Researching Human and Organizational Factors Impact for Decisions on Software Quality

Luis Fernández-Sanz[1], Josefa Gómez-Pérez[1], Teresa I. Diez-Folledo[1] and Sanjay Misra[2]
[1]*Dept. of Computer Science, University of Alcalá, Edificio Politécnico, Campus Universitario, Alcalá de Henares, Spain*
[2]*Covenant University, Ota Ogun, Nigeria*

Keywords: Software Quality, Quality Assurance (QA), Human Resources, Organizational Factors, Stakeholders, Quantitative Research, Qualitative Research, Decision Making Processes.

Abstract: Quality is an essential factor for European competitiveness as low price strategies based on low labour costs can be difficult to implement. Although software quality assurance has a long tradition, there is a lack of research on some practical aspects. In particular, the extended study of the influence of human and organizational factors (HOF) on the quality of software development, maintenance and management has been neglected. However, different studies have identified these as key factors in software projects with impact in terms of cost, quality and results measuring quantitative and qualitatively their impact. As part of the Iceberg project, funded under the Marie Curie IAPP EU-funded program, some relevant evidences of the influence of HOF on software quality has been reviewed and analysed to discuss the challenges in this area confirming the need of promoting deeper and wider research efforts.

1 INTRODUCTION

Software has become an essential asset for European companies, organizations and society. Software is an important part of ICT not only in information systems but also as an embedded component of many products or services. It is the distinctive part of products and services enabling them to be competitive or innovative. So software quality is strategic for all European stakeholders. A compilation of incidents by Peter Neumann ("Inside risks") from the 1980s to today shows there is room for improvement in software quality. For European industry, competitiveness in software depends on innovation and quality given that prices cannot be reduced without impacting the workforce. Therefore, Europe should lead all aspects of software quality both in research and practice.

Advances in software engineering and quality have led to relevant techniques, methods, and tools: process models and methods (e.g. CMMI, ISO/IEC 15504, agile methods, etc.), quality assurance techniques and methods (software testing, review processes, metrics, quality models, etc.), etc. Many of them have been successfully implemented but are extremely focused on technical aspects.

However, the workforce represents the main cost of any project: professional software development is mainly a social activity. Only a small percentage of studies have addressed human factors and their impact on software, maybe because it requires a multidisciplinary approach, with strong links to industry, practitioners and professional bodies. Although software quality assurance has a long tradition, there is a lack of research on this practical aspect. In particular, the extended study of the influence of human and organizational factors (HOF) on the quality of software development, maintenance and management has been neglected. Thus project and corporate managers do not have the basis for decision making methods to forecast their effects on quality results of software development.

For example, some research works have revealed problems in training, performance and good practice in organizations as well as in professionals' attitudes in software testing or impact of training in software professionals: unsystematic test case design leads to 35% of unnecessary cases while covering less than 50% of functions (Jones, 1998), at least 10 days of annual training increase productivity in 8% (Fernandez-Sanz and Misra, 2012), etc. These are clear indications which may feed decision models for managers to decide investment in human and organizational aspects of software quality.

However, wider and deeper analysis is needed as we will show in next sections. In the end, this lack of

Sanz, L., Pérez, J., Díez-Folledo, T. and Misra, S.
Researching Human and Organizational Factors Impact for Decisions on Software Quality.
DOI: 10.5220/0006003702830289
In *Proceedings of the 11th International Joint Conference on Software Technologies (ICSOFT 2016) - Volume 1: ICSOFT-EA*, pages 283-289
ISBN: 978-989-758-194-6

more comprehensive data implies a lack of firm evidence for managers and organizations to help them to organize and optimize the performance of the main resource: software professionals.

The problem in particular arises from the fact that the extended study of the influence of human and organizational factors on the quality of software development, maintenance and management has been neglected in the research spheres as well as in the industry and practitioners' world. Obviously this situation is a consequence of several contributing causes, some of them related to the difficulty of this type of research (e.g. it usually requires a cooperative and multidisciplinary approach, with good connections to relevant stakeholders outside the academia as well as the need of coping with local variations and conditions) but others with the disdain of the researchers and industry to work on this topic (e.g. a proposal on HOF and software quality was poorly assessed by an EU COST program evaluator because it was not "real research, just something to leave to industry"). This is a paradoxical situation because there are many evidences that confirm that HOF are connected to the most expensive parts of software projects. In parallel managers still insist on investing more money in technical aspects with reduced ROI, always trying to find and buy a "silver bullet" normally with the shape of a new tool or development environment. A contradictory view when considering results of a complete study of factor influencing productivity (Trendowicz and Münch, 2009): success of software projects still relies upon humans while second most commonly considered factors are tool and method but even the best tool or method alone is not a silver bullet and cannot be a substitute for skilled people and effective work organization.

