Journal of Universal Computer Science, vol. 27, no. 2 (2021), 111-134
submitted: 6/4/2020, accepted: 1/2/2021, appeared: 28/2/2021 CC BY-ND 4.0

An Instrument for Measuring Perception about Social and Human Factors that Influence Software Development Productivity

Liliana Machuca-Villegas
(Universidad del Valle, Cali, Colombia
Universidad de Medellín, Medellín, Colombia
https://orcid.org/0000-0002-5150-4570, liliana.machuca@correounivalle.edu.co)

Gloria Piedad Gasca-Hurtado
(Universidad de Medellín, Medellín, Colombia
https://orcid.org/0000-0003-0157-1959, gpgasca@udem.edu.co)

Solbey Morillo Puente
(Universidad de Medellín, Medellín, Colombia
https://orcid.org/0000-0002-2129-1121, smorillo@udem.edu.co)

Luz Marcela Restrepo Tamayo
(Universidad de Medellín, Medellín, Colombia
https://orcid.org/0000-0001-9123-9030, lmrestrepo@udem.edu.co)

Abstract: In terms of productivity in software development, there is specific interest in identifying its influencing factors. For this purpose, several classification approaches have been previously used, which have already recognized technical factors, organizational factors, product factors, project factors, and personal factors. However, these approaches often focus on technical factors over social and human factors (SHFs). Nevertheless, in addition to the obvious technical aspects, the software development process involves problem-solving skills and cognitive aspects and social interaction. In this sense, determining SHFs can lead to software organizations designing strategies for improving team productivity. In this study, we first conducted a preliminary classification of the SHFs identified in the literature. Because this study seeks to assess the factors from the standpoint of software development professionals, we developed and validated an instrument to measure the perception of software development team members about SHFs that may be affecting their productivity. For this purpose, the first four stages of survey-based research were followed: objective definition, survey design, instrument construction, instrument validity, and reliability assessment. The instrument included 79 items assessing 13 different SHFs. After assessing both their validity and reliability, the results demonstrated that the instrument is a valid and reliable tool for measuring SHFs perception among software development team members.

Keywords: Social and Human Factor, Software Development Productivity, Survey Instrument.
Categories: A.1, D.2.0, D.2.9
DOI: 10.3897/jucs.65102

1 Introduction

Software development is considered a people-centered activity that does not only include technical or technological aspects. The software product generated from this process incorporates problem-solving skills, cognitive aspects, and social interaction [Capretz, 14]. From this perspective, social and human factors (SHFs) play an important role in software engineering and may even affect the productivity of the software development team. Therefore, "soft skills" not only necessary and important in software engineering [Matturro, 19] and software process improvement [Muñoz, 14] but also in software development education because students are fostered to effectively and efficiently perform professional activities in a globalized world [Zurita, 16].

Regarding software development productivity, there is specific interest in identifying its influencing factors. Studies report different classification schemes for these factors such as technical and soft factors [Wagner, 08]; influence and context factors [Trendowicz, 09]; team emerging states factors, individual factors, and support task factors [Dutra, 15]; product factors, people factors, and project factors [De Barros, 10]; and individual factors, interpersonal factors, and organizational factors [Pirzadeh, 10]. These classifications include patterns that are used to separate factors using technical, organizational, or software development process features. Other researchers even discuss person-related classification patterns or non-technical factors known as soft factors. However, SHFs are not explicitly addressed in any manner that may facilitate their identification and adoption to propose strategies for improving the productivity of the software development team.

In this sense, 57 SHFs were identified in the literature [Machuca-Villegas, 19]. As part of this study, these SHFs were assessed from an organizational psychology and a software engineering perspective; considering their significance, any relationship may exist between them and their complexity in the context of this study. Consequently, 13 SHFs were identified for subsequent assessment from the perception of software development professionals. The perception assessment seeks to validate whether the SHFs reported in the literature affect professionals in terms of their productivity. Similarly, based on the results obtained, it expects to identify prevailing SHFs. Therefore, this study discusses the design and validation of an instrument developed for measuring the perceptions of software development team members on the SHFs that influence their productivity. Identifying these SHFs and facing their corresponding challenges can help software organizations to reduce software project management issues, reduce development time, reduce product costs, and ultimately improve team productivity.

The paper is structured as follows. In Section 2, studies focusing on the SHFs that influence software development productivity are listed. In Section 3, the 13 SHFs included in the instrument from a theoretical perspective are discussed. In Section 4, the survey design and validation process are described. Then, in Section 5, the results of the instrument evaluation are provided. Finally, in Section 6, the conclusions and future work are discussed.

2 Related Work

This section describes some previous research projects that address factors influencing software development productivity and provides certain examples of survey-based studies that have been conducted in this context.

Some studies reported a tertiary review on factors affecting productivity [Oliveira, 18]. However, based on the reviewed studies, the tertiary review mentions the lack of a factor common classification. As part of their study, they propose a factor classification categorized into organizational and human factors, highlighting human factors as determining factors for improving the productivity of a software developer.

In [Pirzadeh, 10], authors reported results from a systematic literature review on human factors in software development. The assessment of the primary studies was aimed at identifying and characterizing human factors that exert influence on the software development process from the standpoint of software development cycle and management. Moreover, this study assessed the approach level for these factors by classifying them at organizational, interpersonal, and individual levels in which the individual level was the most prominent.

The systematic review performed by [Wagner, 08] showed a list of productivity factors classified into two categories: technical and non-technical (*soft*) factors. Among non-technical factors, corporate culture, team culture, team member capabilities and expertise, the environment and project factors are important. Furthermore, as a productivity success factor, this review mentioned the importance of communication.

Likewise, [De Barros, 10] reviewed productivity factors and describes strategies to maximize or minimize factor influence whether positive or negative. The factor classification was based on Boehm's proposal for establishing productivity boosting strategies.

