are calculated with respect to observed weights [51]. A SEM model can be used for measuring the correlations and covariation among the latent constructs, *where the regression model is designated in the structural part of the model, and factor analysis model is designated in the measurement model* [52] . There are four main steps in a typical SEM analysis; (i) model development (building a conceptual framework), (ii) path diagram construction (building a representation of associations), (iii) assessment of measurement model, (iv) assessment of structural model [53].

A SEM can be specified in several formats such as path diagrams. However, these figures usually follow *de facto* standards. A typical SEM model represents how the researcher relates the hypothetical constructs and the collected data based on observed variables (illustrated in rectangles). These variables are derived from a set of questionnaire in a survey tool. To represent these items, a limited number of graphics are used such as ellipse, which signifies the latent constructs that are estimated from the observed variables, single headed arrows, which represent predictive relationships and a double headed curved arrow between two latent variables, which indicates that they are correlated.

Based on the variance-covariance matrix, a good-fitting model designates that a theoretical or hypothetical construct is consistent with the empirical dataset. Such a model is useful for examining the relationships of the causal paths of a SEM model, which can improve the original form [54]. However, sometimes a model that seems like a good-fitting model may not be a working model. Therefore, it is important to use several model validation techniques to evaluate the validity of a SEM model to obtain more conclusive results. A chi-square test, the comparative fit index (CFI), the goodness of fit index (GFI), the adjusted goodness of fit index (AGFI), and the root mean square error of approximation (RMSEA) are the most common fit-indices used in SEM investigations [49]. In addition, sample size is another parameter that affects the validity of a model [52, 49], where a number of researchers suggest that constructing a model with no latent variable is somehow more suitable for a limited sample size.

One of the earliest current fit-indices in SEM research is the chi-square test statistics. It is frequently used for testing the model fit by investigating whether a null hypothesis is true or false. Barrett [55] argues that a chi-square test is enough for investigating the model fit. Although for a large sample of data this test usually shows statistically significant results, it is still used as a measure of general model fit to identify whether a theoretical model differs from the sample variance-covariance matrices calculated from the data [51]. It is affected by the highness of the correlations, which results in poor fit for the proposed model. Moreover, the evidence collected from simulation studies confirms the sensibility of chi-square test in terms of the size of the sample set [56].

The root mean square error of approximation (RMSEA) index is probably the best-known index for model fitting. Analogous to other fit indices, RMSEA uses a complexity parameter depending upon the degrees of freedom of a model [54]. According to Browne and Cudeck [57], RMSEA value measured below .05 indicates a good model fit between the observed data and theoretical model, while values below .08 is considered as a reasonable fit [53].

Based on the parameters identified above, we select a set of indices to evaluate the models constructed in this study, namely chi-square goodness-of-fit test, ratio of chi-square to degrees of freedom, root mean squared error of approximation (RMSEA), and two other kind of measures known as goodness-of-fit index (GFI), and adjusted goodness-of-fit index (AGFI). Table 1 presents the descriptions of and thresholds for several indices based on the works of Bagozzi and Yi [58], Cote et al. [59], and Ping [60], etc.

Table 1. Descriptions and Cut-offs for the Fit Indexes

Fit index	*Descriptions*	*Cut-offs*
χ^2	*Displays the disagreement between hypothetical model and collected data*	$p < .05$
χ^2/df	*As chi-square test is depended on the size of a sample*	2-1 or 3-1
RMSEA	*Displays the level of fitness of a model*	<.05 good <.08 reasonable
GFI	*A de facto measure of the descriptive adequacy of a model*	0 no-fit, 1 perfect-fit
AGFI	*GFI adapted for degrees of freedom*	0 no-fit, 1 perfect-fit
NNFI	*Displays the level of improvement compared to null model*	0 no-fit, 1 perfect-fit
CFI	*Shows betterness of a model fit with respect to a null model*	0 no-fit, 1 perfect-fit

Lastly, so as to apply SEM properly, the hypothesized measurement model should be illustrated by a diagram in which measured (observed) variables are called factors or indicators. In a SEM model, observed variables are represented in the form of rectangles where latent (unobserved) variables are shown by a circle and the relationships between these variables are usually shown by arrows. To achieve a precise measurement result, the indicators that are used to measure the latent constructs should be validated by using methods such as literature reviews, and expert reviews.

3. Research Design of the Field Study

In this section, we elaborate the research process for the field study. We describe the details of the research process used for the empirical investigation of the factors affecting software development productivity by an exploratory field study using a middle-sized software company. In particular, we consider productivity, social productivity and social capital as latent variables that cannot be directly observed.

Therefore, we seek several potential factors to measure them.

The details of our research methodology (see Figure 1) comprises the following steps:

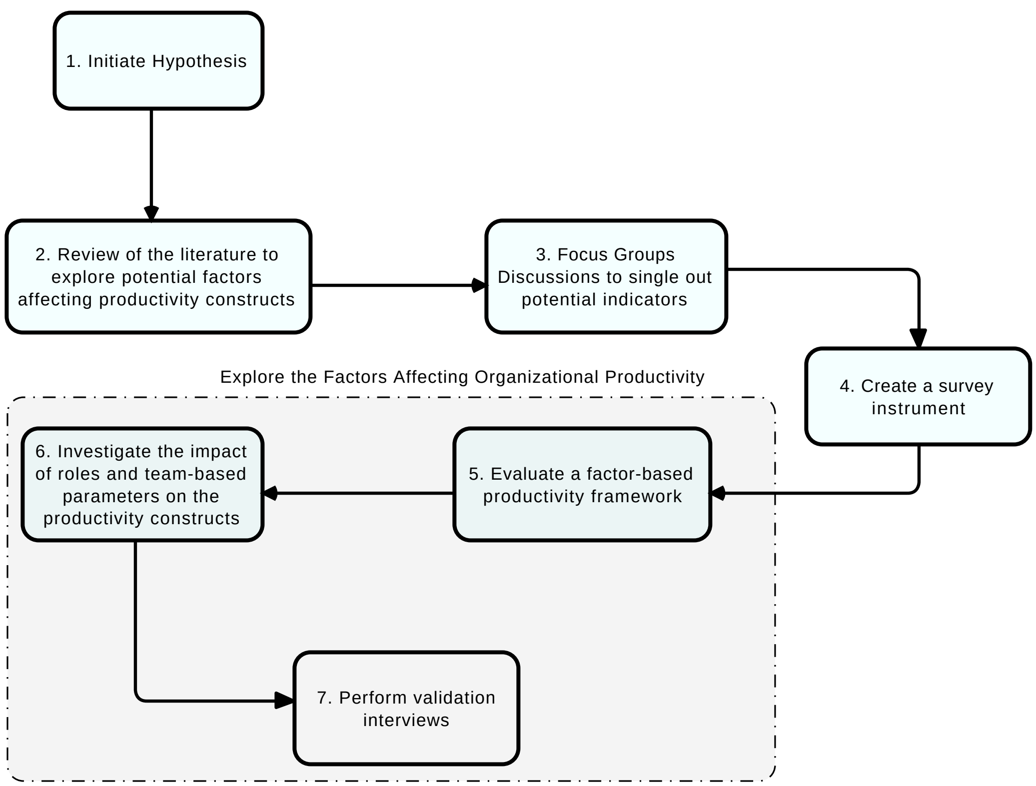

Fig. 1. The Systematic Approach for Exploring the Factors of Productivity

(1) We form our hypothesis, which represents the fact that there is an observable relationship among the pairs of productivity, social productivity and social capital based on the selected indicators.
(2) To explore the potential indicators of hypothesized constructs, we review the literature to consolidate the key factors affecting the productivity, social productivity and social capital of a software development organization.
(3) To select the indicators for identifying the latent constructs, we conduct a series of industrial focus group studies where we single out indicators for creating three structure models (hypothetical models, as shown in Figure 2, Figure 3), and Figure 4) having three latent variables that are identified.

(4) To investigate the degree of participant's agreement on indicators affecting productivity, we developed a survey instrument with 41 questions on a Likert scale between 1 (strongly disagree) and 5 (strongly agree). We also asked 4 questions exploring such dimensions as work experience of a participant, gender, actual and the ideal team size.
(5) Fifthly, we evaluate a factor-based productivity framework by an exploratory field study using a confirmatory factor analysis approach, which allows us to assess the hypothesized relationship patterns between latent constructs and observed variables. The dataset from questionnaire is analyzed through the LISREL tool and then the tripartite unified SEM model (see Figure 5) is generated. After that, the consistency between the hypothetical model and the SEM model is checked by applying the good-fit indexes (see Table 1).
(6) Next, we investigate the impact of roles and team-based parameters on the latent constructs of productivity, social productivity and social capital.
(7) Finally, we perform a set of validation interviews to discuss the results obtained from the tripartite SEM model with the management team of our industrial partner, which yields some interesting insights and interpretations.