Obviously the ideal situation would be one where studies calculating ROI or other economic indicators, similar to some already created for specific software quality assurance (SQA) or software engineering methods or approaches (e.g. (Rico, 2004)) could be available for guiding managers' decisions on HOF and software quality. However, this has not been still possible for the general set of software quality trade-off (Barney et al, 2012), 61% of the research is providing non-empirically assessed solution proposals and only 28% offers empirical evidence.

This paper is organized as follows. Section 2 reviews the general panorama of the literature contributions to the study of HOF in software quality focusing on evaluating the impact of factors on software quality results. Section 3 analyses some relevant contributions to detect and discuss the challenges in the field of HOF to overcome the poor state of development of research. Final Section 4 outlines some conclusions and suggests actions for future work in the area.

2 GENERAL ANALYSIS OF CONTRIBUTIONS

Searching literature to find contributions where the relationship between HOF and software quality is identified and analysed is not an easy task. Terms for searching literature databases are less precise and concise than the ones used in m ost technical topics in software engineering. Moreover, a term like "human" is frequently used in the field of HCI (Human-Computer Interaction) and the development of effective UI (user Interfaces) thus populating search results with a good number of references which do not address our perspective. This trend is also seen in compilation of contributions like (Saaed, 2014) where 26 academics and practitioners present practice reports, discussions, inherent issues, implementation strategies, latest research, as well as case studies from around the world focusing on the human factors aspect of software design and development (but not exactly addressing software quality).

It is worth mention that there was a seminal initiative by DeMarco and Lister with their book Peopleware (Demarco and Lister, 1999) already in 1987, where they grouped under this expression all people's related issues: teamwork, group dynamics, organizational factors, with a special focus on productivity and all types of factors related to people who work in software projects. This happened many years after Brooks stated in the Mythical Man Month book (Brooks, 1975) that product quality is strongly affected by organization structure. Others (Nygaard, 1986) (Naur, 1991) had also remarked the human, the social side of cooperation in teams and working environments of software development. In many cases, HOF have been present in studies as linked to the pair productivity and quality (Fernandez-Sanz and Misra, 2011) but the side of quality has frequently attracted less attention and detailed data than the productivity one. However, an early literature review of productivity factors in software projects also confirmed that soft or human-related factors in software engineering are often not analysed with equal detail as more technical factors (Wagner and Ruhe, 2008). This study also outlined an important cross-relationship between quality and productivity: the quality of the product influences motivation and hence productivity. The relationship between social

issues and productivity has been also analysed in (Yilmaz and O'Connor, 2012), where it is suggested that social capital is essential for building highly productive development environments.

The permanent problem is the lack of continuance of this line of action. However, there was a new impulse some years ago when some researchers insisted on promoting a wider variety of research approaches to deal with the undeniably social nature of software development. The consideration of social sciences methods and their adaptation to software engineering resulted in their acceptance in the software engineering research mainstream: e.g. qualitative research with examples like ethnographic methods, etc. (Dittrich et al, 2010).

Although there is not a definitive and clear review of the research contributions to the connection between HOF and software quality, other works have tried to cover the relationships between HOF and similar aspects of software engineering and development. A systematic literature review (SLR) that covered papers published in 2000-2010 (Pirzadeh, 2010) was carried out to identify human factors and their impact on development process. The SLR concludes that despite HOF impact on process success, performance and quality, there is an evident lack of primary and secondary studies on HOF: again software engineering studies still tend to be mainly focus on technical context. It also confirms that most researchers in this area think that HOF are the centre of development process.

The results presented in (Seth, 2015), empirically collected in 13 software companies, suggest that the human factor is the basis of software quality. Regarding organizational factors, the author concludes that structures, mode of operation and communication channels contribute to success or failure in the software quality construction process.

