Measurement of the Non-Technical Skills of Software Professionals: An Empirical Investigation

Lisa L. Bender[1], Gursimran S. Walia[2], Fabian Fagerholm[3], Max Pagels[4], Kendall E. Nygard[5], Jürgen Münch[6]
Department of Computer Science
University of Houston - Clear Lake[1]; North Dakota State University[2, 5]; University of Helsinki[3, 4, 6]
lbender@uhcl.edu[1]; {gursimran.walia[2], Kendall.Nygard[5]}@ndsu.edu; {fabian.fagerholm[3], max.pagels[4], juergen.muench[6]}@cs.helsinki.fi

***Abstract*— Software development managers recognize that project teams need to be developed and managed. Although technical skills are necessary, non-technical (NT) skills are equally necessary for project success. There are several tools that assist in measuring the effectiveness of the technical skills that teams use to perform projects, but there are no proven tools to measure the NT skills of software developers. Behavioral markers (BM), observable behaviors that have positive or negative impacts on individual or team performance) are beginning to be successfully used by airline and medical industries to assist managers in assessing NT skills of project teams and individuals. The purpose of this research is to develop and validate a NT skills taxonomy for software developers. This paper presents an empirical investigation to develop and validate a NT skills taxonomy which was in turn used to construct a BM system tool for said developers and software development teams.**

Keywords-Non-technical Skills; behavior marker; performance.

I. INTRODUCTION

The software development process is a team activity and the success of a software project depends on the effective performance of the software project team. West [1] notes that working in a team is not automatically beneficial. Simply bringing people together does not ensure that they will function as an effective team. The Project Management Institute recognizes the need to develop project teams. The most recent PMBOK Guide [2] states "*teamwork is a critical factor for project success, and is one of the primary responsibilities of the project manager*". PMBOK also acknowledges that non-technical (NT) skills, in comparison to technical skills, are equally important for project success and team development. Multiple authors agree that the NT skills are critical to project success [3, 4]. Other authors assert that NT skills can have the largest impact on software development [5, 6].

The cognitive and interpersonal skills which underpin software professionals and technical proficiency are being recognized as requirements for a competent software developer [7]. One major factor that is driving the demand for NT skills is the requirement for an agile workforce to support agile organizations [8]. Agile software development methodology is based on incremental and interactive development. This development is carried out through the collaboration between self-organizing, cross-functional teams. Agile teams depend greatly on efficient communication, taking responsibility, initiative, time management, diplomacy, and leadership.

While the performance of individuals is very important to creating an effective team, there are no established guidelines for measuring team effectiveness. Different authors have identified different criteria for assessing team effectiveness [9, 10]. These criteria generally include measurements of task performance as well as the interpersonal skills of the team members, which include attitudes and behaviors. While there is extensive literature with respect to different ways to measure task performance (e.g., lines of code) for software development [11], little research has been performed on measuring NT skills. On that note, the aviation and health care industries have already recognized the importance of NT skills to the success of their teams, and have been using behavioral marker systems to structure individual and team assessments of these NT skills. We believe that software teams can also draw upon these models to improve teamwork in software development.

As software project development managers and educators, this is one of the factors that motivated our research. How can managers objectively measure the NT skills of their employees to determine if their NT skills are adequate or if they need improvement? How would feedback be provided to the team members so that they could improve their performance? The research reported here is an attempt to answer these kinds of questions. Thus, the purpose of this research is to identify the NT skills required by effective software professionals, and to develop a behavioral marker system for evaluating these skills.

II. BACKGROUND

This section provides background on the NT skills (Section II.A) and the behavior marker systems (Section II.B).

A. Non Technical (NT) Skills

NT skills are the cognitive, social, and personal resource skills that complement technical skills and contribute to overall task performance [12]. Classic examples of NT skills are leadership, patience, cooperation, communication, decision making, conflict management, stress and workload management, attention to detail, empathy, and confidence. In short, NT skills cover both the social and cognitive side of a person. In a survey released in 2013 by the Association of American Colleges and Universities [14], it was found that employers feel that NT skills, both cognitive and interpersonal, are more important than a particular major. Even professional organizations such as Professional Engineering Competence (UKSPEC), IEEE Computer Society etc. state that professional engineers have an obligation to possess NT skills [15].