Table 2 shows the summary of the research activities, the inputs, outputs, and the research methods that were used during each step of the proposed research design.

Table 2. Decsription of each research activity with input, output, and used technique.

Activity	Input	Output	Technique
1.Hypothesis Generation	Experience and Knowledge	Hypothesis	Observation
2.Background Formation	Potential Indicators	Potential Key Factors	Literature Review
3.Indicator Selection	Literature Review Results	Selected Indicators	Focus Group
4.Measuring Agreement	Selected Indicators	Survey Results	Survey
5.Analyzing Results	Survey Results	Factor Structure	Factor Analysis
6.Investigating Roles and Teams	Survey Results	Correlations	Statistical analysis
7.Validating Results	All findings	Evaluation	Validation Interviews

3.1. *Research Hypotheses*

To understand the implications of the hypothesized factors that are potentially affecting software development productivity, a SEM model has been proposed with emphasis on social and economic indicators from the literature of software development research. Thus, we claim that exploring the relationship between productivity and its social dimensions that are measured by socio-technical factors, will provide a way to investigate the software development productivity. In addition, we envision to explore the impact of software development roles (i.e. job titles) and a set of team-based variables (e.g. team size, years they spent in the industry) upon hypothesized constructs.

Here, we present a list of research questions, which guide the research.

Research Question 1: *Are we able to capture the software development productivity by using a set of indicators and latent constructs (e.g. social capital and social productivity) that are potentially affecting productivity?*

Research Question 2: *Is there a positive correlation between productivity, social productivity and social capital for software development?*

To date, as it is qualitative in its nature, software productivity as a notion has been found difficult to measure [8]. Ultimately, the goal of the first research question is to measure the relationship between latent variables that we propose and the observable variables for each latent construct found in the literature. The second research question seeks a correlation between three latent constructs: productivity, social productivity, and social capital based on the identified indicators. An implication of the second research question is the possibility to explore the factors that most contribute to variance in the productivity based on the hypothesized latent constructs (i.e. social productivity and social capital).

In light of these two research questions, this study is concerned with investigating the relationship between productivity and its hypothesized constructs, and hence the first hypothesis has been developed to support this endeavor.

Hypothesis 1: *There is a positive correlation between the selected factors affecting software development and the productivity of software development.*

Despite the fact that tailoring software development roles is a common practice in industrial software development [5], there has been little discussion about the perspective differences among roles of software practitioners regarding software development and team productivity. Therefore, the second set of research questions intend to uncover the ways to explore team productivity using identified variables such as the actual and ideal size of a team for better productivity, the years a practitioner spent in a company, years of experience in industry, etc. Furthermore, we seek out a relationship between these variables, our latent constructs and the roles that practitioners are assigned during the course of the development activities.

Research Question 3: *How roles affect the perception of the relationships between roles of software practitioners and the software team productivity?*

Research Question 4: *Is there any empirical relationship between social capital and identified variables to measure the variations in software team productivity?*

The second hypothesis relies on the argument that our hypothetical constructs such as social capital are related with the identified variables that are potentially affecting the productivity of software development. To seek answers to these questions, we have established our second hypothesis:

Hypothesis 2: *There is a relationship among the perceived team productivity, roles and our hypothetical (latent) constructs of software productivity.*

3.2. *Underlying Models*

Based on the factors reviewed from a set of systematic literature reviews, we illustrate Figure 2, which shows the conceptual model for the factors affecting the productivity of software development organizations. Starting with this figure, variables observed by the literature are represented in the form of rectangles where latent (unobserved) variables are shown by a circle and the relationships between these variables are illustrated by arrows. Here, productivity is considered as a composite variable where the causal action flows from the observed variables through a latent construct. Detailed information on the preliminary development of the model, and a sizable majority of the studies surveyed here can be found in [37].

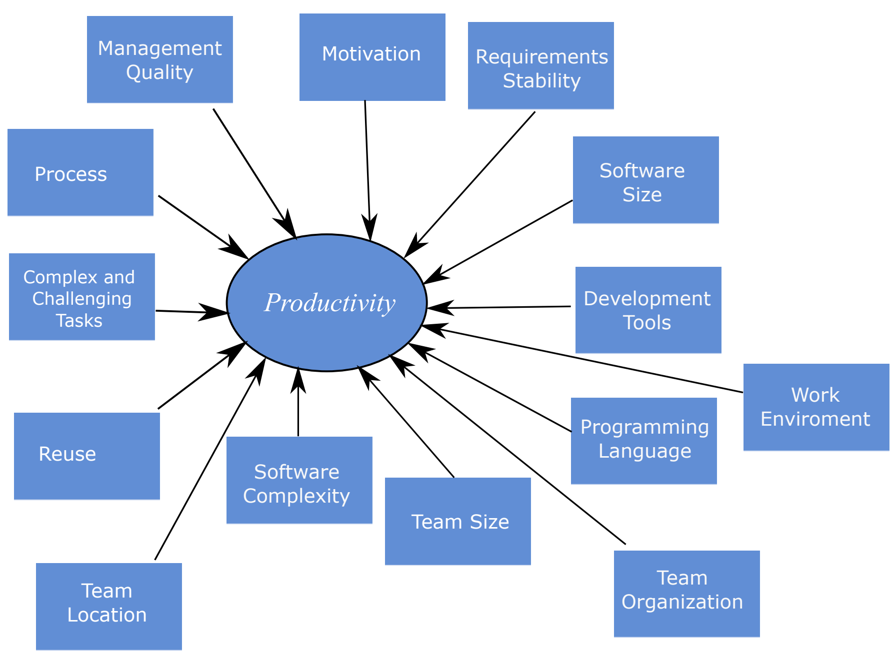

Fig. 2. A productivity Model Based on Factors Affecting Software Development. The ellipse denotes latent constructs, rectangles shows the observed variables that are potentially affecting these constructs where single headed arrows illustrate the relationships between an observed variable and a latent construct.

As inspired from the work of Narayan and Cassidy [12], we selected a set of seven factors that were hypothesized to measure the social capital. Using these seven indicators of social capital (as shown in Figure 3), we build a social capital questionnaire to measure the factors namely neighborhood connections, group characteristics, generalized norms, togetherness, everyday sociability, volunteerism, trust. The details regarding to these indicators can be found in [12].

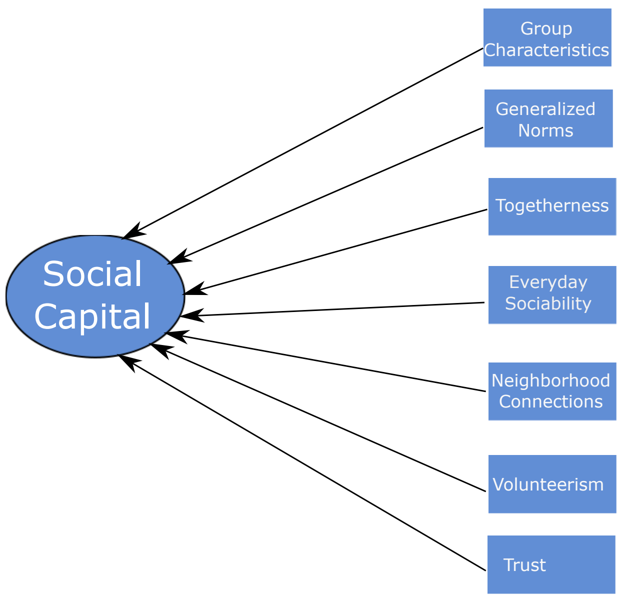

Fig. 3. A Hypothesized Model of Social Capital. The ellipse denotes latent constructs, rectangles shows the observed variables that are potentially affecting these constructs where single headed arrows illustrate the relationships between an observed variable and a latent construct.

Figure 4 illustrates a social productivity model we propose, based on the hypothetical factors affecting social productivity of software development.

4. Data Collection

In order to conduct this part of the research, we selected a middle-sized software development organization, *Simurg*[d]. The first reason was that they were willing to participate in the research. Secondly, the company employs more than two hundred software practitioners. Therefore, the size was adequate, likely to increase the reliability of outputs. Although Simurg consist of individuals who are highly motivated with diverse experience levels in software development, management team reported

[d]To protect the identity of the firm, we use a fictitious name.