Further, Murphy-Hill et al. [Murphy-Hill, 19] polled 622 software developers from three organizations on productivity. In this study, the prevailing factors were motivation, peer support for new ideas, and receiving useful performance reviews. This assessment included questions on productivity, 48 productivity factors, and demographic variables. Then, the factors were categorized into practices, focus, experience, job, work, capabilities, people, project, software, and context. In fact, both format and style of survey items served as a foundation for designing the instrument proposed.

Similarly, Canedo and Santos [Canedo, 19] studied factors affecting software development productivity projects and open-source projects. For this purpose, regarding team productivity factors, they conducted a literature review and polled software developers on their perceptions. As part of their results, they identified 37 factors influencing productivity, which were further categorized into people, product, organization, and open-source projects. In this paper, we assessed the questions used in this survey, however, they did not perfectly fit our SHF definitions, for this reason, we create new items to measure SHF.

For agile development, several factors exerting influence on productivity have been observed. Fatema and Sakib [Fatema, 18] used system dynamics to model productivity in agile development teams and identified 38 factors that affected team work productivity. Here team effectiveness, team management, motivation, and customer satisfaction were identified as the most outstanding factors. Furthermore, their work included a survey used to measure the perception from agile team members on the influence from productivity factors. This survey also examined the level of influence (Low, Medium, and High) of these factors. For designing our instrument, the survey used by [Fatema 18] was not considered because the questions only mentioned or presented minimal details of each factor.

Iqbal et al. [Iqbalm, 19] performed an empirical analysis on factors influencing agile teams. This study describes factors that arise with certain frequency in an agile development team such as inter-team relationships, team speed, team vision, and other factors related to team member and team lead roles, as well as their influence on productivity. The instrument used served as a guide for designing certain questions included in our survey.

To summarize, even when these studies discussed factors that influenced software development productivity, they only focused on their identification and on generating recommendations for increasing team productivity. There is no explicit definition of SHFs that may foster improvement strategies; therefore, a classification of SHFs specifically aimed at fostering software development team productivity for establishing concrete and decisive improvement strategies and actions for SHFs is a necessary input.

As for survey-based studies, some instruments used in the literature reviewed can be applied in cases where it may want to measure perceptions on influencing factors or measure productivity from the developer's standpoint. Although these instruments embody a broad and diverse set of factors influencing software development productivity, it was unable to align them to the study's requirements because we could not determine which questions from the surveys assessed might conform to the SHFs identified by us. However, the instruments reported in our literature review served as foundation for designing our instrument.

3 SHFs that Influence Software Development Productivity

[Machuca-Villegas, 19] identified 57 SHFs from a literature review. Based on an entire semester and in several frequent meetings, these factors were assessed by a psychologist along with the research team (two software engineers and two statisticians). As part of this study, SHFs were reviewed from the perspective of organizational psychology and software engineering, assessing their significance, the relationship that may exist between them, and their complexity within the context of this research. From this process, 13 SHFs were selected and defined. Table 1 lists the factors with a brief definition, and these definitions were constructed as per their psychological rationale and the previously conducted literature review.

Name	**Definition**
Communication	Within human social nature, communication is defined as the way in which one subject is linked to another [Vallejo-Nágera, 02]. In this sense, communication is mediating all human interaction as the bridge that allows thoughts, emotions, and information to be transmitted [Consuegra, 10].
Collaboration	Collaboration is directly related to the feeling of being supported by others, thus implying an impulse to act together and maintain a spirit of solidarity. In a work team, collaboration manifests itself when several individuals work together for mutual benefit [Tomasello, 10].
Commitment	Commitment is the level of responsibility that a subject is willing to assume in their tasks within their work team, just as the work team is responsible for the goals set within the project.
Motivation	Motivation is the factor that moves people toward actions [Vallejo-Nágera, 02]. Motivation is intrinsic to each person, evidences magnitude, strength and duration, varies depending on the targets and goals, and determines a part of human behavior [Hernández Lopez, 14].
Work satisfaction	Work satisfaction is determined by the difference between what the person wants and what the person receives from work [Judge, 10]. It depends on context and the challenging and stimulating activities of each position at work.
Leadership	Leadership is the ability to influence other people within the work team to achieve goals and objectives [Vivas, 09].
Innovation (creativity)	Based on creative thinking, innovation can manifest itself in an individual or work team as the ability to establish new relationships between events or to integrate them in a different, original, and innovating manner. Innovation is related to creativity, which involves the creation of something new, different, and of a certain value, based on the experiences and knowledge the person has previously acquired [Vallejo-Nágera, 02].
Emotional intelligence	The ability exhibited by people to properly identify and process their emotions so that they are in control of their behavior rather than letting their emotions control them.
Autonomy	Capability of making decisions at the workplace without relying on management. Autonomy is directly related to the freedom granted to employees and the work team to make project decisions and the way in which work is conducted [Dutra, 15].

Empathy and Interpersonal Relationship	Empathy is an ability in which a subject identifies with another person or group and shares the same state of mood. During social interaction, some values, such as respect, are raised, which indicates a general consensus on group rules, setting limits, and establishing a floor on which all participants are both providers and beneficiaries [Vallejo-Nágera, 02]. In this manner, proper interpersonal relationships are fostered as a human social nature within the socialization dynamics among work team members.
Team Cohesion	Work team cohesion refers to the existing level of integration between its members so that all efforts are directed toward the same common goal [Vallejo-Nágera, 02].
Capabilities and Experiences in software development process	It is the knowledge and experience in the analysis, design, and development of a software product according to the role played by each member of the team. It involves familiarity with the application domain, the hardware and software platforms, and the programing tools and languages used.
Capabilities and Experiences in software project management	The application of knowledge, skills, tools, and techniques in project activities to meet project requirements [PMI, 13].

Table 1. SHFs influencing software productivity

4 Methodology

This section describes the process followed for developing and validating the instrument that was designed. For this purpose, this study will use four of the six survey-based research stages described in [Kitchenham, 08]. Within this context, a survey-based study was selected because it is a research method that, in addition to gathering information, describes, compares, or explains knowledge, attitudes, or behaviors [Kitchenham, 08]. In our particular case, the goal is to gather information on the perception of software development teams about SHFs that affect their productivity and, based on this information, compare these perceptions against the different variables proposed by software developers.