Another area which has leaded to a more intensive study of HOF is agile development. Based on closely related-to-HOF foundations, agile methods have high-lighted their impact on results. Although agile paradigm was firstly launched without a sustained effort on sound empirical evidences, more relying on conviction and direct observation of promoters, this area has been finally able to produce studies in impact publications (Dyba and Dingsoyr, 2008).

Although many works have tried to establish relationships between psychological factors and performance when developing or testing software, only a few empirical studies have been carried out to study the influence of these specific factors on software quality. Moreover, the context of the studies is frequently defined as a project with non-professional participants (students). Although it is frequently used in other disciplines like psychology, this approach has generated a point of controversy, attracting obvious criticism if the study is not addressed at the behaviour of novice or non-experts (Kitchenham et al, 2012). For instance, (Acuña et al, 2009) presents a quasi-experiment that analyses the relationships between personality and software product quality based on responses to questionnaires by computing students that work in teams. Sometimes the factor is motivation where there is a good number of identified studies in a SLR (Beecham et al, 2008). Cultural factors and collaboration have gained especial relevance after the generalization of Global Software Development (GSD) projects and the corresponding research studies to optimize results as well as overcome detected barriers which hinder the performance in multinational settings (e.g. see the tertiary study on SLR on GSD (Verner et al, 2012)).

3 ANALYSIS OF RELEVANT CONTRIBUTIONS

Although the general review of contributions on HOF in software quality may lead to the impression that there is an acceptable population of contributions, this is not the case of the studies pursuing the determination of a clear impact relation to software quality. It is extremely difficult to find sound work based on the quantitative analysis of impact of HOF in direct quality results. Obviously this varies from one studied aspect to other.

The analysis of the quality in GSD deserves a special mention in our review. According to (Misra and Fernández, 2011) (Misra et al, 2013), one of its major drawbacks is that low quality software can be produced due to the negative impact of people issues on this type of projects. Software is developed through teams, from multiple geographic locations, different cultures, languages, etc. Thus, HOF such as communication, knowledge management, coordination, collaboration and group awareness trust are key to improve the quality of the software. Connected to GSD, (Thomas et al, 1996) is one of the few studies that empirically address the effect of cultural diversity in software quality. The aim of that work is at identifying cultural factors that impact on quality, and to give recommendations and guidelines for software process improvement.

There is little empirical evidence of the effects of organizational factors on software quality. (Lavallée and Robillard, 2015) provides the outcomes of a study carried out in a professional environment. It shows

that certain organizational factors, that might not affect project success, negatively affected software quality. And quality factors can have a major impact on maintenance costs. Similarly, other papers (Mathew, 2007) (Leung, 2001) (Jaktman, 1998) stress that the impact of the organizational culture on software quality has been ignored by researchers in academia and industry. However, there is a more direct relationship in (Nagappan et al, 2008) where authors investigate the relationship between organizational structure and software quality by proposing a set of eight measures that quantify organizational complexity. The results of that work provide empirical evidence that the organizational metrics are related to, and are, effective predictors of failure-proneness. (Ryan and O'Connor, 2013) present an empirical study that shows how social interaction is key in software development teams.

The influence of motivation has been theoretically analysed in many studies (Beecham et al, 2008) (Griesser, 1993) (Basili and Reiter, 1979). Empirical studies on motivation are difficult to find because it is hard to quantify. Although organizations are aware of its importance, they normally focus on other factors that are easier to measure.

Given the difficulty of analysis of HOF in organizations, many studies rely on the use of surveys and questionnaires to collect information or other approaches like Delphi studies. Although this is a valid instrument for this type of research, it is not as effective as controlled experiments or quasi-experiments.

One important observation is that empirical research works in this area usually rely on case studies as methodological approach. Although case studies are very useful methods, they help to investigate a single entity in a particular period of time. Other disadvantage of case studies is that the data collection and analysis are more open to interpretation and research bias. This is why the use of other types of empirical and theoretical research methods is needed.

In a first revision looking for empirical studies that connect HOF and software quality, a low percentage (around 5%) of the revised papers carried out controlled experiments. The searching has been carried out looking for keywords such as empirical, survey, quality, human factors, etc. in the main databases (IEEExplore, ACM, Google Scholar, Springerlink, etc.). One of these mentioned controlled experiments is carried out in (Bernárdez et al, 2014), where the performance of two groups of students is compared to analyse the effects of mindfulness on the development of conceptual models. A good example of a quantitative study with data taken from industry (although only in one company) and creating a decision model can be found in (Krishnan et al, 2000): personnel capability is identified as a clear positive and quantified impact on the quality of the product.