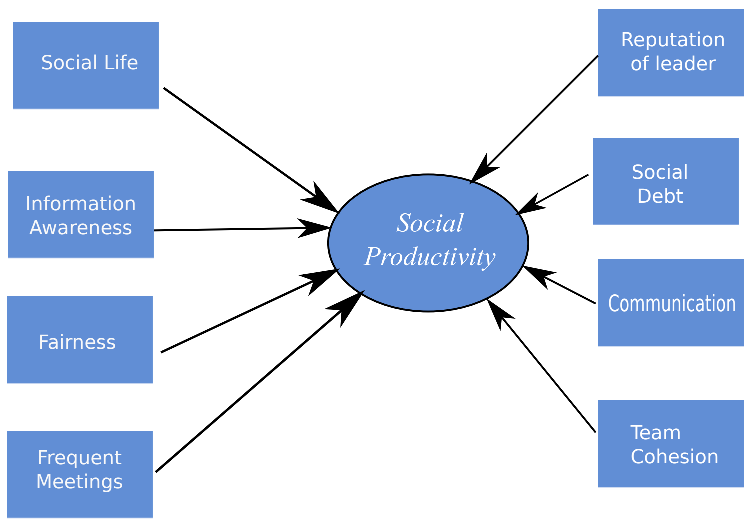

Fig. 4. A Social Productivity Model Based on Factors Affecting Software Development. The ellipse denotes latent constructs, rectangles shows the observed variables that are potentially affecting these constructs where single headed arrows illustrate the relationships between an observed variable and a latent construct.

that there were several short-term productivity fluctuations in software teams that need to be investigated. After introducing our novel approach to capture the factors of productivity specifically to Simurg, the management group of Simurg was very interested in the potential factors affecting their organizational productivity, and therefore they were willing to contribute to the study. Ultimately, the entire study took nearly 10 months to complete in the industrial setting (at Simurg) in Turkey.

Additionally, Simurg is a software development organization in which a combination of agile and traditional software development methodologies is used. The software is developed in a contract-based approach with incremental steps where all of their clients expects detailed documentation. Although Simurg has a functional hierarchical structure, the company is highly organized regarding projects like a projectized organization where most of their work depends on the projects undertaken. In such an organizational structure, several projects need to be managed concurrently, and therefore adjusting projects versus dedicated resources is considered as quite challenging. The company has several development activities occur in multiple locations, and the number of software teams vary between four to forty members. This would increase our chances of for observing different team configurations especially required for the second part of this study.

To evaluate our hypothetical model with the empirical data, we developed a survey instrument based on two different resources: (i) literature review of the factors of productivity, social productivity, and social capital affecting a software development organization, (ii) focus group study conducted with the management team. Furthermore, we examine a set of the documentation and a series of case reports previously prepared about organizational productivity.

4.1. *Industrial Focus Group*

Focus group is a form of group interview (i.e. researcher-led group discussion [61]) conducted to capture a content in the research process where participants are asked about their opinions, understandings, stories or perceptions as regards a previously selected subject [62].

After reviewing the potential factors affecting both productivity and social productivity, a focus group study was conducted to obtain a broad range of opinions on productivity factors from the software company. The discussion group was composed of ten personnel of the entire management team[e]. Table 3 outlines the profile of the 10 participants including their roles (titles), age, years of experience and level of education.

Table 3. Participants' Information

Participant ID	Title	Age	Years of Experience	Education
P1	IT Specialist	33	6	MSc.
P2	Project Manager	47	7	PhD.
P3	Software Architect	37	12	BSc.
P4	Software Developer	31	6	BSc.
P5	Software Developer	33	7	BSc.
P6	R&D Team Lead	39	14	PhD..
P7	Software Tester	32	4	MA.
P8	System Analysis	34	9	BA.
P9	R&D Team Member	32	7	MSc.
P10	R&D Team Member	31	5	MSc.

The researcher stimulates the group conversation by posing a set of questions regarding to the core topic of interest so that individuals can freely discuss about a subject. In the opening statement researcher introduced himself, pronounce the importance of the subject matter, highlight the value of each member's contribution, and explain the purpose of the group meeting.

To avoid bias in factor selection, we first asked the question: *What is your opinion of the factors that are affecting the software development productivity?* During the first phase of the discussion, we asked each participant to prepare a (secret)

[e]The focus group participants were mainly from the management team of the software company who were not participated to the survey study.

list of factors on a blank sheet of paper that they think affecting both social and the technical aspects of productivity. The researcher asked each participants: *What social and technical factors do you think are affecting the productivity?, How would you describe the social and technical factors of productivity?* In addition, they were asked to update their list by adding any factors that came to mind during the discussion process.

To let everyone participate, as a second phase, each participant was asked to share their list by reading loudly while others checked theirs for duplicates. Each factor numbered and recorded on a while board with the original wordings of the participants. After gathering preliminary the factors from participants, the researcher revealed the factors found from the literature and asked focus group to discuss the findings. The researcher would then asked each participants: *Do you record any of these factors found in the literature in your work sheet?, Which one do you think is more important among these ones for the software development productivity?* To clarify the indicated items, he then asked the participants about their understanding of each factor.

To sum up, the focus group activity was useful to refine potential factors found in the literature for generating a mixed viewpoint using both the literature and the industry. As a result, the second part of the discussion started with the factors that the authors found significance, and it proceeded by investigating the most important factors from participants' viewpoint.

The focus group activity provided us with an opportunity to discuss the factors found in the literature with an industrial perspective. Consequently, while constructing our SEM model, we used the experience gained during this session.

4.2. *Survey Instrument*

The questionnaire had questions about the potential factors from literature of software development productivity using a 5-point Likert scale grading between strongly agree (5) and strongly disagree (1) for productivity. Additionally, the survey had several questions like gender, years of work experience of the participants in this company as well as the ideal team size and the actual size of their software team. The survey questions were developed to measure the relationships between the observable factors and latent constructs of productivity. To ensure proper interpretation of each question, we worked with several experts from Simurg's management team. In light of these efforts, a set of survey questions were constructed, and refined. Ultimately, the management team of Simurg announced the finalized version of the survey at their internal web-portal. As the arrangements were kindly requested by the management, all individuals participated. The survey was conducted in Turkey.

To increase accessibility, the questionnaire was prepared with the LimeSurvey. It is an open source web-based tool for conducting surveys, which was employed as the primary instrument of data collection. The survey ran approximately for one month, which has 45 questions and also has an introductory letter and confidentiality statement. To increase the understandability of questions, we built five question categories that were presented in saveable web-based sessions. The five categorical parts represent five different aspects that we were investigating. It was also used as a stopover for participants and to store their answers if need be. Although nearly all members of the company are bilingual, our survey was available in both Turkish and English with the following parts: (i) 17 questions about the factors affecting software productivity elicited from the literature such as motivation, management quality (e.g. process, development tools, programming languages), complexity issues (e.g. task, process, product), work environment, re-usability, requirements stability, team issues (e.g. size, organization, location); (ii) 12 questions about the social productivity factors identified from the literature such as conflicts and reputation of a team leader, social interaction, social life, information awareness, team cohesion, fairness, frequent meetings, and social debt; (iii) 10 questions about the factors of social capital surveyed in the literature such as neighborhood connections, group characteristics (personality types), generalized norms, togetherness, everyday sociability, volunteerism, and trust; (iv) 6 general questions regarding participants' age, gender, years of experience, and job title.

5. Results

To test our hypothetical model of productivity and to reveal the relationships of the latent factors and the factors that are observable by the literature, we perform a confirmatory factor analysis by using the linear and continuous framework of LISREL [63], which is one of the most popular computer tool for SEM analysis especially for theory testing. By distinguishing the latent and the observed variables, LISREL allows researcher to estimate a set of parameters based on simultaneous equation, which is useful for exploring complex interaction structures. Based on maximum-likelihood estimation techniques, it uses the standard SEM notation to model the covariance structure of observed (i.e. manifest) variables, and to estimate several parameters (such as loadings, paths, etc.) by exploring an overall model fit as a whole.

There are several reasons for employing LISREL both as a statistical model and as well as a computer program. First, it combines the measurement considerations as both latent constructs and observed variables into one structural model, which enables us to build a broad range of models, based on variance/covariance matrices. Secondly, it brings a conceptual clarity that highlights the relationships among variables, i.e. what indicators are expected and that are observed. Thirdly, a LISREL model is superior to other tools for measuring model testability where it enables a researcher to evaluate the probable model predictions more efficiently.

Of the initial cohort of 213 industrial participants who returned the questionnaire, 21 were excluded as their questionnaires had missing pairs. We ended up with 192 *appropriate* observations (cases) 24% of which were female, and 76% of which were male participants. Prior to data analysis, the Turkish translation of the survey was checked for both consistency and the language by a number of experts from industry and academia. Next, we analyzed the role distribution of the sample from the company Simurg. Table 4 shows the initial results.