4.1 Setting Objectives

The first step in developing a survey-based study is setting objectives. Hence, our expected purpose and the research questions we expect to respond through the survey must be clearly defined.

As part of this study, the purpose of the survey, resource availability, scope, the target population, and the research variables were determined. This allowed us to set our research objective, which is to measure the perception from software development professionals on the SHFs that influence their productivity. For this purpose, we designed an instrument to measure how software development team members perceive SHFs in terms of their productivity. Then, the corresponding assessment factors were defined as per the perspectives from organizational psychology and software engineering, thus deriving survey items for each factor that was selected. Once the survey items had been completed, subject matter experts were asked to assess based on relevance, representativeness, and clarity of the items. Based on expert assessments, a content validity coefficient was calculated. We expect with the application of the instrument to the sample: i) to determine the perceptions from software developers, and ii) to provide a foundation for identifying prevailing social and human factors.

Furthermore, this study is framed in the following research question: Which SHFs influence the productivity of the software development team?

This instrument can be used by development teams who may wish to identify and rank SHFs that affect their productivity. However, note that this instrument does not measure productivity itself.

4.2 Survey Design

This stage establishes the type of study design, which may be focused on a cross-sectional or a longitudinal study. Similarly, this step defines the survey implementation method that should be used.

In this light, our study is defined to be cross-sectional observational. Participants will be polled based on their experience as software developers, and then the survey will be conducted in a single point in time. This type of study favors time and cost investments because its execution requires less time than a longitudinal study and does not imply monitoring of individuals. In fact, most software engineering surveys used are quite similar in type and design [Kitchenham, 08].

Regarding the survey implementation method, we decided to use a web-based self-administered questionnaire because this format facilitates survey distribution and coverage. After assessing the advantages and disadvantages of different survey application systems, we selected Google Forms as a platform.

4.3 Survey Instrument Development

At this stage, related literature about the research topic must be taken into account, in order to identify similar instruments that may be applied to the context of the research being carried out. This review will help to determine whether a new instrument is required and to identify the methods used by other researchers in similar works. If a new instrument is required, questionnaires are often the method of choice. In this case, we must carefully consider the type of question used and the type of answer we expect to receive. Furthermore, we must consider the format of the survey, its length, and any biases from the researcher.

In the literature, certain studies addressed the factors that influence productivity wherein instruments were used either for measuring these factors or to measure productivity. In fact, certain studies, such as [Fatema, 18] [Murphy-Hill, 19] [Iqbal, 19] [Canedo, 19] were considered as survey design references. However, because none of these instruments really fit our SHFs definitions, it was not possible to adapt them in whole or in part. Nevertheless, the questions from these surveys served as a foundation for designing our own survey items.

Once we had decided to create a new survey, we selected our theoretical approach, which was organizational psychology, and then settled on a definition for our SHFs [Section 3]. Then, the survey items were written, based on a four-point Likert scale: strongly disagree; disagree; agree; strongly agree. The items were derived from the theoretical definition for each SHF. At least five different items were written for each factor because using multiple items improve survey reliability by reducing the respondent errors and increase the accuracy with which each concept is measured [Kitchenham, 08].

Based on the theoretical definition that was selected, each research team member proposed items to measure the corresponding SHFs, thus building an item bank. Once the item bank was completed, periodic item review meetings were held in which item–factor correspondence was assessed and the proper wording for each item was reviewed. This allowed the entire research team to clean the bank of items and select only those items that should be included in the survey based on the opinion of the investigation team. The cyclical dynamics of each meeting are as follows: i) reading the definition of the SHF; ii) reading an item proposed by a research team member; iii) discussing item relevance; iv) consensual decision on whether the item should be included or not; v) reviewing the wording of the item selected; and vi) reading the next item proposed by the research team member. Through this item proposal and review process, we were able to mitigate researcher biases regarding survey items.

In the end, a total of 123 items were selected to assess the 13 SHFs, including both affirmative and negative statements as per the guidelines provided by Aiken [Aiken, 96]. Table 2 presents an example of two of the items used to assess SHFs. Each item was preceded by the general statement "To improve the productivity of software development teams …." Although the number of items is high, the research team decided to use all of them to provide a broad set of options for the validation process, thus expecting that certain items may have to be removed from the process.

Social and Human Factors	**Item**
Communication	• Communication between team members is important • Oral or written communication between members of the software development team should be avoided (REVERSE ITEM)
Commitment	• Work team performs tasks required to guarantee project success based on the set objectives. • The team is committed, accountable for the deliverables, and is readily available to perform any task required to achieve their goals.
Motivation	• Tasks assigned to team members should be perceived as interesting • Team members should feel that tasks they perform are part of something important
Work satisfaction	• The activities performed by team members should contribute to both personal and professional growth • Team members are satisfied with the activities they perform

Emotional intelligence	• Each team member should face challenges without letting them negatively affect their emotions and the emotions of their peers • Each team member should be able to adapt themselves to project-related changes
Collaboration	• Team members should collaboratively work to jointly achieve project goals • Team should be characterized by endorsing and supporting all members
Team Cohesion	• It is important that members feel identified with the team by participating in an autonomous and motivated manner • Each member of the software development team should place their best skills at the service of the team's goals
Empathy and Interpersonal Relationship	• Social activities, both within and outside the workplace, are necessary • Each member should participate in workplace activities.
Leadership	• The team allows any member to lead activities within the project • The leader should promote positive attitudes and build trust among team members
Innovation	• Team members enjoy performing new activities that have not been previously attempted • Team members should avoid using solutions that have not been satisfactorily tested (REVERSE ITEM)
Autonomy	• All team members should be granted the autonomy to make any decision without direct supervision • The team can organize itself to set and meet their goals
Capabilities and Experiences in Software Development Process	• The work team should be updated regarding the best tools and practices to perform the tasks • Team members should be familiarized or have worked on similar applications
Capabilities and Experiences in Software Project Management	• Each team member should adequately manage deadlines • Each team member should be able to plan, execute, or control project activities

Table 2. Sample Survey Items

4.4 Survey Instrument Evaluation

At this stage, the instrument is assessed before its application. We reviewed whether the items were understandable and determined the validity and reliability of the instrument. We describe the process used to assess instrument reliability and validity below. Then, the first draft of the instrument was applied to a pilot sample.