Universities and colleges have strived to create curricula to prepare students to be Software Engineers. Some researchers have defined competencies (both technical and NT) for Undergraduate Software Engineering students, however these

do not encompass all of the competencies, such as many necessary interpersonal skills, needed for a Software Engineering professional [16, 17]. Other researchers have developed expert profiles (tools that communicate the technical and NT skills required) for engineering professionals that include input from both academia and industry; however, they do not define specific competencies required for a Software Engineer [18]. Educators summarize important course knowledge and skills that the student's should develop in course syllabi. Employers list minimum requirements for new hires in job advertisements. With so many different sources and kinds of information available, it is difficult to synthesize what competencies and in particular, NT skills, are required in the software profession. Therefore, this research attempts to develop a NT skills taxonomy for software professionals.

B. Behavior Markers(BM) and Behavior Marker System

Behavioral markers (BM) are defined [19] as "*observable, non-technical behaviors that contribute to superior or substandard performance within a work environment*". They are derived by analyzing data regarding performance that contributes to successful and unsuccessful outcomes. These markers are often structured into categories (e.g. communication, situational awareness, and decision making). The overall purpose of a BM system is to provide a method to assess team and or individual behaviors using markers. In general, BM systems have a taxonomy or listing of NT skills that are associated with effective job performance. This listing is combined with a rating scale to allow the skills (which are demonstrated through behaviors) to be assessed by trained observers. These BM systems are part of an observation-based method to capture and assess individual and team performance on data rather than on gut feelings. Observers use the BM tool, designed in the form of a structured list of skills, to rate skill and behavior performance. This allows an individual's or team's skills to be rated in their real context. BM systems can provide feedback on performance to individual/ teams as well as a common language for discussing and teaching NT skills.

BM systems have demonstrated value for assessing NT skills, for providing feedback on these skills, for improving training programs for NT skills, and in the use of building databases to identify norms and prioritize training needs. Given the prevalence and success of BM systems, we believe that they can be effective at improving NT skills in software development teams. However, a BM system developed for one domain cannot simply be transferred to another domain. It is important to recognize that BM systems need to be specific to the domain and culture. O'Conner et al. [20] noted that the Human Factors Analysis and Classification System developed for aviation was not appropriate for assessing the NT skills for U.S. Navy divers.

A brief description of the domains in which BM systems have been successfully used (i.e., airline, medicine) follows:

1) Airline Industry: The first BM system developed for the airline industry had two primary purposes: to evaluate the effectiveness of crew resource management (CRM) by measuring observable behaviors and to aid the development of future CRM programs [21]. The Line Operation Safety Audit (LOSA) is a very successful BM system, and many of the BM systems in other industries were adapted from this audit tool. It focuses on interpersonal communication, leadership, and decision making in the cockpit. Trained observers (pilots and human factors experts) ride along in the cockpit and observe the flight crews during normal flight operations. They score the behaviors of the crew using LOSA. This tool has been very successful in measuring the strengths and weakness of the flight crews' interpersonal skills and is endorsed by the International Civil Aviation Organization [22].

2) Medical Industry: To help improve teamwork in healthcare, BM systems are being adopted. Two predominate tools to date include the Anesthetists' NT Skills (ANTS) System and the Observational Teamwork Assessment of Surgery (OTAS). ANTS provides a taxonomy for structured observations of anesthetists [23]. This system has proven very useful in assessing the NT skills of anesthetists in simulation training and has provided important performance feedback for the individuals. OTAS was developed to evaluate the technical and interpersonal skills in surgery teams [24]. Empirical studies have shown that the underlying cause of many adverse events in surgery were the result of poor communication, coordination, and other aspects of teamwork rather than technical failures. OTAS has been found to be a valid measure the technical and NT performance of surgical teams.

Our goal is to develop a BM system that can improve software professional team member performance by providing feedback in the form of an objective and documented assessment of the NT skills of the team members.

III. RESEARCH APPROACH

The research approach included activities with a major focus on a) developing a taxonomy of NT skills of software professionals, b) validating and refining the taxonomy, and c) developing a BM system for software developers. More details on the study design and results appear here.