Table 4. Distribution of Roles of the Participants in Development Organization

Role	**Number of Individuals**	**Percentage (%) in Organization**
IT Specialist	25	13
Project Manager	17	9
Software Architect	4	2
Software Developer	66	35
Team Leader	13	7
Software Tester	23	12
Software Specialist	29	15
System Analyst	10	5
System Engineer	5	2
Total	192	100

To asses the *internal consistency* of the survey, we use Cronbach's α [64], a frequently used variable to measure the reliability of responses collected by psychometric instruments [65]. The values around .70 or higher are reliable, where a high Cronbach's α value signifies that there are highly correlated variables that are found in the survey [66].

$$\alpha_{Cronbach} = \frac{N}{N-1}\left(\frac{S^2 - \sum S_i^2}{S^2}\right) \tag{1}$$

where N is the number of items in the questionnaire, S^2 is the *variance of total score* for each participants, $\sum S_i^2$ is the summation of variances for each question. Depending upon what is evaluated, the number of respondents or the number of questions is shown by j and the variance is calculated as follows:

$$S^2 = \frac{1}{j-1}\left(\sum_{i=1}^{j}(X_i - \bar{X})^2\right) \tag{2}$$

Table 5 below illustrates the Cronbach's α values for our survey instrument. Overall, it was apparent from our calculations that the responses to our survey had high Cronbach's α values, which provides estimate about the proportion of variability of the questionnaire. In addition, we checked the consistency of each set of questions for the constructs of the survey. The important result to emerge from these calculations was that our survey instrument had an adequate consistency according to Cronbach's α values calculated for each of the selected constructs.

Table 6 presents responses for all identified factors, their descriptions, standard deviations, and the variances as descriptive statistics calculated for each factor that potentially affects the productivity of software development.

Table 5. Individual Sections of the Questionnaire with respect to Reliability Coefficients

	Survey Constructs		
	Productivity	Social Productivity	Social Capital
Cronbach's α values	.68	.73	.76

In response to the survey instrument, most of the questions indicated that nearly all the factors proposed to affect software development productivity measures had a rating higher than 3, which was considered as the middle point in a 5-point Likert scale. This ensures our survey questions are relevant to participants.

Table 6. Means, Variances and Standard Deviations of the Factors of Productivity

Factor ID	Descriptions	Mean	s.d.	Variance
X1	Level of individuals motivation	4.72	0.49	0.24
X2	Level of interests of individuals for their assigned tasks	4.56	0.66	0.44
X3	Development process or methodology	3.94	0.77	0.59
X4	Programming language	3.77	0.90	0.81
X5	The tools and technologies used	4.29	0.67	0.45
X6	Complex and challenging tasks	3.97	0.95	0.89
X7	Large and complex structured projects	3.37	0.89	0.80
X8	Tasks and their complex connections	3.80	0.74	0.55
X9	The work environment	4.17	0.74	0.55
X10	Using an off-the-shelf product	3.80	0.97	0.93
X11	The ability of an organization to stabilize requirements	4.22	0.78	0.61
X12	The changes in requirements of a project	3.68	1.04	1.09
X13	The team size	3.64	1.02	1.05
X14	Verbal communication of team members	4.39	0.70	0.49
X15	Non-verbal communications	3.03	1.10	1.22
X16	Teams in different locations	2.33	1.10	1.21
X17	Internal problem solving skills of a team	4.19	0.76	0.58
X20	Team Leaders conflict resolution skills	3.74	0.91	0.82
X21	Team leaders general skills	4.42	0.59	0.35
X22	Communication with all team members	4.30	0.80	0.64
X23	Social life out of the work place	3.65	0.89	0.79
X24	Knowing the tasks of others	4.22	0.85	0.73
X25	Collective team memory	4.07	0.61	0.37
X26	The unity in the service of team goals	4.16	0.69	0.47
X27	Enjoying teammates company	3.81	1.04	1.09
X28	Working less than the others	3.50	1.17	1.36
X29	Fair allocation of work	4.08	0.73	0.54
X30	Frequent Meetings	3.96	0.95	0.90
X31	Social Debt	4.29	0.67	0.45
X32	Social connections	3.64	0.97	0.93
X33	Efficient usage of the social connections	3.68	0.95	0.90
X34	Social connections and career success	3.39	1.00	0.99
X35	Variation of personalities	3.42	0.96	0.92
X36	Generalized norms	3.21	0.92	0.85
X37	Togetherness	2.21	1.05	1.10
X38	Everyday sociability	3.27	1.13	1.27
X39	Extra potion of work for more social connections	3.14	1.03	1.05
X40	Volunteerism	3.66	0.88	0.78
X41	Trust	3.84	0.81	0.66

After the survey was closed, we conducted a series of interviews to understand the problematic items in the questionnaire. The question, *teams in different locations*, was not interpreted properly. Later we found that the term *location* was understood differently, e.g. in the same office or otherwise in the same country, etc. Similarly, our interviews revealed that *non-verbal communication* might also not interpreted as expected because it had different meanings for the participants. In general, therefore, the two problematic items were found and excluded from the structural model.

5.1. *The Tripartite Measurement Model*

The content of this section is concerned with estimating the relationships in the surveyed productivity factors with respect to each other. Structural modeling is therefore used for linking a theoretical perspective (i.e. proposed model) with the observed data. To investigate the expected pattern of relationship between the hypothesized latent constructs based on the predicted potential factors, we propose a tripartite unified model for explaining the association between hypothesized latent constructs. We built a model based on all parameters mentioned in the survey data for the components of the tripartite model namely; productivity, social productivity and social capital.

To obtain observed variables, which were suitable for the analysis, we combined related survey questions into categories by arranging them regarding their commonalities to reflect broader themes. In fact, there were excessive numbers of variables (41 items) for a limited sample size (N=216), which may lead improper conclusions [67]. Consequently, Bentler and Chou [68] suggest that a moderately large sample size should be considered as a requirement of structural equation modeling in order to obtain statistically significant (stable and trustworthy) parameter estimates. Item parceling (i.e. combination of a set of conceptually similar items to improve the precision of results) is used to reduce the number of parameters as well as the model complexity [69]. Ultimately, the summary score on each clustered item serves as an observed variable, which optimizes measurement structure, and improves reliability [70]. In this manner, we obtained seven observed variables for each latent constructs suitable for SEM analysis.

The newly formed predictors are summarized in Table 7. We calculated the average scores for these new themes in an attempt to improve our accuracy in measuring the latent constructs. For example, the average scores were calculated for factors of productivity related to team issues $X13$, $X14$, $X17$, all of which were combined to form a new category $Y7$. Concurrently, complexity issues were formed by the average scores from the factors classified as $X6$ through $X8$ to form $Y3$.

Table 7 presents all factors that were transformed to Y with calculated means, variances, and standard deviations as descriptive statistics. Once again, we calculated the reliability coefficients called the Cronbach's α for updated survey. It was found as .8, which confirmed that the internal consistency of data items for a factor-based model.

Finally, we constructed a tripartite model (Figure 5) to investigate the relationships between productivity, social productivity, and social capital, and to show

Table 7. Means, Variances and Standard Deviations of the Combined Factors

Factor ID	**Factor Name**	**New Factor ID**	**mean**	**s.d.**	**var.**
X1 - X2	Motivation	Y1	4.64	0.48	0.23
X3 - X4 - X5	Management Quality	Y2	4.00	0.59	0.35
X6 - X7- X8	Complexity Issues	Y3	3.71	0.61	0.37
X9	Work Environment	Y4	4.17	0.74	0.55
X10	Re-usability	Y5	3.80	0.97	0.93
X11 - X12	Requirements Stability	Y6	3.95	0.67	0.45
X13 - X14 - X17	Team Issues	Y7	3.52	0.48	0.23
X20 - X21	Team Leader	Y8	4.08	0.62	0.38
X22 - X23	Social Interaction and com.	Y9	3.98	0.67	0.45
X24 - X25	Information Awareness	Y10	4.14	0.60	0.36
X26 - X27	Team Cohesion	Y11	3.98	0.69	0.48
X28 - X29	Fairness	Y12	3.79	0.71	0.51
X30	Frequent Meetings	Y13	3.96	0.95	0.90
X31	Social Debt	Y14	4.29	0.67	0.45
X32 - X33	Neighborhood Connections	Y15	3.66	0.81	0.65
X34 - X35	Group Characteristics	Y16	3.40	0.80	0.65
X36	Generalized Norms	Y17	3.21	0.92	0.85
X37	Togetherness	Y18	2.21	1.05	1.10
X38	Everyday sociability	Y19	3.27	1.13	1.27
X39 - X40	Volunteerism	Y20	3.40	0.85	0.72
X41	Experience and Trust	Y21	3.84	0.81	0.66

the factors affecting these latent constructs. To measure the hypothesized influence between the observed and latent variables, we built a model with three constructs, all of which were found statistically significant ($p < .05$) and ranged between .30 and .73. The independence model was clearly rejectable where the χ^2 for independence model with 210 degrees of freedom is 1680.137. The proposed model yielded a good-fit, where $\chi^2(186, N = 192) = 296.896, p < .001$, and the fit indices for the tripartite model were $RMSEA = .0559, GFI = .90, AGFI = .84, CFI = .914, NNFI = .90$). Furthermore, a χ^2 difference test was conducted, $\Delta\chi^2(24, N = 192) = 1383.241, p < .001$). Management quality (*Standardized Path Coefficient*= .59) was a significant predictor for productivity, which was followed by motivation (*Standardized Path Coefficient*= .53) and work environment (*Standardized Path Coefficient*= .47).