4.4.1 Validity Assessment

Hernández et al. [Hernández, 14] suggest conducting a literature review to identify how other researchers have measured the variable as the first step in order to determine content validity. Then, based on this review, a "universe of possible items or reactions may be written to measure the variable and its dimensions (the universe should be as exhaustive as possible). Later, researchers acquainted with the variable must be engaged to determine whether the universe is truly exhaustive" (p. 209). The items included in the instrument were generated from the theoretical review of SHFs reported in software development productivity literature.

Validity refers to the degree to which an instrument actually measures the variable you intend to measure based on several types of evidence: content, construct and criteria. For our instrument, we assessed its content validity by understanding it as the degree to which our instrument reflects a specific content domain of what we intended to measure. For this purpose, we calculated the content validity coefficient (CVC) [Pedrosa, 13]. Because many factors may affect instrument validity and reliability, we must implement validation methodological processes to reduce application and interpretation biases before applying our measuring instrument based on criteria and theories that will enable us to obtain valid, reliable, and objective results [Flórez, 15].

To account for content validity, the following procedure was used for the proposed instrument. First, a literature review of existing academic and scientific papers and surveys on the subject matter was conducted. From these papers, we selected 13 SHFs, and, for each factor, 123 items were written. Then, to determine whether the items written for each factor effectively represented what we intended to measure and were aligned with the objectives set forth, the instrument was submitted to the consideration of subject matter experts (SME) [Pedrosa, 13] who were asked to validate their content. For this purpose, we selected five SME reviewers based on their expertise on software development and in the linguistic and psychological fields. The survey was provided in a validation format, i.e., it did not include the response options, but rather spaces where SME reviewers may write their assessment of each item, and any comments or the corresponding theoretical definitions. SME reviewers were instructed to read the instructions and assess each item as listed in Table 3. Then, the data received from these five reviewers were transcribed into an Excel spreadsheet for CVC calculation. The corresponding results are reported in the following section.

Score	Assessment Criteria
1	• Not relevant so it should be removed • This item is not clear • This item has no logical relationship with the factor. This item has a tangential relationship with the factor • This item can be removed without affecting the factor measurement
2	• It must be rewritten • The item requires several modifications or a very large modification in terms of wording or structure • A very specific modification is required for some wording • The item has a moderate relationship with the factor it is measuring • The item is relatively important
3	• This item is relevant • This item is clear with proper semantics and syntax • This item is completely related to the factor being measured • This item is very relevant and must be included

Table 3. Item assessment criteria

4.4.2 Reliability Assessment

An instrument is reliable if, when administered several times, approximately the same distribution of results is obtained each time it is applied [Kitchenham, 08]. In this sense, reliability is related to measurement consistency. The less results change, the more consistent the measurement and therefore more accurate [Espinosa, 08].

For this study, we estimated reliability via internal consistency. This is a procedure that measures the correlations between items included in the same test, i.e., the items produce similar results in repeated applications. There are several procedures used to determine reliability through internal consistency [Pérez, 09]:

- The Spearman–Brown procedure (correlation between halves)
- The Rulon procedure (variance between differences)
- The Guttman scaling procedure (variance between split halves)
- The Kuder–Richardson procedure (item intercorrelations)
- The Cronbach's alpha procedure (the sum of variances of the individual items divided by total variance weighed by the total number of items)

From the abovementioned procedures, we selected Cronbach's alpha [Quero, 10], which is based on the covariances between items measured through any multiple-choice scale [Martínez, 96]. This measurement requires a single instrument administration and produces values ranging from 0 to 1 [Molina, 13]. If Cronbach's alpha is high (> 0.7), items are highly correlated and together measure a single construct [Kitchenham, 08]. The alpha coefficient is derived from the following equation [Martínez, 96]:

$$\propto = \frac{n}{n-1}\left(1 - \frac{\sum \sigma_j^2}{\sigma_x^2}\right)$$

where *n* is the number of test items, σ^2 represents item variance (*j*= 1,2, …, n), and σ_x^2 represents total test variance. The following section discusses Cronbach's alpha results for each factor and for the entire instrument.

4.4.3 Pilot Sample Application

After validating the SHFs to be included in the instrument, we designed the draft version of the survey using Google Forms. Once hosted on the Google platform, the research team reviewed the items once again to streamline wording, structure, and operational details. Then, the corresponding adjustments were made, and the final file was structured. The instrument includes the following sections:

- A contextualization, data management authorization, and reliability section.
- A personal data section.
- One section for each factor. We agreed that the instrument must specify the name of each factor measured.
- An additional data section.

This first version of the instrument was applied to a pilot sample of 23 participants in Medellín, Colombia. The purpose of this test was to estimate the time respondents required to answer the survey, review the results structure generated by the platform, and identify opportunities for improvement. The corresponding results are discussed in the following section.

5 Results

This section discusses instrument evaluation results based on its validity and reliability assessments, as well as its application to the pilot sample and final version of the instrument.

5.1 Validity and Reliability Assessment Results for the Instrument Designed

The content validity coefficient (CVC) was calculated from the data obtained from the five SME reviewers. Through the CVC, the degree of agreement among the experts was determined for each of the items included in the instrument as well as for the total instrument. These results allowed us to identify those items that exceed 0.7 in a scale from 0 to 1 and therefore must be included in the final version of the instrument. All items that reported a CVC under 0.8 were deleted (24 items). Moreover, the reviewers suggested merging several items because their content was similar, thus reducing the final item number to 79 items.