As a first step, a systematic literature review was performed to identify and analyze the NT skills of software professionals. A *systematic literature review* is a systematic search process that focuses on particular research question and provides an exhaustive summary of literature relevant to that question. By performing a systematic review, researchers can be more confident that they have found background information relevant to their study. The more common *ad hoc* approach does not provide this same level of assurance [25]. Next, a focus survey was developed using the process recommended by Davis et al. [26]. The focus groups consisted of employers, SE and CS industrial professionals and instructors. The software professional NT skills profile survey used the NT skills information gathered from the systematic literature review and was developed with the assistance of a focus group. The results of the survey were used to develop the software professional NT skills taxonomy. Finally, a review of related BM system literature was carried out in order to develop a BM system for software developers, and a BM rating tool was created.

IV. RESEARCH DESIGN

This section describes the research design. Section IV.A describes the systematic review protocol. Section IV.B presents the focus group design, survey procedure, and the process of developing the BM system and the behavior examples.

A. Systematic Literature Review

In accordance with systematic review guidelines [27], the review protocol was developed that specified the questions to be addressed, the databases to be searched and the methods to be used to identify, assemble, and assess the evidence. To properly focus the review, a set of research questions (RQ's) were developed. With the underlying goal to develop a software professional NT skills profile, the high-level question addressed by this review was:

> *"What are the NT skills required of software professionals performing well in their field and how can we discover what NT skills are valued by employers?"*

This high-level question was then decomposed into the more specific RQ's. ***RQ#1*** identified the existing empirical studies reported on desired competencies in software professionals. ***RQ#2*** focused on efforts, methods or tools that are used to identify or can be used to identify a comprehensive list of NT skills. If any of these methods or tools has been implemented, we also analyzed their level of success and lessons learned. ***RQ#3*** combined the results of the first two RQ's in an attempt to develop a software professional NT skills profile. Details on the review protocol (sources searched, search execution, inclusion and exclusion criteria, quality assessment, data extraction) can be referred to in a report [28].

B. Focus Group

After an initial list of NT skills was identified from the literature review, the skills were clustered into major categories (details in Section V). Synthesis of the literature review of software developer NT skills was then incorporated into a survey. This survey was sent to a diverse group of individuals from SE academia and industry for review. We employed two online surveys to assist in gathering this input. The focus group members who participated in these surveys were located in three different states, thus using a survey questionnaire was an efficient way of collecting the NT skills input. Both surveys used a cross-sectional survey design in which we gathered information about the NT skills important to a professional software developer at a specific point in time. First, the initial list of NT skills was compiled and then the first electronic survey was created. The first survey used an initial draft of NT skills gathered from the literature review as a basic guideline and then gathered NT skills priorities, missing NT skills, description clarifications, and comments to produce a more robust NT skills inventory. Once this survey was complete, an updated NT skills profile was created. The purpose of the second survey was to gather examples of good and poor behaviors for the top rated NT skills from the first survey. The details of the survey design appears in the subsections

1) Survey Methodology

The surveys intend to evaluate the NT skills of a software developer performing well in professional practice, and identify the observable actions of the NT skills of a software developer.

Survey Participants: A group of 20 individuals (SE professors, and industry managers representing both publically and privately held companies from small to large software development departments) was asked to provide input on the list of NT skills. Because cultural differences have been found to have a significant impact on individuals [29], we decided to only seek input from educators and employers along the I-29 corridor of Minnesota, North Dakota, and South Dakota. Three universities (Dakota State University, North Dakota State University, University of Minnesota Crookston), along the I-29 corridor, were identified to have programs that would produce graduates suitable to being employed as Software Engineers and individuals were selected from each university. Each of the industry collaborators were selected because the companies they were associated with were located along the I-29 corridor; they all employed many new graduates that work in SE and software development related jobs; and they all have well developed human resource departments with sufficient resources to have created comprehensive competency expectations for company's employees and thus would have clearly defined expectations. The industry collaborators included managers of software professionals from each of the companies.

Survey Procedure: The survey included following 3 steps.