The most significant predictor for social productivity was found to be information awareness (*Standardized Path Coefficient*= .65), which was followed by the predictors of social debt (*Standardized Path Coefficient*= .60), and fairness (*Standardized Path Coefficient*= .56). The most significant predictor for social capital was group characteristics (*Standardized Path Coefficient*= .73), which was followed by neighborhood connections (*Standardized Path Coefficient*= .68). In addition, all of the structural correlations among the latent variables were statistically significant. The correlation between productivity and social productivity was .70; social productivity and social capital was .55, and productivity and social capital was .48.

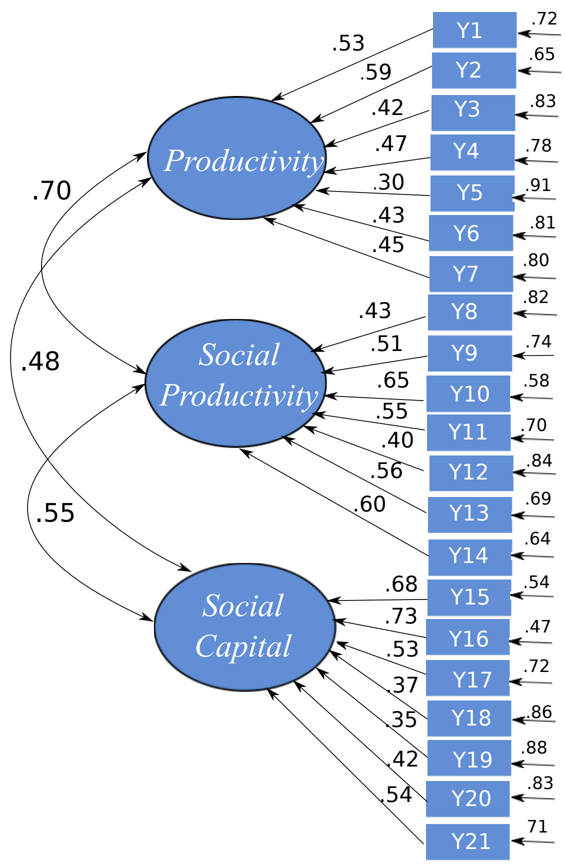

Fig. 5. Model IV for Productivity, Social Productivity and Social Capital in a Software Development Organization. The ellipse denotes latent constructs, rectangles shows the observed variables that are potentially affecting these constructs where single headed arrows illustrate the relationships between an observed variable and a latent construct and the factor loading values. The arrows entering the rectangles show the values of unexplained variances due to measurement errors. Two-headed arcs with values show the correlations between latent variables.

5.2. *The Impact of Teams and Roles to Productivity, Social Productivity, and Social Capital*

The purpose of this part of the study was to explore the association between the latent constructs and the identified job roles. In particular, it sought to examine the importance of these constructs with respect to roles and team-based parameters based on the questions asked during the survey.

In this part of the analysis, we categorized the latent constructs with respect to the identified job roles in Simurg. For each role identified by our survey, we calculated means, variances, and standard deviations, that is descriptive statistics, presented in the Table 8. To test the homogeneity of the data, here we calculated three coefficients of variation (CV), which is the percentage ratio (a comparison) of standard deviation to mean (see Equation 3). The data is called homogeneous when CV is below 33%, while values above cut-off value signify that there are outliers or some unwanted measurement errors that can affect the outputs. Since our coefficients fell within the threshold value, we confirmed that the data was homogeneous.

$$CV = \frac{s}{\bar{X}}(100) \tag{3}$$

Table 8. Roles versus the Means, Standard Deviations, and Coefficient of Variations

Roles	Productivity	Social Productivity	Social Capital
IT Specialist	3.82	4.00	3.40
Project manager	3.85	4.01	3.23
Software architect	3.93	4.10	2.85
Software developer	3.88	3.97	3.29
Team Leader	3.89	4.04	3.26
Software Tester	3.96	4.29	3.60
Software specialist	3.96	4.00	3.39
System Analyst	3.70	3.82	3.20
System Engineer	3.64	3.82	3.70
Mean($\bar{X}$)	3.85	4.01	3.32
Standard deviation(s)	0.11	0.14	0.25
Coefficient of variation (%)	2.92	3.58	7.40

To investigate the difference between the ideal and the actual team size for Simurg, we asked two questions in our survey. Question 18; *How many members are in your immediate development team* ($TEAMSIZE$), and Question 19; *In your view, how many of your team members are operating at high levels of productivity*, ($IDEAL\_TEAM\_SIZE$). Using this information, we derived three variables namely, $EXCESS\_TEAM\_SIZE$, which was identified by actual team size minus ideal team size.

$WHETHER\_IDEAL\_TEAM$, a boolean variable, which can be true or false (zero or one), and finally a variable called $UNDER\_IDEAL\_OVER$. In addition, we asked the participants about their years at the industry ($WYEAR$), and the years they spend in this company ($WTHISFIRM$). The descriptive statistics with the averages of constructs defined for the team-based variables with respect to the role of individuals were presented at Table 9.

This table is highly revealing in several ways. Firstly, it shows the average of years of experience both in this organization and outside, and as a whole, different roles identified by the survey. Secondly, it is apparent from this table that software architects have the highest experience average, and system analysts work in the biggest teams. By comparing with other roles, software specialists and IT specialists

Table 9. Mean Scores of Roles versus Team Constructs

Roles	WYEAR	WTHISFIRM	TEAMSIZE	IDEAL TEAMSIZE	EXCESS TEAMSIZE
IT Specialist	7.14	1.52	5.36	3.84	1.52
Project Manager	12.18	3.06	5.94	3.71	2.24
Software Architect	17.25	4.50	10.00	6.00	4.00
Software developer	5.67	3.14	7.44	4.80	2.64
Team Leader	10.77	2.85	8.38	4.85	3.54
Software Tester	3.85	2.00	7.30	5.04	2.26
Software Specialist	1.88	1.41	8.28	7.24	1.03
System Analyst	8.20	4.40	14.80	9.10	5.70
System Engineer	13.40	2.20	8.40	5.40	3.00
Mean	8.93	2.79	8.43	5.55	2.88
Standard deviation	4.91	1.13	2.76	1.71	1.40

on the other hand think that they are working close to the ideal team size. From this data, we can see that the lowest value for years of experience both in this firm and in general are found as software specialists.

Furthermore, the results indicate that, of the 192 participants who completed this part of the questionnaire, 80 participants (22 female, 58 male) thought that they were in a team that is in the ideal size, while 112 participants (25 female, 87 male) believe that their actual team is not at the ideal size. The role of participant with respect to their belief in under, ideal, and over-sized teams are shown in the Table 10.

Table 10. Roles versus Participants Thoughts on Team Size

Role	Under-sized	Ideal Team size	Over-sized
IT Specialist	0	13	12
Project Manager	0	5	12
Software Architect	0	1	3
Software developer	2	21	43
Team Leader	0	5	8
Software Tester	1	10	12
Software Specialist	0	19	10
System Analyst	0	4	6
System Engineer	0	2	3
Total Personnel	3 (%.02)	80 (%.42.98)	109 (%57)

From the data in Table 10, it is apparent that 57.02% of survey participants thought that their team was not in ideal size. What is interesting in this data was that many of the software developers think that they were in an over sized team. However, there were only two software developers and one software tester who thought that they might need additional members to their teams to reach the ideal team size.

Turning now to the experimental evidence based on our survey, we seek the degree of causal (strength of) relationships between different variable pairs. One way to investigate the linear relationships between a pair of variables is to construct a correlation structure. To understand how the data trends together, the relationship can be quantified by a coefficient called correlation coefficient. It is a coefficient that measures how strongly the variables are connected, and what values they take between -1.0 and $+1.0$. The minus sign shows the changes in the negative direction (i.e. inverse relationship), so when the correlation is $+1.0$, it is called a perfect positive correlation. Using a set of n observation of a pair of variables, $(X_1, Y_1), (X_2, Y_2), ...(X_n, Y_n)$, the correlation coefficient for this part of the study was calculated by following equation 4.

$$r = \frac{\sum_{i=1}^{n} (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^{n} (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^{n} (Y_i - \bar{Y})^2}} \tag{4}$$

To calculate the significance of correlation (r), a T-test can be performed, and calculated by equation 5. The test comprise of a comparison of a cataloged t value with respect to an empirical one.

$$|t| = \left| \frac{r}{\sqrt{1 - r^2}} \times \sqrt{n-2} \right| \tag{5}$$

where n is the number of roles, and r is the correlations (n=9 in our case), n-2 is the number of degrees of freedom. Table 11 provides the significant correlations and values of T-tests, which were used to analyze the relationship between pairs of variables.