Table 4 below lists the CVC results, the number of items per each SHF, the percentage of positive valences worked for the SHFs, the factors that have been removed based on their CVC, and the number of items per SHF included in the final version of the instrument. Because the CVC for each factor exceed 0.80, the instrument is deemed valid [Pedrosa, 13].

SHFs	**Number of Items**	**% items with positive valences**	**Removed (CVC <0.80)**	**Accepted***	**CVC**
Communication	13	76.9	6	6	0.81
Collaboration	7	85.7	2	4	0.85
Commitment	7	85.7	1	6	0.85
Motivation	11	100.0	4	5	0.83
Work satisfaction	13	100.0	2	7	0.91
Leadership	12	83.3	3	7	0.92
Innovation (creativity)	11	81.8	1	6	0.94
Emotional intelligence	10	100.0	0	8	0.97
Autonomy	9	100.0	4	5	0.86
Empathy and Interpersonal Relationship	12	91.7	1	9	0.89
Team Cohesion	7	100.0	0	6	0.91
Capabilities and Experiences in software development process	7	100.0	0	7	0.94
Capabilities and Experiences in software project management	6	100.0	0	4	0.97
Total	123	91.9	24	79	0.86

Table 4. Instrument content validity results

**Some items were merged, therefore the total of accepted items does not correspond to the difference between the number of items and the removed items*

Table 5 below lists the assessment values from the SME reviewers. Values were calculated from the Excel spreadsheet using SPSS. Here, the SME reviewers approved a high number of the items assessed with percentages ranging from 61.6% to 83.2%. However, two of the five SME reviewers believed that no item should be removed, while reviewer 4 suggested that 16.8% should be removed, i.e., the highest percentage from any of the reviewers.

Reviewer	**Assessment**					
	Not relevant		**Needs rewording**		**This item is relevant**	
	No. Items	%	No. Items	%	No. Items	%
REVIEWER 1	0	--	21	16.8	104	83.2
REVIEWER 2	5	4.0	17	13.6	103	82.4
REVIEWER 3	15	12.0	33	26.4	77	61.6
REVIEWER 4	21	16.8	24	19.2	80	64.0
REVIEWER 5	0	--	21	16.8	104	83.2

Table 5. Instrument Item Assessment from Reviewers

Table 6 shows the CVC values such as the minimum, maximum and average (0.86) CVC values. Because the average CVC value exceeds the suggested optimum value (0.8), the instrument is deemed to be valid [Pedrosa, 13].

Minimum Rating	**Maximum Rating**	**Average CVC**	**Number of Items**
0.53	1.00	0.86	123

Table 6. Summary of Instrument Item Assessment Statistics

After the instrument was considered to be valid in terms of content, we proceeded to calculate the reliability for each factor and for the instrument as a whole using Cronbach's alpha coefficient based on the covariances between items. For these calculations, we used SPSS (version 21 for Windows) and the corresponding results are summarized in Table 7. Because all of Cronbach's alpha values are within the range suggested by [Oviedo, 05] for one-dimensional scales (0.7–0.9), we can safely assume that all instrument items exhibit proper internal consistency.

Social and Human Factors	Number of Items	Cronbach's alpha
Communication	13	0.743
Collaboration	7	0.753
Commitment	7	0.782
Motivation	11	0.690
Work satisfaction	13	0.753
Leadership	12	0.761
Innovation (creativity)	11	0.751

Emotional intelligence	10	0.804
Autonomy	9	0.749
Empathy and Interpersonal Relationship	12	0.713
Team Cohesion	7	0.797
Capabilities and Experiences in software development process	7	0.788
Capabilities and Experiences in software project management	6	0.793
Total	123	0.958

Table 7. Instrument Reliability Results

5.2 Assessment of Pilot Sample Application Results

Rather than assessing the responses from pilot sample respondents, we focused on response time and the difficulties that might affect proper survey completion. According to our results, the average response time was 25 min and certain items required some wording adjustments. Some of the adjustments suggested were as follows:

- All wording must be consistent. All items must be reworded using "should"
- All factors and questions must be numbered
- In the Work Team Cohesion factor, the "Activities should be performed in a timely and comprehensive manner" must be rewritten as "Activities should be performed in a timely manner and with participation from all stakeholders"
- All questions and answers must be separately worded rather than using the matrix used in the original version of the instrument.

Based on these results, we decided to restructure the entire instrument and apply it to a different group of participants because this would provide additional feedback and allow us to make the corresponding adjustments. Once the second pilot was applied, additional minor wording adjustments were made, and we were able to determine that these wording changes did not significantly change response times.

5.3 Instrument Final Version

From the results of the instrument evaluation and the pilot sample, we generated a new version. With this instrument, it is expected to measure the perception of software development team members about 13 SHFs that can influence their productivity, therefore its purpose focuses, particularly, on aspects associated with the person, in contrast to some research that they encompass other types of factors that influence productivity [Canedo, 19] [Fatema, 18] [Iqbalm, 19] [Murphy-Hill, 19]. This particularity allows us to go deeper into each SHF and expect more detailed results.

This instrument includes additional SHFs to those included in [Canedo, 19] [Fatema, 18] [Iqbalm, 19] and [Murphy-Hill, 19]. For instance, in [Canedo, 19], Canedo and Santos included eight factors associated with the person, five of them are related to the SHFs Communication, Collaboration, Team Cohesion, Motivation, Capabilities and Experiences in software development process, Capabilities and Experiences in software project management.

The results contribute to the researchers focused on studying the factors that influence the productivity of software development. It adds value to the related works. The instruments used in the literature reviewed were a reference for the instrument design described in this article. The instrument's final version is presented below.

Title

Perceptions from the software development team members

Contextualization, Data Management Authorization, and Reliability Section.

As part of the framework of the "Classification and Influence of Social and Human Factors on Software Development Team Productivity in Small and Medium IT Businesses" project, this study seeks to identify these factors; therefore, we hereby request and acknowledge your valuable participation in this study.