Focus Group Survey #1: The focus group was emailed an electronic survey and asked to rank the importance of each NT skill to software professionals. The skills were listed in the categories (discussed in the results section). The survey also included the descriptions for each skill (can be found in [28]). The ranking that we asked the focus group to produce provides prioritization of NT skills that most reflect expert activities. The focus group was also asked to provide inputs (suggested revisions to the NT skills, clarifications of the NT skill descriptions, missing elements, assess quality, and any further comments) to the NT skills. The quality of the NT skills was assessed per the guidelines provided by Davis and Beyerlein [18] by asking the focus group to provide feedback. The focus group helped create a more robust NT skill list.

Compile High Priority List of NT Skills: The results of this first survey were compiled into an improved NT skills taxonomy. Some competencies were re-grouped, and the list trimmed of the competencies that did not meet the quality standards. This more robust NT skill list only include the most highly prioritized NT skills, which was intended to make it easier for the focus group to complete the second survey.

Focus Group Survey #2: The second electronic survey sent to the focus group posed open-ended questions. The participants were asked to provide examples of observable actions that indicate good performance and behavior of each NT skill as well as examples of observable actions that indicate poor performance and behavior of each NT skill. They were asked to provide as many examples as they wished for each skill. The examples of good and poor behaviors were collected. Based on the inputs from the second survey, we developed a behavior-based software engineer NT skills taxonomy and used it as the basis for the behavioral marker system. The resultant examples of good and poor behavior for a subset of skills appear in the results section

V. Research Results

This section provides results and findings organized around the activities described in Section IV.

A. Non Technical (NT) Skills based on Literature Review

After an initial list of NT skills was identified from the literature review, we clustered the skills into four major categories: *communication*, *interpersonal*, *problem solving*, and *work ethic*. These categories are illustrated in Fig. 1 and were reviewed by the involved researchers. We also had performed research to find meaningful descriptions for each skill. In many instances, it was felt that an identified skill overlapped with another NT skill, thus a list of synonyms was created to help provide clarity. Details of this information for one of the NT skills category (i.e., "*Communication*") can be seen in Table I and more details appear in [28].

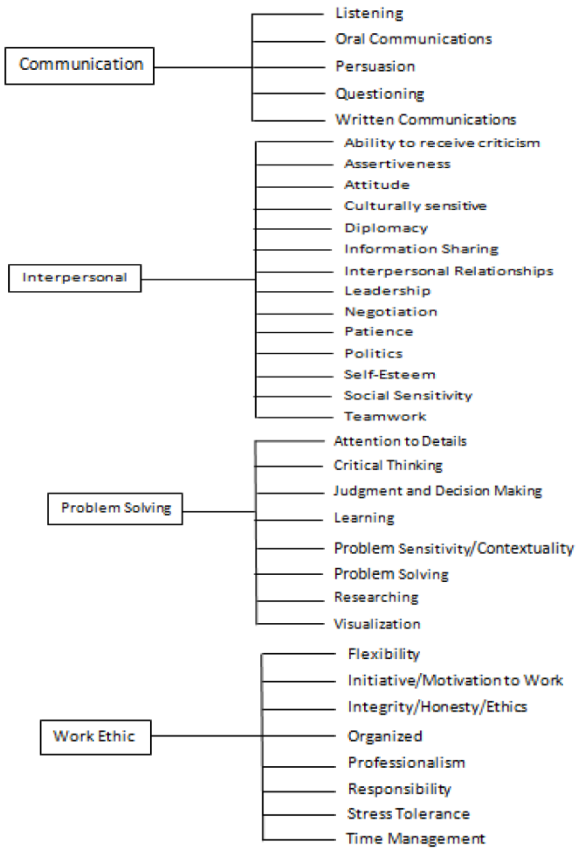


Fig. 1: Desired NT skills of Software Professionals

B. Results from the Focus Group# 1 and # 2

As mentioned earlier, individuals during the focus group# 1 ranked the importance of each NT skill to software professionals. The skills were listed in the categories and in the same order (as seen for one of the categories in Table I). The survey also included the descriptions listed (as in Table I).