Table 11. Statistically Significant Pairwise Correlations for Roles from the Survey

Variable Pairs	r	t^a
WTHISFIRM - TEAMSIZE	.68	2.45
WTHISFIRM - Social Capital	-.76	-3.05
WTHISFIRM - EXCESS_TEAM_SIZE	.86	4.49
IDEAL_TEAMSIZE - TEAMSIZE	.91	5.75
TEAMSIZE - EXCESS_TEAM_SIZE	.86	4.48

[1] $t_{critical}$ (df=7, 0,05)=2,3651, $p < .05$

Table 11 illustrates that the years participants spent at the company and the team size had a positive correlation, .68, and the correlation between the years they spent at the company and excess team size was .86, whereas a strong negative correlation, $-.76$, was observed between the years participants spent at the company and the value that participants gave to social capital. We confirm that the longer period participants worked in the company, the bigger teams they started to work with where they tended to think that their team was too large to obtain higher productivity.

Interestingly, participants who spent more time with Simurg were inclined to give less importance to social capital (see Table 11). Furthermore, the correlation between team size-ideal team size was positive and higher than the correlation between team size-excess team size. We conclude that for those participants who work in a larger team size, their ideal team size gets higher, and they also tend to think that their team size was not ideal for software development productivity.

6. Discussion

Prior studies have noted the importance of measuring software productivity for software development organizations. Several earlier studies have shown that software productivity can be modelled in terms of the software size [71], lines of code [72], function points [73], which favours the user perspective for assessing the functionality of a product, function points per hour [27], and measurement of effort [74].

There are only a few studies in the software engineering literature concerning the quantification of factors affecting the productivity of software development especially by using a sophisticated method like structural modelling. There are, for example, a SEM model for application development productivity [75], and a SEM model of feasibility evaluation and project success [76]. However, no previous studies investigate the software productivity using a complex factor structure based on several aspects of software productivity. In addition, no earlier study published research from an empirical perspective reveal the relationship between productivity and social aspects of software development from multi-dimensional viewpoint.

The present study was designed as an empirical assessment for evaluating the latent constructs that are expected to be conceptually and empirically related. Based on our earlier findings [37, 77], the values of the indicators calculated by the data collected from students slightly different from the industrial version of the study. There are several reasons for this: First, we have a limited number of indicators that were identified and used in the classroom-tested version. Second, we refined the preliminary version of the questionnaire, which was later discussed with experts from academia and industry for several adjustments. Third, previous models were only tested using the data collected from students; many students have limited understanding of the notions that they have not yet practically experienced.

Kitchenham et al. [78] suggest that using students as subjects instead of practitioners does not create a major problem. A potential weakness with this argument, however, is that particularly for quantitative and complex surveys, students might not have enough experience to evaluate the survey questions like the participants from the industry. Contrary to Kitchenham's argument, particularly for SEM modeling, by comparing the two different settings, authors can confirm that there were significant differences in the variance of data collected from an industrial settings (e.g. this study) with respect to a classroom environment (e.g. [77]).

In this study, we empirically evaluate the hypothesized relationships between the latent constructs and several factors that are affecting them. The results confirm that productivity is highly associated with social productivity, and moderately associated with social capital. There is, however, a moderate correlation observed between social productivity and social capital. In particular, these empirical findings strongly support the notion that social factors dramatically influence software productivity. Returning to the social and organizational issues posed at the beginning of this study, it is now possible to state that most of the factors selected from the literature are affecting the productivity of a software development organization.

In general, the current findings add substantially to our understanding of the economic and social factors of productivity, which can be quantified using a structural approach. In addition, this exploratory field study is currently a comprehensive (empirical) research that holds a significant value for industry and academia especially in that it develops a multifactor productivity analysis. The multi-dimensional factor structure of the tripartite SEM model includes seven variables for measuring three of our constructs. To the authors' knowledge, this is also the first study of this nature to assess the implications of roles, team size and social capital on software development.

6.1. *Threats to Validity*

Here, we consider several potential threats that were addressed for the validity of the exploratory field study. To deal with construct validity issues, first we conducted a comprehensive literature review to build the theoretical model; secondly we asked a group of experts from both academia and industry to assess our initial constructs and the potential factors that are the representatives of the constructs being measured. It was suggested to conduct an initial implementation in an industrial focus group in order to check the validity of our research questions and conduct a test study with our preliminary ideas. Then, we published the initial results of a pilot study [77] and got some early feedback before conducting the exploratory field study. In brief, our initial research questions were taken from a purely theoretical perspective and aligned with practical industrial viewpoint. In addition, we revised our survey questions based on the initial comments from experts to increase the clarity of items. After conducting the survey, for the first part of the study, we used Cronbach's α values to asses the reliability of our survey. For the second part, we checked the data homogeneity by using the coefficients of variation. In accordance with these, we believe that both latent and observed variables, and pairwise correlations in this study have been measured properly.

To cope with internal validity problems, we built the structural model based on the selected constructs and tested them with a set of factors identified from the literature and refined through focus groups. Secondly, we selected the most convenient time for participants to start the survey (we had to wait for a while to capture such a time frame), and limited the time for respondents to two weeks time to avoid any history effect. Thirdly, the measures taken through the survey were collected consistently (i.e. without changing the dependent variables in the survey instrument) so as to deal with any instrumentation effect.

To manage external validity problems, we conducted validation interviews in which the measured factors were reviewed by a group of experts who had already contributed to the several aspects of the study; (i) to check the validity of the identified factors and their importance, (ii) to check whether the findings are generalizable. Finally for the reliability aspect, we clarified the data collection method and documented the processes and the protocols that were developed.

6.2. *Validation Interviews*

To understand how well we measure the productivity scale, one of the issues that emerge from our model is a need to evaluate them by a series of model validation interviews [79] with individuals from the management team of Simurg.

We validated the tripartite model with the company by asking participants questions about the factors in the model and their opinion about the validity of measured factors using questions such as "*What do you think about the company-based results we have found with SEM model?*", "*Do you think that any factor is missing or misrepresented in the productivity model? If so, which ones?*", "*Does your organization benefit from this new productivity perspective?*", "*Do you think these results may help the software development organization to improve their productivity?*"

As the management team discussed a series of simplified version of the structural model in a previous focus group study [77], they were delighted to examine the outcomes in this part of the work. The interviewees were encouraged to comment on the relationships between the predictors, and latent constructs. Although some of the interviewees suggested some minor alterations about sorting the priority of factors, most of the participants had found these results consistent with respect to their expectations. The overall results of our structural model help the management team to discuss about the social factors, quantified latent constructs, and most importantly methods to improve their organizational productivity by using their implications.

Lastly, the results of this study indicate that software development organizations should be able to use our technique for investigating their organizational specific factors of productivity. An implication of this is the possibility of the management team's constructing a scale and measuring the causal relationships between indicators to see how causal ordering happens among these variables.

6.3. *Limitations*

The present study has a number of limitations. Firstly, the literature review on the factors of productivity is limited to the data we found in the literature. Therefore, the SEM model is limited with the factors we were able to identify. Secondly, although we have nearly two hundred participants from a software company (Simurg), which can be considered as a substantial sample set in terms of software engineering studies to draw some empirical conclusions, we collected our data from a single software company, which should be tested with different settings for model comparison. Thirdly, there are possibilities for inadvertent sampling bias. Hence, to test the significance of common method error, model was tested for a single factor solution. Fourth, although this study benefits from an adequate sample size according to the SEM literature, we may extend our study to a greater sample size in a wider set of companies. To protect participants' confidentiality, participants were ensured their anonymity. Although there was no enforcement on the company level, we were able to obtain a substantial set of the data. Fifth, this work relies on a self-report measure. Therefore, we were unable to identify whether the same results can be observed with other data collection methods. Moreover, we conducted a cross-sectional study, i.e. our survey was conducted at a single point in time to obtain the variables and the constructs. Accordingly, the direction of causation and causal ordering cannot be determined by the collected data that does not provide significant substantiation for causality. In other words, the model is based on correlational data that cannot be used to draw firm conclusions about the causal relationships. However, field studies and surveys were paired together as multiple methods to reduce the method bias.