As you are working in our area of interest, you have been randomly selected to respond this survey. Note that there are no right or wrong answers. We would only like to know your perceptions based on the below factors. We guarantee that your answers will remain strictly confidential and in full compliance with the applicable laws on statistical secrecy and protection of personal data. All information provided herein is anonymous and confidential, and all individual survey will be destroyed and data will be aggregated with all other respondents.

Would you like to continue? YES NO

General Data.

Please answer each of the following questions according to your current situation

- Country
- City
- Age
- Gender
- Maximum academic level reached
- Profession
- Position
- Years in the company
- Years of professional experience in software development

Questions by Social and Human Factors

Select the most appropriate answer for each of the following statements, according to this scale: Strongly Disagree; Disagree; Agree; Strongly Agree.

1. Communication

1.1 To improve productivity in software development processes, communication between team members is paramount.

1.2 Project objectives and their corresponding activities should be explicit and clear to all team members to improve productivity in software development processes.
1.3 To improve the productivity in software development processes, each task should have a clearly identified person in charge.
1.4 Team members should maintain fluid communication to improve productivity in software development processes.
1.5 To improve productivity in software development processes, team members should be informed in a timely manner about goal progress and objective achievements.
1.6 To improve productivity in software development processes, a communication protocol between the team members and external personnel should be defined.

2. Commitment
2.1 To improve productivity in software development processes, the work team should perform the tasks required to achieve project success according to the objectives that were set.
2.2 The members should be assigned a level of responsibility that facilitates assuming their tasks in favor of team objectives to improve productivity in software development processes.
2.3 To improve productivity in software development processes, the team should be clear about its responsibility for completing tasks and be willing to help as necessary.
2.4 All team members should be accountable for the obtained results, fulfill their duties, and admit their mistakes to improve productivity in software development processes.
2.5 To improve productivity in software development processes, the team members should completely and promptly complete all of the tasks that were assigned.

3. Motivation
3.1 To improve productivity in software development processes, positive attitudes are essential to achieve the project's objectives.
3.2 The tasks assigned to team members should be perceived as interesting and challenging to improve productivity in software development processes.
3.3 To improve productivity in software development processes, team members should believe that the tasks they perform contribute to achieve objectives.
3.4 Team members should receive rewards for activities performed because this improves productivity in software development processes.
3.5 To improve productivity in software development processes, good furniture, computer equipment, and optimal environmental conditions should be available.

4. Work Satisfaction
4.1 To improve productivity in software development processes, the contribution from each team member should be recognized.
4.2 The activities performed by team members should contribute to their personal and professional growth and improve productivity in software development processes.
4.3 To improve productivity in software development processes, team members should be satisfied with the equitable distribution of work.

4.4 Team members should be satisfied with the activities they perform to improve productivity in software development processes.
4.5 To improve productivity in software development processes, the tasks assigned to each team member should largely correspond to what each one wants to do.
4.6 To improve productivity in software development processes, tasks should be assigned according to the profile of each team member.
4.7 Team members should be satisfied with the possibility of acquiring additional knowledge about software development to improve their productivity.

5. Emotional Intelligence
5.1 To improve productivity in software development processes, each team member should be able to adapt themselves to project changes.
5.2 To improve productivity in software development processes, each team member should be able to express disagreement in a timely manner and to the right person.
5.3 To improve productivity in software development processes, team members are required to perform their activities, even when they become difficult.
5.4 Team members need to know how to appropriately manage their emotions to improve productivity in software development processes.
5.5 To improve productivity in software development processes, it is important for each team member to listen to criticism and act accordingly.
5.6 To improve productivity in software development processes, the team should be appropriately able to solve conflicts.
5.7 To improve productivity in software development processes, team members should know how to recognize the emotional states of their colleagues and empathetically act.
5.8 To improve productivity in software development processes, team members should build relationships based on both trust and respect.

6. Collaboration
6.1 To improve productivity in software development processes, team members should collaboratively work to achieve the project's goals.
6.2 There should be trust among team members for the performance of their duties and protection of common interests, which contributes to improving productivity in software development processes.
6.3 To improve productivity in software development processes, team members should be willing to help, endorse, and support their colleagues.
6.4 To improve productivity in software development processes, team members should share their knowledge, information, and experience with their colleagues.

7. Team Cohesion
7.1 To improve productivity in software development processes, team members need to work at similar rates.
7.2 Importantly, members feel identified with the team participating in an autonomous and motivated manner, which helps to improve productivity in software development processes.
7.3 To improve productivity in software development processes, team members should put their best skills at the service of the project objectives.

7.4 To improve productivity in software development processes, team members should enjoy performing tasks with their colleagues.
7.5 The activities should be executed in a timely manner and with the participation of everyone involved to improve productivity in software development processes.
7.6 To improve productivity in software development teams, team members should know what each of their colleagues does.

8. Empathy and Interpersonal Relationships
8.1 To improve productivity in software development teams, team members should participate in social activities both within and outside the work environment.
8.2 To improve productivity in software development teams, team members may have little or no relationship with each other as long as they do their job.
8.3 To improve the productivity of the software development teams, team members should recognize that failing to perform their tasks may affect the performance of their colleagues and the team as a whole.
8.4 To improve productivity in software development teams, team members should receive training in interpersonal relationships, assertive management of emotions, teamwork, and quality.
8.5 To improve productivity in software development teams, there should be good personal relationships between team members.
8.6 To improve productivity in software development teams, members of the team should be able to take the place of the other when you need help and collaborate to meet your need.
8.7 To improve productivity in software development teams, each member should participate in workplace activities.
8.8 To improve productivity in software development teams, the members should promote a pleasant work environment.
8.9 To improve productivity in software development teams, team members should respect the agreed-upon coexistence agreements.