We looked at different ways to analyze the Likert data from the first survey. One method was to look at the NT skills that received the highest percentage of essential ratings. In that vein, the list of the top skills and the percentage of respondents who thought this skill was most essential (rank = 1) can be found in Table II. A second, and very common method considered, often used in analyzing the Likert data, was to simply summarize the Likert values for each NT skill. Based on the summaries, the most essential NT skills, in order, are 1) *teamwork*, 2) *attitude*, 3) *listening*, 4) *initiative/motivation to work*, 5) *critical thinking*, 6) *problem solving*, 7) *attention to detail*, 8) *flexibility*, 9) integrity/honesty/ethics, 10) *time management*, 11) *interpersonal relationships*, 12) *oral communications*, 13) *questioning*, 14) *learning*, 15) *leadership*, and 16) *responsibility*. These two lists were very similar; however, after discussing these results with focus group, it was decided to combine the two lists to comprise the second draft of the NT skills that should be considered in the focus group#2.

TABLE II. Essential NT Skills Ratings

Non-Technical Skill	% of respondents who rated skill as essential
Teamwork	**91%**
Initiative/Motivation to work	**73%**
Listening	**73%**
Attitude	**64%**
Critical Thinking	**64%**
Oral Communications	**64%**
Leadership	**64%**
Problem Solving	**64%**
Attention to Detail	**55%**
Flexibility	**55%**
Integrity/Honesty/Ethics	**55%**
Time Management	**55%**

The output of focus group # 2 were examples of observable actions that indicated good performance and behavior of each NT skill as well as examples of observable actions that indicate poor performance and behavior of each NT skill. They were asked to provide as many examples as they wished for each skill. The skills under consideration were: *teamwork*, *initiative/motivation to work*, *listening*, *attitude*, *critical thinking*, *oral communication*, *leadership*, *problem solving*, *attention to detail*, *flexibility*, *integrity/honesty/ethics*, *time*

TABLE I. Detailed NT Skill of SW Devel's: Communication

Category	Skill	Synonyms	Description
Communication	Listening	Listen and Understand	Paying attention to and concentrating on what is being said, and asking questions that refine points about which one is uncertain.
	Oral Communications	Communication; Verbal Communication; Communication Skills; Presentation Skills	Presenting your ideas in a manner easily understood by your audience, both in group meetings and person to person. Reinforcing the message to others through gestures and facial expressions.
	Persuasion	Change Agent; Salesman; Influence; Influence and Control; Ability to Influence; Sales; Managing Power/Expectations	Promoting the system you advocate; persuading others to accept your viewpoint.
	Questioning	Interviewing	Asking the right questions in order to obtain the information needed.
	Written Communications		Preparing written documents that accurately communicate ideas in a manner that is easily understood by intended readers.

Listening	
Paying attention to and concentrating on what is being said, repeating points of discussions to ensure mutual understanding and asking questions that refine points about which one is uncertain.	
Examples of Good Behaviors	**Examples of Bad Behaviors**
Restates, rephrase (paraphrase), or summarize the message to provide feedback if the message was clear and understood. Questions of confirm understanding of the message (e.g. Listening carefully to a teammates presentation, asking questions, and providing constructive feedback in a supportive way).	Passively participating and not paying full attention. (e.g. Looking around the room, checking emails, or other activities on a laptop or phone that show they are not paying full attention to the speaker).
Pauses before restatement or question to show that the message was carefully considered.	Creates distractions while someone else is talking such as whispering to others while someone else is speaking.
Asks clarifying questions. E.g. asks questions to ensure design met the requirements.	Cutting in to the conversation, not letting others complete thoughts. Talking at the same time as someone else.
Lets others talk; does not interrupt.	Combative listening mode - not receptive to other points of view and immediately want to promote their point of view.
Looks at the person speaking. Maintains good eye contact with the speaker.	Exhibits poor body language: crossing arms and legs, stands too close or too far away to the speaker and causes the speaker noticeable discomfort.
Exhibits receptive body language such as leaning forward, nodding or shaking head, etc. (note that nonverbal clues vary from culture to culture. e.g. in some cultures vertically nodding head usually means agreement, in other cultures shaking head means agreement)	