7. Conclusions and Future Work

The following conclusions can be drawn from the present study. In the first part of the analysis, the empirical evaluation suggests that there is a significant positive correlation among the productivity constructs, all of which can be explained by the identified factors (see Figure 5). Based on these correlations, the empirical findings in this exploratory field study provide a new understanding for the dimensions of productivity in terms of social productivity and social capital. It is evident that there is a relationship between social capital and social productivity; while social productivity has more impact on productivity. Therefore, this study offers some important insights into the multidimensional nature of software productivity. With regard to practical implications, we conclude that social capital and its transformation to social productivity deserve more attention because this process has the potential to improve software development productivity.

Jones [8] reported that there are as yet no effective methods for investigating software development productivity. Therefore, this research can be considered as a novel attempt to investigate software development productivity with the factors found from the literature and further evaluate the results from an industrial perspective. It makes marked contributions to the field by suggesting an exploratory mechanism for the factors affecting the productivity of a software development organization. Although several previous studies mentioned the importance of social aspects of software development, there are no tools designed particularly for predicting such aspects. Therefore, the implications of social capital on software development productivity has not been deeply investigated. To bridge this gap, we built a tripartite SEM model, and introduce the notion of social productivity of software development, linking it with both social capital and software development productivity.

The team factors with respect to the roles of the participants, which are performed in the second part of the analysis promotes that there were significant correlations between several team based variables and productivity constructs. For example, individuals who are more experienced in the software development organization were observed to work in larger sized teams and they are inclined to think that the social capital was of less importance.

Recently, there have been several empirical investigations into the effects of team size on software development productivity [4]. However, since Brooks [80] initiated a discussion about the possible effects of team size on the productivity of software development, team size has become a central issue for empirical research in software engineering. From a socio-technical perspective, the system dynamics model was developed to investigate human capabilities such as planning a set of possible staffing procedures on a variety of project costs with different schedules [81]. In addition, studies of software development productivity showed the importance of the average team size [33]. Perhaps more importantly, the findings of the current study were consistent with those of Putman [82], who found evidence that the productivity of software development was found to be higher for smaller software teams. Recent evidence from a number of management studies suggests that small teams are performing better [83]. In particular, a study indicated that the size of the most effective software teams varies between 3 to 6 members [84]. Taken together, our findings further support the recent investigations in team size for software development projects.

In this study, we empirically evaluate the hypothesized relationships between the latent constructs and several factors that are affecting them. The results confirm that productivity is highly associated with social productivity, and moderately associated with social capital. There is, however, a moderate correlation observed between social productivity and social capital. In particular, these empirical findings strongly support the notion that social factors dramatically influence software productivity. Returning to the social and organizational issues posed at the beginning of this study, it is now possible to state that most of the factors selected from the literature are affecting the productivity of a software development organization.

Given such theory about the connections between productivity and factors affecting it, it is possible to interpret that proposed comprehensive productivity framework may act as a check list, which can be altered based on empirical measurements to explore the factors affecting organizational productivity. Taken together, these findings enhance the understanding that a company can obtain about productivity factors within their organization. They further offer a useful framework for monitoring productivity indicators. This exploratory field study can assist software engineering managers in exploring company-specific problems, hence improving overall organizational productivity. However, it is recommended that additional research be undertaken to examine the associations between socio-technical factors of productivity and software organizations by concerning the generalizability of these findings.

Future research should therefore concentrate on the investigation of the productivity constructs with more samples from alternative software organizations in several different settings. Such a study would be of great value for understanding the socio-technical aspects of software development, and could be useful for managing the factors that are potentially affecting the productivity of a software development organization.

Acknowledgments

This work is supported, in part, by Science Foundation Ireland grant number 10/CE/I1855 to Lero, the Irish Software Engineering Research Centre (`www.lero.ie`).

References

[1] T. DeMarco and T. Lister, *Peopleware: productive projects and teams.* Dorset House Publishing Company, 1999.
[2] D. A. Tamburri, P. Lago, and H. v. Vliet, "Organizational social structures for software engineering," *ACM Computing Surveys (CSUR)*, vol. 46, no. 1, p. 3, 2013.
[3] Y. Dittrich, C. Floyd, and R. Klischewski, *Social thinking-software practice.* The MIT Press, 2002.
[4] D. Rodríguez, M. Sicilia, E. García, and R. Harrison, "Empirical findings on team size and productivity in software development," *Journal of Systems and Software*, 2011.
[5] M. Yilmaz, R. O'Connor, and P. Clarke, "A systematic approach to the comparison of roles in the software development processes," in *Software Process Improvement and Capability Determination*, vol. 290. Springer-Verlag, 2012, pp. 198–209.
[6] R. Selby, *Software engineering: Barry W. Boehm's lifetime contributions to software development, management, and research.* Wiley-IEEE Computer Society Pr, 2007.
[7] N. Schmitt, "Productivity research in the behavioral and social sciences," *Academy of Management Review*, vol. 10, no. 4, pp. 865–867, 1985.
[8] C. Jones, *Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies.* McGraw-Hill Osborne Media, 2009.
[9] G. Hancock, *Structural equation modeling: A second course.* Information Age Pub Inc, 2006, vol. 1.
[10] S. C. d. B. Sampaio, E. A. Barros, G. S. d. Aquino Junior, M. J. C. e. Silva, and S. R. d. L. Meira, "A review of productivity factors and strategies on software development," *Proceedings of the 2010 Fifth International Conference on Software Engineering Advances*, pp. 196–204, 2010.
[11] K. Petersen, "Measuring and predicting software productivity: A systematic map and review," *Information and Software Technology*, vol. 53, no. 4, pp. 317–343, 2011.
[12] D. Narayan and M. Cassidy, "A dimensional approach to measuring social capital:

development and validation of a social capital inventory," *Current Sociology*, vol. 49, no. 2, p. 59, 2001.
[13] J. Shore and S. Warden, *The Art of Agile Development.* O'Reilly Media, Inc., 2007.
[14] C. Dale and H. Van Der Zee, "Software productivity metrics: who needs them?" *Information and Software Technology*, vol. 34, no. 11, pp. 731–738, 1992.
[15] A. Trendowicz and J. Munch, "Factors influencing software development productivitystate-of-the-art and industrial experiences," *Advances in computers*, vol. 77, pp. 185–241, 2009.
[16] W. Scacchi, "Understanding software productivity: towards a knowledge-based approach," *International Journal of Software Engineering and Knowledge Engineering*, vol. 1, no. 3, pp. 293–321, 1991.
[17] N. Nagappan, B. Murphy, and V. Basili, "The influence of organizational structure on software quality: An empirical case study," in *Proceedings of the 30th International Conference on Software Engineering*, ser. ICSE '08. New York, NY, USA: ACM, 2008, pp. 521–530.
[18] S. Wagner and M. Ruhe, "A structured review of productivity factors in software development," Technische Universitt Mnchen, Tech. Rep., 2008.
[19] B. Boehm, "Improving software productivity," *Computer*, vol. 20, no. 9, pp. 43 –57, sept. 1987.
[20] W. Scacchi, "Understanding software productivity," *Software Engineering and Knowledge Engineering: Trends for the Next Decade*, vol. 4, pp. 273–316, 1995.
[21] W. Yu, D. Smith, and S. Huang, "Software productivity measurements," in *Computer Software and Applications Conference, 1991. COMPSAC'91., Proceedings of the Fifteenth Annual International.* IEEE, 1991, pp. 558–564.
[22] B. Boehm, *Software Engineering Economics.* Prentice Hall, Nov. 1981.
[23] 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, Aug. 2008.
[24] T. Hall, N. Baddoo, S. Beecham, H. Robinson, and H. Sharp, "A systematic review of theory use in studies investigating the motivations of software engineers," *ACM Transactions on Software Engineering and Methodology (TOSEM)*, vol. 18, no. 3, pp. 1–29, 2009.
[25] 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.
[26] B. Boehm and P. Papaccio, "Understanding and controlling software costs," *IEEE Transactions on Software Engineering*, vol. 14, no. 10, pp. 1462–1477, 1988.
[27] K. Maxwell and P. Forselius, "Benchmarking software development productivity," *IEEE Software*, vol. 17, no. 1, pp. 80–88, 2000.
[28] S. Faraj and L. Sproull, "Coordinating expertise in software development teams," *Management Science*, vol. 46, no. 12, pp. 1554–1568, 2000.
[29] V. Basili, L. Briand, and W. Melo, "How reuse influences productivity in object-oriented systems," *Communications of the ACM*, vol. 39, no. 10, pp. 104–116, 1996.
[30] B. Boehm, "Managing software productivity and reuse," *Computer*, vol. 32, no. 9, pp. 111–113, 2002.
[31] D. Simmons, *Software Organization Productivity.* Software Productivity Improvement Laboratory,Department of Computer Science Texas A & M University, 2007.
[32] Z. Jiang, P. Naudé, and C. Comstock, "An investigation on the variation of software development productivity," *International Journal of Computer, Information, and Systems Sciences, and Engineering*, vol. 1, no. 2, pp. 72–81, 2007.