9. Leadership
9.1 To improve productivity in software development teams, any team member may have qualities to lead activities within the project.
9.2 To improve the productivity of the software development teams, team members should feel that they can offer solutions to project problems.
9.3 The leader should promote positive attitudes and build trust among software development team members to improve productivity.
9.4 To improve productivity in software development teams, work decisions should be made in group discussions and not unilaterally.
9.5 The leader should coordinate and guide the activities of the software development team toward project objectives and goals to improve productivity.
9.6 To improve productivity, the leader should be equitable in his/her treatment and demand toward software development team members.
9.7 To improve productivity in software development teams, the leader should be concerned with both project results and human relations.

10. Innovation
10.1 To secure process innovation and to improve productivity in software development teams, team members should consider customer's suggestions, complaints, and/or claims.
10.2 Company policies should promote the incorporation of innovation in projects to improve productivity in software development teams.
10.3 To improve productivity in software development teams, using solutions that have not been satisfactorily tested should be avoided.
10.4 To improve productivity in software development teams, their members should be able to take on new challenges and develop various skills.
10.5 The members of the software development team should support and be receptive to new ideas for improving productivity.
10.6 To improve productivity in software development teams, the leader should encourage members to place their own ideas into practice and identify new ways to deal with problems.

11. Autonomy
11.1 To improve productivity in software development teams, members should be empowered to make project decisions and the way in which they work.
11.2 To improve productivity in software development teams, their members can organize themselves to set and meet their goals.
11.3 To improve productivity in software development teams, team members should make decisions about the methods, techniques, and strategies to perform the tasks.
11.4 To improve productivity in software development teams, team members should trust their abilities to execute the tasks that have been assigned to them.
11.5 To improve productivity in software development teams, team members should be able to take corrective measures on their own initiative.

12. Capabilities and experiences in the software development process
12.1 To improve productivity in software development teams, team members should remain updated on the best tools and task execution practices.
12.2 To improve productivity in software development teams, team members should be knowledgeable on the subject matter or have worked in similar contexts.
12.3 To improve productivity in software development teams, team members should have knowledge or experience in the tools and programing languages used in the project.
12.4 To improve productivity in software development teams, team members should have knowledge or experience in the analysis, design, construction, or implementation of software.
12.5 To improve productivity in software development teams, team members should exhibit logical reasoning and systemic thinking skills.
12.6 To improve productivity in software development teams, team members should be able to implement efficient solutions to meet project requirements.
12.7 To improve productivity in software development teams, team members should have knowledge or experience in the analysis, design, construction, or implementation of software.

13. Capabilities and experiences in software project management
13.1 To improve the productivity of the software development teams, team members should adequately manage deadlines.
13.2 To improve productivity, software development team members should be able to plan, execute, or control project activities.
13.3 To improve productivity, software development team members should have knowledge or experience in project management tools and techniques.
13.4 To improve the productivity of the software development teams, team members should have knowledge or experience in the use of metrics used for project monitoring.

Additional Data

- Please specify how many people are included in your team, including yourself
- Please specify the main activity of the company you work for
- Please specify the legal nature of the company you work for
- Please specify the number of employees in the company you work for
- Please specify the number of projects completed last year in the company you work for
- Please specify the certifications achieved by the company you work for
- Please specify your role within the company you work for
- Please specify the type of software development methodology used by the company you work for
- Please list the software development methodologies used by the company you work for

6 Conclusions and Future Work

This study discusses the development and evaluation of an instrument used to measure perceptions on the SHFs that exert influence on the productivity of the software development team. For this purpose, the authors designed a 79-item survey covering 13 SHFs. Then, the instrument developed was assessed in terms of validity and reliability using the CVC and Cronbach's alpha coefficient. Moreover, the first version of the document was applied to a pilot sample.

The results from this instrument assessment revealed an average CVC of 0.86 and a Cronbach's alpha coefficient of 0.958, which indicates that the instrument is both valid and reliable.

Overall, this study improved the comprehension of the significance of the SHFs that affect software development productivity because, as confirmed by the corresponding content validity results, all items included in the instrument reflect a specific subject matter domain.

This instrument can be used by development teams that may wish to identify and rank SHFs that affect their productivity. Software development organizations can use these instruments to measure perceptions on the SHFs that may affect team productivity and implement improvements based on their results.

As for future work, the instrument designed will be applied to software development team members in Colombia and other Latin American countries. For this purpose, we plan to make the survey available for up to two months or until they obtain enough responses for a representative sample. Based on the results obtained, the authors expect to identify and rank the SHF identified as most significant by our subject matter experts and propose gamification strategies aimed at improving productivity of software development teams. According to [García-Mireles, 19], gamification has a positive impact in terms of improving productivity; it improves the quality of the software product and increases the developer's performance.

Additionally, the study of SHFs can be expanded by deepening factors related to user experience and customer satisfaction.

Acknowledgements

This study is part of the "Classification and Influence of Social and Human Factors on Software Development Team Productivity in Small and Medium IT Businesses" project, financed by La Universidad de Medellín (Colombia), El Centro de Investigaciones en Matemáticas (Zacatecas-México) and La Universidad de Guadalajara (México). We express our gratitude to all the funding institutions.

In addition, we also thank the SME reviewers Bell Manrique-Losada, María Clara Gómez Alvarez, Vianca Vega, Belkis Bosetti Carnevali and Liliana Paola Muñoz Gómez, who assessed the items to calculate the content validity of the instrument and the participants in the pilot study.

References

[Aiken, 96] Aiken, L.: Test psicológicos y evaluación, Editorial Prentice Hall Iberoamericana / México D.F (1996)

[Canedo, 19] Canedo, E. D., Santos, G. A.: Factors Affecting Software Development Productivity: An empirical study, In Proc. XXXIII Brazilian Symposium on Software Engineering, September 2019, 307-316.

[Capretz, 14] Capretz, L. F.: Bringing the human factor to software engineering, IEEE Software, 31, 2, 104–104, 2014.

[Consuegra, 10] Consuegra, N.: Diccionario de psicologia-Ecoe Ediciones, 2010.