Fig. 2: Example behavioral markers for listening

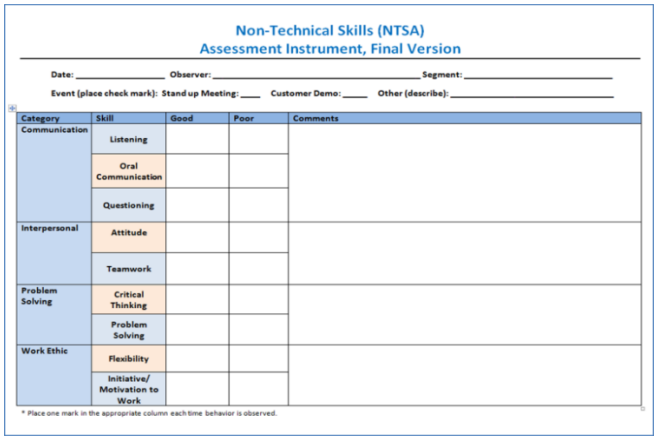
management, and *questioning*. A total of 408 examples of good and poor behaviors were collected.

These examples of good and poor behavior provided by the focus group were analyzed and, using an adaptation of the consensual qualitative methodology [30], reviewed and redundant examples were eliminated. The researchers then reviewed the remaining behaviors and evaluated their clarity and how observable they were. Some behavioral examples, such as "*being a good team player*" and "*body language and persona emitting that you do not enjoy your work*", were too ambiguous and removed. It was also felt that the "*Leadership*" skill did not have enough observable behaviors that would be able to be clearly identified, so that NT skill was removed. The result of the second survey was a behavior-based software engineer NT skills taxonomy. Fig. 2 shows the resultant examples of good and poor behavior for the "*Listening*" skill. The same process was used to create examples of good and poor behavior for each NT skill, and can be referred in [28].

C. Creating a Behavior Marker System

The literature review on existing behavioral marker systems showed that there are no BM systems currently being used in the software industry, but did identify existing BM systems in aviation, health care, nuclear power, rail transport and maritime transport. Each system's structure was examined. The Communication and Teamwork Skills (CATS) Assessment showed the most potential for use in software development because it was devised to measure communication and teamwork providers in a variety of medical environments rather than focusing on a specialization. It also provided an easy to use scoring method [31]. Our results have shown that the NT skills important to good software development team practice include *communication*, *interpersonal*, *problem solving*, and *work ethic*. Our NT skills taxonomy also includes examples of good and poor behaviors for each skill.

Using the results, we developed a BM marker audit tool (see Fig. 3) that we refer to as the NT Skill Assessment for Software Developers (NTSA). The NTSA is designed to be used by an observer (i.e. manager, team leader, coach) during routine team interactions or meetings. It is intended that each time a behavior is observed, a mark is placed in the appropriate column by clicking on the column: observed and good,


Non-Technical Skills (NTSA)
Assessment Instrument, Final Version

Date: _____ Observer: _____ Segment: _____

Event (place check mark): Stand up Meeting: ___ Customer Demo: ___ Other (describe): _____

Category	Skill	Good	Poor	Comments
Communication	Listening			
	Oral Communication			
	Questioning			
Interpersonal	Attitude			
	Teamwork			
Problem Solving	Critical Thinking			
	Problem Solving			
Work Ethic	Flexibility			
	Initiative/ Motivation to Work			

* Place one mark in the appropriate column each time behavior is observed.

Fig. 3: NT skills assessment instrument

variation in quality or expected but not observed. Observations can be clarified by placing explanations in the comments section. The observer can see skill definitions and examples of good and poor behavior for a particular behavioral marker by viewing the second page. A manager is allowed to list as many or as few skills as desired in the behavioral marker column. The reason for this flexibility is that different organizations and different managers may wish to focus on a certain subset of NT skills. The observer will score the behaviors base on how well the behavior meets the behavioral examples and its definition. Our NTSA behavioral marker tool will be very usable for practitioners. Empirical validation of our NTSA tool is under progress and our aim is to provide a tool that requires minimal training to use.

VI. DISCUSSION OF RESULTS

An underlying goal of this research was to develop a useful taxonomy of the NT skills required for software engineers. NT skills are important to the success of projects, but a complete and relevant list approved by both academics and industry has never been developed until now. To accomplish this goal, this research used information found during a literature review and further refined by a focus group of experts in the field to develop NT skills taxonomy. This taxonomy can be used by software developers, educators, and industry to identify the NT skills required by software engineers and software developers that are necessary to have in order to build high-quality

software. Based on evidence gathered from the systematic literature review and the results of two rounds of focus group of experts in the software industry, an NT skills taxonomy was created and validated.