One of the area where there are promising advances is software testing. Recent interest by industry on testing during the recent economic crisis has been synergetic with the existing tradition of research. Relevant connections between HOF in testing and quality results have been detected by contributions in the specific literature. For instance, (Fernández-Sanz and Misra, 2012) focuses on the proper generation of test cases and shows the results from experiences with more than 70 software professionals. Relevant trade-off connection was detected: software practitioners with specific trained in testing were better than the untrained ones in detecting defects and in being more efficient: less repetition of similar test cases which do not offer detection value (average repeated test cases was more than 50%). Moreover, it showed the inconsistency of test case design to their own rank of priority for functionality. Another example is the survey in (Kanij et al, 2014) where participated more than 100 software testing practitioners concluding with evidence that HOF are essential to improve the performance of software testers. Also, the empirical study shown in (Krishnan et al, 2000) remarks that personnel capability has significant positive impact on the quality of the product. The survey presented in (Fernández-Sanz et al, 2009) investigates the influence of 23 factors on testing performance. It reveals that training and other soft factors have a positive effect on the results. The qualitative approaches have also started to be present in the testing with studies working with the ethnographic research approach (Rooksby et al, 2009) (Martin et al, 2007).

4 CONCLUSIONS

As a conclusion, we want to support that a sustained effort on researching HOF as a key factor is essential for making relevant informed decision for trading-off investments in software quality. HOF are usually overlooked in software engineering because the relationship to software development is complex to investigate and involve many different domains. This is not a recent idea (Siakas and Georgiadou, 2002) but the pass of time has not leaded to a much better situation. Academia and researchers who are

frequently self-confined to strict branches of knowledge are not much inclined to work in multidisciplinary projects and teams. But this should not preclude the work on HOF: our analysis confirms they are frequently more influential in terms of costs and effects than many other technical topics extensively studied.

Other engineering branches even less exposed to the influence of human resources than software development have accepted in a natural way that they need to study in a formal and scientific way the impact of HOF on quality. A clear case is aviation maintenance and inspection where the studies e.g. on human error management (Latorella and Prabhu, 2000) (Xavier, 2005) cumulate a long path of scientific contributions leading to its implementation as best practices in companies and regular compulsory training programs for all employees in the area. Why do not we think in analysing human errors in software development too by studying successful methods tested in other disciplines?

Software quality would benefit from the work in other similar engineering areas without forgetting that it is always necessary to be cautious when adapting them to the very specific case of software projects. They study the factors which determine the best results for total quality management (TQM) (Shahraki et al, 2011) even determining that the main explanation for TQM failure is the lack of attention to human side (Edwards and Sohal, 2003). They are also giving us interesting clues on the impact of HOF and there are already studies working on the area of software quality confirming that employee empowerment is the factor with highest impact on customer satisfaction (Parzinger and Nath, 2000) or the especial role of leadership styles (Parzinger et al, 2001).

We would like to see more researchers and more work in HOF as an inspiration for the area of software quality. We agree with (Barney et al, 2012): without empirical research, practitioners, managers and researchers are unable to determine which approach is the most suitable for a given context. HOF is the least explored and analysed area. Greater empirical research is needed: while there is a wide range of proposed solutions, the lack of empirical evidence implies only limited comparisons and evaluations are possible between options. This research could be hard to do if there is not a change in perceptions by experts and funding agencies' evaluators: they should also review their assumptions to avoid this field remained improperly explored, keeping software engineering out of the largest and most important asset for efficiency and effectiveness: people and their organizations.

This analysis resulted from research during the Iceberg project, aimed at providing new research skills and broad horizons in the evaluation of Software Quality Assurance investment oriented to support decision-making through a model-based process. As a continuation of this work, a specific SLR on HOF and software quality is in progress as well as the creation of an inventory of evidences of impact. Both should act as basis for further work on decision models. This database will hopefully serve as reference of the specific identified trade-offs between HOF and software quality given visibility to the area, attracting new researchers and industry managers.