[33] Z. Jiang and C. Comstock, "The factors significant to software development productivity," in *Proceedings of World Academy of Science, Engineering and Technology*, vol. 21. Citeseer, 2007, pp. 160–164.
[34] I. Chiang and V. Mookerjee, "Improving software team productivity," *Communications of the ACM*, vol. 47, no. 5, pp. 89–93, 2004.
[35] C. Behrens, "Measuring the productivity of computer systems development activities with function points," *IEEE Transactions on Software Engineering*, vol. SE-9, no. 6, pp. 648–652, 1983.
[36] N. Christakis and J. Fowler, *Connected: The surprising power of our social networks and how they shape our lives.* Little, Brown and Company, 2009.
[37] M. Yilmaz and R. V. O'Connor, "Social capital as a determinant factor of software development productivity: An empirical study using structural equation modeling," *International Journal of Human Capital and Information Technology Professionals (IJHCITP)*, vol. 3, no. 2, pp. 40–62, 2012.
[38] V. Wickramasinghe and P. Weliwitigoda, "Benefits gained from dimensions of social capital: a study of software developers in sri lanka," *Information Technology & People*, vol. 24, no. 4, pp. 393–413, 2011.
[39] J. Coleman, "Social capital in the creation of human capital," *The American Journal of Sociology*, vol. 94, no. 1, pp. 95–120, 1988.
[40] L. Barnett, "Social Productivity, Law, and the Regulation of Conflicts of Interest in the Investment Industry," *Cardozo Public Law, Policy and Ethics Journal*, vol. 3, p. 793, 2004.
[41] T. Stober and U. Hansmann, *Agile Software Development: Best Practices for Large Software Development Projects.* Springer-Verlag, 2009.
[42] S. Koh and S. Maguire, *Information and Communication Technologies Management in Turbulent Business Environments*, ser. Premier Reference Source. Information Science Reference, 2009.
[43] D. A. Tamburri, P. Kruchten, P. Lago, and H. van Vliet, "What is social debt in software engineering?" in *Cooperative and Human Aspects of Software Engineering (CHASE), 2013 6th International Workshop on.* IEEE, 2013, pp. 93–96.
[44] D. Anderson, *Agile management for software engineering: applying the theory of constraints for business results*, ser. The Coad series. Prentice Hall PTR, 2004.
[45] O. Hazzan and Y. Dubinsky, *Agile Software Engineering*, ser. Undergraduate Topics in Computer Science. Springer, 2008.
[46] A. Kelly, *Changing software development: Learning to become agile.* Wiley, 2008.
[47] D. Churchville, *Agile Thinking: Leading successful software projects and teams.* ExtremePlanner Software, Dec. 2008.
[48] L. Hayduk, *Structural equation modeling with LISREL: Essentials and advances.* Johns Hopkins Univ Pr, 1987.
[49] R. Kline, *Principles and practice of structural equation modeling.* The Guilford Press, 2010.
[50] M. Lovric, *International Encyclopedia of Statistical Science*, 1st ed. Springer, 2011.
[51] R. Schumacker and R. Lomax, *A beginner's guide to structural equation modeling.* Lawrence Erlbaum, 2004, vol. 1.
[52] P. Cuttance and R. Ecob, *Structural modeling by example: Applications in educational, sociological, and behavioral research.* Cambridge Univ Pr, 2009.
[53] A. Vieira, *Interactive LISREL in Practice: Getting Started with a SIMPLIS Approach.* Springer Verlag, 2011.
[54] T. Raykov and G. Marcoulides, *A first course in structural equation modeling.* Lawrence Erlbaum, 2006.

[55] P. Barrett, "Structural equation modelling: Adjudging model fit," *Personality and Individual Differences*, vol. 42, no. 5, pp. 815–824, 2007.

[56] G. Cheung and R. Rensvold, "Evaluating goodness-of-fit indexes for testing measurement invariance," *Structural Equation Modeling*, vol. 9, no. 2, pp. 233–255, 2002.

[57] M. Browne and R. Cudeck, "Alternative ways of assessing model fit." *Testing structural equation models (1993) Bollen, Kenneth A.; Long, J. Scott. Newbury Park: Sage Publications.*, 1993.

[58] R. Bagozzi and Y. Yi, "On the evaluation of structural equation models," *Journal of the academy of marketing science*, vol. 16, no. 1, pp. 74–94, 1988.

[59] J. Cote, R. Netemeyer, and P. Bentler, "Improving model fit by correlating errors," *Journal of Consumer Psychology*, vol. 10, no. 1/2, pp. 87–88, 2001.

[60] R. Ping, "On assuring valid measures for theoretical models using survey data," *Journal of Business Research*, vol. 57, no. 2, pp. 125–141, 2004.

[61] L. Given, *The Sage encyclopedia of qualitative research methods.* Sage Publications, 2008.

[62] J. Smith, D. Wright, and G. Breakwell, *Research methods in psychology.* SAGE Publications, 2012.

[63] K. Joreskog and D. Sorbom, *LISREL 8: users reference guide. Lincolnwood, IL: Scientific Software International.* Inc, 2001.

[64] L. Cronbach, "Coefficient alpha and the internal structure of tests," *Psychometrika*, vol. 16, no. 3, pp. 297–334, 1951.

[65] S. VanderStoep and D. Johnson, *Research methods for everyday life: Blending qualitative and quantitative approaches.* Jossey-Bass, 2008, vol. 24.

[66] A. Lehman, *JMP for basic univariate and multivariate statistics: a step-by-step guide.* SAS Publishing, 2005.

[67] I. H. Bernstein and G. Teng, "Factoring items and factoring scales are different: Spurious evidence for multidimensionality due to item categorization," *Psychological Bulletin*, vol. 105, no. 3, pp. 467–477, 1989.

[68] P. M. Bentler and C.-P. Chou, "Practical issues in structural modeling," *Sociological Methods & Research*, vol. 16, no. 1, pp. 78–117, 1987.

[69] B. Thompson and J. G. Melancon, "Using item" testlets"/" parcels" in confirmatory factor analysis: An example using the ppsdq-78." *Paper presented at the annual meeting of the Mid-South Educational Research Association*, vol. 1, p. 27, 1996.

[70] S. W. Huck, *Reading statistics and research.* Pearson, 2012.

[71] M. Zelkowitz, *Advances in Computers: Highly Dependable Software.* Gulf Professional Publishing, 2003.

[72] S. Kan, *Metrics and Models in Software Quality Engineering, Second Edition.* Prentice Hall, 2003.

[73] A. Albrecht, "Measuring application development productivity," in *Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium*, vol. 10. SHARE Inc. and GUIDE International Corp. Monterey, CA, 1979, pp. 83–92.

[74] M. Bundschuh and C. Dekkers, *The IT measurement compendium: estimating and benchmarking success with functional size measurement.* Springer, 2008.

[75] L. Foulds, M. Quaddus, and M. West, "Structural equation modelling of large-scale information system application development productivity: the Hong Kong experience," in *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on.* IEEE, 2007, pp. 724–731.

[76] S. Ng, Y. Wong, and J. Wong, "A structural equation model of feasibility evaluation and project success for public–private partnerships in hong kong," *Engineering Management, IEEE Transactions on*, vol. 57, no. 2, pp. 310–322, 2010.

[77] M. Yilmaz and R. V. O'Connor, "An empirical investigation into social productivity of a software process: An approach by using the structural equation modeling," in *Proceedings of the 18th European System and Software Process Improvement and Innovation Conference (EuroSPI 2011)*, vol. 172. Springer-Verlag, 2011, pp. 155–166.

[78] B. Kitchenham, S. Pfleeger, L. Pickard, P. Jones, D. Hoaglin, K. El Emam, and J. Rosenberg, "Preliminary guidelines for empirical research in software engineering," *IEEE Transactions on Software Engineering*, vol. 28, no. 8, pp. 721–734, Aug. 2002.

[79] A. Bryant and K. Charmaz, *The Sage handbook of grounded theory.* Sage Publications, 2010.

[80] F. P. Brooks, *The Mythical Man-Month: Essays on Software Engineering, Anniversary Edition (2nd Edition).* Addison-Wesley Professional, 1995.

[81] T. Abdel-Hamid and S. E. Madnick, *Software project dynamics: an integrated approach.* Prentice Hall Englewood Cliffs, NJ, 1991.

[82] L. Putnam, "A general empirical solution to the macro software sizing and estimating problem," *IEEE Transactions on Software Engineering*, no. 4, pp. 345–361, 1978.

[83] R. Daft and D. Marcic, *Understanding management.* South-Western Pub, 2006.

[84] M. Hoegl, "Smaller teams–better teamwork: How to keep project teams small," *Business Horizons*, vol. 48, no. 3, pp. 209–214, 2005.