The management of software developers' NT skills is particularly important to today's teams because more and more industries are using agile methodologies which rely less on documentation and more on informal interactions between people. Professional software organizations feel that these skills need to be tracked and feedback provided so that software development team project members can improve. However, there are currently no tools available to assist with this task. To accomplish this goal, we developed a behavioral marker system for software developers based on the NT skills taxonomy. Individuals responsible for measurement and development of the NT skills of their software development teams can use the marker system as a tool across projects to determine areas of strength and areas that need improvement, providing objective feedback to teams and managers. A tool such as the NTSA provides a mechanism to improve a team and by extension the software that they produce.

VII. Conclusion And Future Work

This work will benefit researchers, educators, and industry professionals in identifying relevant NT skills to research, and to provide focus on improving the NT skills in software professionals that are so important to software project success. ***For researchers***, this work can serve as a starting point for future research into improving the relevant NT skills of software professionals. ***For industry***, this work provides a method for managers to measure and manage the NT skills of their software professionals. Industry can use the NT skills taxonomy to identify the NT skills they feel are most relevant to their organizations. The NTSA provides a common language with which to understand and communicate about NT skills important to software professionals. Our future work will include performing studies to refine the NTSA tool and ultimately validate it for the eight NT skills identified as most important to software developers. We expect that future work can further deepen the understanding of which skills are important specifically for software development in contrast to skills that are relevant for teamwork in general. Further validation is also needed in different cultural contexts and development domains. We have begun collaborating with other researchers to use the BM system to investigate industry and student teams in different development environments. Ultimately, the NT skills ratings should be used as independent variables in studies examining the impact of non-technical skill performance on project outcomes.

References

[1] M.A. West. Effective teamwork: practical lessons from organizational research. Malden, MA: Wiley-Blackwell, 2004, pp. 9-14.

[2] Project Management Institute. A Guide to the Project Management Body of Knowledge (PMBOK Guide). Newton Square, PA: Project Management Institute, 2008, pp. 215.

[3] S. Acuna, N. Juristo, and A.M. Moreno, "Emphasizing Human Capabilities in Software Development", IEEE Software, vol. 23, 2006, pp. 94-101.

[4] E. Amengual, and A. Mas, "Software Process Improvement through Teamwork Management," in Proceedings of the 8th International Conference on Product-Focused Software Process Improvement, 2007, pp. 108-117.

[5] A. Cockburn, and J. Highsmith, "Agile software development: The people factor", Computer, vol. 34, 2001, pp. 131-133.

[6] N. Gorla, and Y. Wah Lam, "Who Should Work With Whom?" Communications of the ACM, vol. 47 No. 6, pp. 79–82, Jun. 2004.

[7] Ahmed, F. Capretz, L.F., Bouktif, Salah, and Campbell, P., "Soft Skills and Software Development: A Reflection from Software Industry", International Journal of Information Processing and Management(IJIPM), vol. 4, number 3, May 2013, pp. 171-191.

[8] Abell, Angela, Information World Review; Dec 2002; 186; ABI/INFORM Complete pg. 56

[9] S.G. Cohen, and D.E. Bailey, "What Makes Teams Work: Group Effectiveness Research from the Shop Floor to the Executive Suite", Journal of Management, vol. 23, 1997, pp. 239-290.

[10] J.J. Jiang, J. Motwani, and S.T. Margulis, "IS team projects: IS professionals rate six criteria for assessing effectiveness", Team Performance Management, vol. 3, 1997, pp. 236-242.

[11] O. Hazzan and I. Hadar, "Why and how can human-related measures support software development processes?" The Journal of Systems and Software 81, 2008. Pp/ 1248-1252.

[12] R. Flin, P. O'Connor, and M. Crichton., "Safety at the sharp end: A guide to non-technical skills", 2008, Burlington, VT: Ashgate Publishing Company. Pg. 264

[13] D. Carnegie, How to Win Friends and Influence People. New York, NY: Pocket Books, 1998, pp. xvi.