[De Barros, 10] De Barros Sampaio, S. C., Barros E. A., De Aquino G.S., Carlos e Silva, De Lemos S. R.: A review of productivity factors and strategies on software development, In Proc. 5th Int Conf Softw Eng Adv ICSEA, 2010, 196–204.

[Dutra, 15] Dutra, A. C., Prikladnicki, R., França, C.: What do we know about high performance teams in software engineering? Results from a systematic literature review, In Proc. 41st Euromicro Conference on Software Engineering and Advanced Applications, August 2015, 183-190.

[Fatema, 18] Fatema, I., Sakib, K.: Factors influencing productivity of agile software development teamwork: A qualitative system dynamics approach, In Proc. 24th Asia-Pacific Software Engineering Conference (APSEC), December 2017, 737-742, IEEE.

[Flórez, 15] Flórez, I., Fajardo., L.: Validación de un instrumento para la caracterización de la comunicación no verbal intencional de niños con habilidades diversas (sordoceguera y pluridefiencia), Corporación Universitaria Iberoamericana / Bogotá (2015)

[García-Mireles, 19] García-Mireles, G. A., Morales-Trujillo, M. E.: Gamification in Software Engineering: A Tertiary Study, In Proc. International Conference on Software Process Improvement, October 2019, 116-128. Springer, Cham.

[Hernández, 14] Hernández Sampieri, R., Fernández, C., Baptista, M.P.: Metodología de la investigación, McGraw-Hill /Interamericana editores, S.A / México, D.F (2014)

[Hernández López, 14] Hernández López, A.: Medidas de productividad en los proyectos de desarrollo de software: una aproximación por puestos de trabajo, 2014.

[Iqbal, 19] Iqbal, J., Omar, M., Yasin, A.: An empirical analysis of the effect of agile teams on software productivity, In Proc. 2nd International Conference on Computing, Mathematics and Engineering Technologies (iCoMET), January 2019, 1-8, IEEE.

[Judge, 10] Judge, T. A., Piccolo, R. F., Podsakoff, N. P., Shaw, J. C., Rich, B. L.: The relationship between pay and job satisfaction: A meta-analysis of the literature, Journal of Vocational Behavior, 77, 2, 157-167, 2010.

[Kitchenham, 08] Kitchenham, B.A., Pfleeger, S.L.: Personal Opinion Surveys, In: Shull F., Singer J., Sjøberg D.I.K. (eds) Guide to Advanced Empirical Software Engineering, Springer, London 2008.

[Machuca-Villegas, 19] Machuca-Villegas, L., Gasca-Hurtado, G. P.: Towards a Social and Human Factor Classification Related to Productivity in Software Development Teams, In Proc. International Conference on Software Process Improvement, October 2019, 36-50, Springer, Cham.

[Matturro, 19] Matturro, G., Raschetti, F., Fontán, C.: A Systematic Mapping Study on Soft Skills in Software Engineering, J.UCS (Journal for Universal Computer Science), 25,1, 16-41, 2019.

[Muñoz, 14] Muñoz. M., Mejia, J., Duron, B., Valtierra, C.: Software Process Improvement from a Human Perspective. In: Rocha Á., Correia A., Tan F., Stroetmann K. (eds) New Perspectives in Information Systems and Technologies, Volume 1. Advances in Intelligent Systems and Computing, vol 275. 2014, Springer, Cham.

[Murphy-Hill, 19] Murphy-Hill, E., Jaspan, C., Sadowski, C., Shepherd, D., Phillips, M., Winter, C., …Jorde, M.: What Predicts Software Developers' Productivity?, IEEE Transactions on Software Engineering, 2019.

[Oliveira, 18] Oliveira, E., Conte, T., Cristo, M., Valentim, N.: Influence factors in software productivity - a tertiary literature review, International Journal of Software Engineering and Knowledge Engineering, 28(11n12), 1795-1810, 2018.

[Oviedo, 05] Oviedo, H., Campo-Arias, A.: Aproximación al uso del coeficiente Alfa de Cronbach, Rev. Colomb. Psiquiatr, 34, 4, 572-580, 2005.

[Pedrosa, 13] Pedrosa, I., Suárez-Álvarez, J., García-Cueto, E.: Evidencias sobre la validez de contenido: avances teóricos y métodos para su estimación, Acción psicológica, 10,2, 3-18, 2013.

[Pirzadeh, 10] Pirzadeh, L.: Human Factors in Software Development: A Systematic Literature Review, Master Sci Thesis Comput Sci Eng, 368, 2010.

[PMI, 13] Project Management Institute, IEEE Computer Society, Software Extension to the PMBOK® Guide Fifth Edition, Fifth. Project Management Institute, Inc. 2013

[Quero, 10] Quero, M.: Confiabilidad y coeficiente Alpha de Cronbach, Telos, 12, 2, 248-252, 2010

[Tomasello, 10] Tomasello, M.: ¿Por qué cooperamos?, Katz Editores, 2010.

[Trendowicz, 09] Trendowicz, A., Münch, J.: Factors Influencing Software Development Productivity-State-of-the-Art and Industrial Experiences, Adv Comput, 77, 185–241, 2009.

[Vallejo-Nágera, 02] Vallejo-Nágera, J.: Guía Práctica de Psicología, Ediciones Temas de Hoy. 2002

[Vivas, 09] Vivas, P., Rojas, J., Torras., M.: Dinámica de Grupos, Fundación para la Universidad Oberta de Catalunya FUOC, 2009.

[Wagner, 08] Wagner, S., Ruhe, M.: A Systematic Review of Productivity Factors in Software Development, Softw Product Anal Cost Estim, 1–6, 2008.

[Zurita, 16] Zurita, G., Baloian, N., Pino, J., Boghosian, M.: Introducing a Collaborative Tool Supporting a Learning Activity Involving Creativity with Rotation of Group Members, J.UCS (Journal for Universal Computer Science), 22 (10), pp. 1360-1379, 2016.