ACKNOWLEDGEMENTS

This work has been partially funded by European Commission for project ICEBERG no. 324356 (7h Framework Programme IAPP Marie Curie program).

REFERENCES

Acuña S., Gómez M., Juristo N. 2009. How do personality, team processes and task characteristics relate to job satisfaction and software quality?, *Information and Software Technology*, v.51 n.3, p.627-639.

Barney, S., Petersen, K., Svahnberg, M., Aurum, A., & Barney, H. 2012. Software quality trade-offs: A systematic map. *Information and Software Technology*, 54 vol. 7, 651-662.

Basili, V.R., Reiter Jr., R.W. 1979. An Investigation of Human Factors in Software Development. *Computer* 12(12), pp. 21–38

Beecham S., Baddoo N., Hall T., Robinson H., Sharp H. 2008. Motivation in Software Engineering: A systematic literature review, *Information and Software Technology*, Volume 50, Issues 9–10, pp. 860-878

Beecham, S., Baddoo, N., Hall, T., Robinson, H., Sharp, H. 2008. Motivation in Software Engineering: A systematic literature review. *Information and Software Technology* 50(910), pp. 860–878

Bernárdez B., Durán A., Parejo J.A., Ruiz-Cortés A. 2014. A controlled experiment to evaluate the effects of mindfulness in software engineering. In *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement* (ESEM '14). ACM, New York, NY, USA, Article 17, 10 pages.

Brooks, F. P. 1975. *The mythical man-month. Reading*, MA: Addison-Wesley.

Demarco, T., Lister T. 1999. *Peopleware: Productive Projects and Teams*, Dorset House, 2nd edition

Dittrich Y., Sharp H., Theophilus H.W., De Souza C., Korpela M., and Singer J.. 2010. Cooperative and human aspects of software engineering: *CHASE 2010. SIGSOFT Softw. Eng. Notes* 35, 5, pp. 27-29

Dyba T., Dingsoyr T. 2008. Empirical studies of agile software development: A systematic review, *Information and Software Technology*, Volume 50, Issues 9–10, pp. 833-859

Edwards R., Sohal A.S. 2003. The human side of introducing total quality management: Two case studies from Australia. *International Journal of Manpower*, 24, 5, pp.551-567.

Fernández-Sanz L., Villalba M.T., J Hilera J.R., Lacuesta R. 2009. Factors with Negative Influence on Software Testing Practice in Spain: A Survey, *Volume 42 of the series Communications in Computer and Information Science* pp 1-12.

Fernández-Sanz, L., Misra, S. 2011. Influence of human factors in software quality and productivity. *In Computational Science and Its Applications-ICCSA* pp. 257-269. Springer Berlin Heidelberg.

Fernandez-Sanz, L.; Misra, S. 2012. Practical Application of UML Activity Diagrams for the Generation of Test Cases. *Proceedings of the Romanian Academy, Series A*, vol. 13, no 3, p. 251-260.

Griesser, J.W. 1993. Motivation and information system professionals. *Journal of Managerial Psychology* 8(3), pp. 21–30

Jaktman C. B. 1998. The influence of organisational factors on the success and quality of a product-line architecture. *Proceedings of ASWEC '98: Aus. Software Engineering Conference*, pp. 2-11

Jones, T. Capers. 1998 *Estimating software costs*. McGraw-Hill, Inc.

Kanij, T., Merkel, R., Grundy, J. 2014. A Preliminary Survey of Factors Affecting Software Testers. *23rd Australian Software Engineering Conference (ASWEC),* 2014, vol., no., pp.180-189.

Kitchenham B. A. et al. 2012. Preliminary guidelines for empirical research in software engineering. *IEEE Transactions on Software Engineering,* vol. 28, no. 8, pp. 721-734

Krishnan M. S., Kriebel C. H., Kekre S., and Mukhopadhyay T. 2000. An Empirical Analysis of Productivity and Quality in Software Products. *Journal Management Science,* vol. 46 Issue 6, pp. 745-759

Krishnan, M. S., Kriebel, C. H., Kekre, S., Mukhopadhyay, T. 2000. An empirical analysis of productivity and quality in software products. *Management science*, 46(6), pp. 745-759.

Latorella, K. A., Prabhu, P. V. 2000. A review of human error in aviation maintenance and inspection. *International Journal of Industrial Ergonomics*, 26(2), pp. 133-161.