[14] Higher Ed News, "Survey Finds Business Executives Aren't Focused on Majors They Hire" accessed Mar. 14, 2014,

[15] UKSPEC,"UK-SPEC UK Standard for Professional Engineering Competence," accessed Mar. 14, 2014, www.engc.org.uk/ecukdocuments/internet/document library/UK-SPEC third edition.pdf

[16] A. Begel and B. Simon, "Novice software developers, all over again," in Proceedings of the Fourth International Workshop on Computing Education Research (ICER08), 2008, pp. 3-14.

[17] M. Devlin and C. Phillips, "Assessing Competency in Undergraduate Software Engineering Teams", IEEE Engineering Education, Universidad Politecnica de Madrid, Apr. 2010, pp. 271-277.

[18] D.C. Davis, S.W. Beyerlein, and I.T. Davis, "Development and use of an engineer profile", in Proceedings American Society for Engineering Education Conf., American Society for Engineering Education, Jun. 2005 .

[19] B. F. Klampfer, R. L. Helmreich, B. Hausler, B. Sexton, G. Fletcher, P. Field, S. Staender, K. Lauche, P. Dieckmann, and A. Amacher. "Enhancing performance in high risk environments: Recommendations for the use of behavioral markers." Behavioral Markers Workshop, 2001, pp. 10.

[20] P. O'Connor and A. O'Dea, "The U.S. Navy's aviation safety program: a critical review," International Journal of Applied Aviation Studies, 2007, vol. 7, p. 312-328.

[21] B. F. Klampfer, R. L. Helmreich, B. Hausler, B. Sexton, G. Fletcher, P. Field, S. Staender, K. Lauche, P. Dieckmann, and A. Amacher. "Enhancing performance in high risk environments: Recommendations for the use of behavioral markers." Behavioral Markers Workshop, 2001, pp. 10.

[22] J.R. Klinect, P. Murray, A. Merritt, and R. Helmreich. "Line Operations Safety Audit (LOSA): Definition and operating characteristics," in Proceedings of the 12th International Symposium on Aviation Psychology, 2003, pp. 663-668.

[23] G. Fletcher, R. Flin, P. McGeorge, R. Glavin, N. Maran and R. Patey, "Development of a Prototype Behavioural marker System for Anaesthetists' Non-Technical Skills (ANTS)," Workpackage 5 Report, Version 1.1. (2003)

[24] N. Sevdalis, M. Lyons, A.N. Healey, S. Undre, A. Darzi, and C.A. Vincent. "Observational Teamwork Assessment for Surgery: Construct Validation with Expert Versus Novice Raters." Annals of Surgery, vol. 249, pp. 1047-1051, 2009.

[25] G.S. Walia and J.C.Carver, "A systematic literature review to identify and classify software requirement errors". Information and Software Technology, vol. 51, pp. 1087-1109, 2009.

[26] D.C. Davis, S.W. Beyerlein, and I.T. Davis, "Development and use of an engineer profile", in Proceedings American Society for Engineering Education Conf., American Society for Engineering Education, Jun. 2005 .

[27] H. Kandeel, K. Wahbe, Competncy models for human resource deveopment: case of Egyptian software industry. Managing Information Technology in a Global Environment. 2001 Information Resources Management Association International Conference. Idea Group Publishing. 2001, pp. 117-121

[28] L.L. Bender and G.S. Walia, "Measurement of Non-Technical Skills of Software Development Teams", Department of Computer Science, North Dakota State University, Fargo, ND, Tech. Rep. NDSU-CS-TR-14-001, Mar. 2014.

[29] G. Fletcher, R. Flin, P. McGeorge, R. Glavin, and R. Patey. "Anaesthetists' Non-Technical Skills (ANTS): evaluation of a behavioural marker system." British Journal of Anaesthesia, vol. 90, pp. 580-588, 2003.

[30] C.E.Hill, S. Know, B.J Thompson, E.N. Williams, S.A. Hess and N. Landany 2005. Consensual qualitative research: an update. Journal of Counseling Psychology 52 (2), 196-205.

[31] F. Robert, A. Abran, and P. Bourque, "A Technical Review of the Software Construction Knowledge Area in the SWEBOK Guide ," STEP 2002, pp. 9