Lavallée M, Robillard P.N. 2015. Why Good Developers Write Bad Code: An Observational Case Study of the Impacts of Organizational Factors on Software Quality. *International Conference on software Engineering (ICSE2015),* pp. 16-24

Leung H. 2001. Organizational factors for successful management of software development. *Journal of Comp. Info. Systems*, vol. 42, pp. 26-37.

Martin, D., Rooksby, J., Rouncefield, M., Sommerville, I. 2007. Good Organisational Reasons for Bad Software Testing: An Ethnographic Study of Testing in a Small Software Company. *International Conference on Software Engineering,* ICSE pp. 602-611

Mathew J. 2007. The relationship of organisational culture with productivity and quality. A study of Indian software organisations, vol. 29, pp. 677-95.

Misra S., Colomo-Palacios R., Pusatli T., Soto-Acosta P. 2013 A discussion on the Role of People in Global Software Development. *A discussion on the role of people in global software development*. pp. 525-531.

Misra, S. Fernández L. 2011. Quality Issues in Global Software Development. *The Sixth International Conference on Software Engineering Advances ICSEA*

Nagappan N., Murphy B, Basili V. 2008. The influence of organizational structure on software quality an empirical case study. *Proceedings of the 30th international conference on Software engineering,* pp. 521-530

Naur, P. 1991. *Computing: A Human Activity*. AddisonWesley.

Nygaard, K. 1986. Program Development as a Social Activity. *Information Processing 86. IFIP*, pp. 189-198.

Parzinger, M. J., Nath, R. 2000. A study of the relationships between total quality management implementation factors and software quality. *Total Quality Management*, 11(3), pp. 353-371.

Parzinger, M. J., Nath, R., Lemons, M. A. 2001. Examining the effect of the transformational leader on software quality. *Software Quality Journal*, 9(4), pp. 253-267.

Pirzadeh L. 2010. *Human Factors in Software Development: A Systematic Literature Review*. Doctoral dissertation.

Rico, D. F. 2004. *ROI of software process improvement: Metrics for project managers and software engineers.* J. Ross Publishing.

Rooksby, J., Rouncefield, M., Sommerville, I. 2009. Testing in the wild: The social and organisational dimensions of real world practice. *Computer Supported Cooperative Work (CSCW),* 18(5-6), pp.559-580.

Ryan, S and O'Connor, R. 2013. *Acquiring and sharing tacit knowledge in software development teams: An empirical study. Information and Software Technology*, 55 (9). pp. 1614-1624.

Saaed S. 2014. *Human Factors in Software Development and Design*, IGI Global.

Seth F.P. 2015. *Empirical Studies on Software Quality Construction: Exploring Human Factors and Organizational Influences*. Doctoral dissertation.

Shahraki, A., Konarizadeh, M., Paghaleh, M. J., Zarei, M. 2011. HRM effects on TQM. *Business Management Dynamics*, 1(3), pp. 1-12.

Siakas, K. V., Georgiadou, E. 2002. Empirical measurement of the effects of cultural diversity on software quality management. *Software Quality Journal,*10(2), pp. 169-180.

Thomas S. A., Hurley S. F. and Barnes D. J.. 1996. Looking for the human factors in software quality management. *International Conference on Software Engineering: Education and Practice*, pp. 474-480

Trendowicz, A., Münch, J. 2009. Factors Influencing Software Development Productivity—State of the Art and Industrial Experiences. *Advances in computers*, 77, 185-241.

Verner J. M., Brereton O. P., Kitchenham B. A., Turner M. and Niazi M. 2012. Systematic literature reviews in global software development: A tertiary study. *16th International Conference on Evaluation & Assessment in Software Engineering,* pp. 2-11.

Wagner, S., Ruhe, M. 2008. *A systematic review of productivity factors in software development. language*, Technischen Universität Münche, TUM-I0832 Report

Xavier, A. J. 2005. *Managing human factors in aircraft maintenance through a performance excellence framework*. (Doctoral dissertation, Embry-Riddle Aeronautical University).

Yilmaz, M. and O'Connor, R. 2012. Social Capital as a Determinant Factor of Software Development Productivity: An Empirical Study Using Structural Equation Modeling. *Int. J. Hum. Cap. Inf. Technol. Prof.* 3, 2, 40-62.