Creativity in Agile Software Development Methods

Broderick Crawford[1,2,3]([✉]), Kathleen Crawford[1], Ricardo Soto[1,4,5],
and Claudio León de la Barra[1]

[1] Pontificia Universidad Católica de Valparaíso, Valparaíso, Chile
{broderick.crawford,ricardo.soto,claudio.leondelabarra}@ucv.cl
kathleen.crawford.a@mail.pucv.cl
[2] Universidad Central de Chile, Santiago, Chile
[3] Universidad San Sebastián, Santiago, Chile
[4] Universidad Autónoma de Chile, Santiago, Chile
[5] Universidad Cientifica del Sur, Lima, Peru

Abstract. Creativity is an inherent aspect to the development of new products, therefore a critical capacity for software development. Indicators to measure creativity are grouped into two main areas: (1) those related to the creative result itself and its quality (including novelty and usefulness) and (2) those related to the creative team itself (considering indicators such as individual and group satisfaction, development of cognitive skills, group interaction, ...). From this perspective, the aim of this work is to introduce some ideas for assesing the creativity of software products.

Keywords: Creativity · Software engineering · Measures of creativity

1 Introduction

Psychology and Computer Science are growing in a interdisciplinary relationship mainly because human and social factors are very important in software engineering. The development of new software products requires the generation of novel and useful ideas. Software is developed for people and by people [9]. However, most of software engineering research is technical and does not emphasize the human and social aspects [2,3].

By other hand, the traditional development process of new products has been recently criticized in [10], pointing out that fundamental creative aspects are not considered at all and as a consequence this development is not useful, viable or innovative. In this context, it is interesting to study the assesing of creativity in software products, being of particular interest to consider how is done it by the agilists.

Agilists value working software, it is more valuable than comprehensive documentation. Agile teams write code first and then document as needed. They deliver working software often and their progress is best measured by using the

software. This represents a shift in the traditional software development paradigm and is best suited for the actual economy, releasing software at any time the market demands [4,5].

Since human creativity is thought as the source to resolve complex problem or create innovative products, one possibility to improve the software development process is to design a process which can stimulate and measure the creativity of the developers and its products. There are few studies reported on the importance of creativity in software development teams. In a few publications the importance of creativity has been investigated in all the phases of software development process [1,7,8] and mostly focused in the requirements engineering [13,15,17].

Nevertheless, the use of techniques to foster creativity in requirements engineering is still shortly investigated. Moreover, in some studies requirements engineering is not recognized as a creative process in all the cases [12]. We think that analysts, designers, programmers, testers, managers, entrepreneurs, users, researchers and other stakeholders involved in sofware development need to be creative.

Clearly, creativity is related with a wide spectrum of business, it is crucial for designing better products, it initiates innovations and aids in problem solving allowing an organization to survive. But it is often difficult to measure the creativity.

A method for assessing the degree of creativity is necessary to help select the most creative products. We are working (based in the study in [19]) in a method that can help identify the degree of creativity in software products. We intend to be able to assess not only whether a product is creative or not, but also how much creative it is. At first, in this paper we try to ilustrate what is meant by creativity, and what its current measures are, and how adequate these are in software industry.

2   Definition of Creativity

In a recent comprehensive survey of the definitions of creativity [18], Sarkar and Chakrabarti analyzed over 160 definitions proposing a common definition of creativity, as follows: "Creativity occurs through a process by which an agent uses its ability to generate ideas, solutions or products that are novel and valuable". Value, in the context of software products, take on the meaning of utility or usefulness.

Similar views of creativity have also been expressed by other researchers. Furthermore, in [18] they propose measures for creativity manifesting that creativity should be measured directly in terms of novelty and usefulness of the results.

Then, according to the above definition, assessing creativity therefore requires assessment of novelty and usefulness. At the following, some definitions and assesing methods for novelty, usefulness, and creativity are briefly presented.

2.1 Defining, Measuring and Assessing the Novelty of Products

"Novel" are those things that are "new" to all people. "Novelty" is the quality of being new and fresh and interesting [14]. Novelty comprises both new (something that has been recently created) and original (the first one made, it is not a copy).

Different researchers proposed methods for measuring novelty [16,20], mainly focus on the identification of novelty of products and not on their degree of novelty. One way of assessing novelty of a product, is to compare the characteristics of that product with those of other products.

Methods that can be used to decompose a product into its characteristic components or features are suitable for supporting this assessment. A widely used model is the Function–Behaviour–Structure (FBS) model, different works on FBS models illustrates its value for classifying product-characteristics [6]. In relation to apply FBS to software engineering, although initially the authors did not have software in mind when developing their framework, in [11] the authors map software engineering to FBS, representing software engineering artifacts and practices using the Rational Unified Process.

Function, behaviour and structure in FBS model are defined as follows:

– Function: descriptions of what a system does: it is intentional and at a higher level of abstraction than behaviour.
– Behaviour: descriptions of how a system does its function. This is generally at a lower level of abstraction than function.
– Structure: it is described by the elements and interfaces with which the system is constructed.

It is noted that product-characteristics can be employed to ascertain the relative degree of novelty of products and the FBS model can be used for determining novelty of software products.

2.2 Defining, Measuring and Assessing the Usefulness of Products

The common definition of usefulness is "the quality of having utility and especially practical worth or applicability". Others definitions consider usefulness in terms of "utility" in terms of appropriateness and social value.

In order to study the methods for assessing product usefulness, we were unable to find in the literature direct measure for usefulness. When a product may be perceived as useful?. It is the actual use of the product and its results that validate its usefulness. Then, the usefulness of a product should be measured by its actual use. In relation with the importance of use or level of importance, it depends on the impact of the product on the lives of its users. Some products are indispensable, while others are not. Accordingly, it is possible identify different levels of usefulness of a product.

Then, a method for assessing the usefulness of products should consider the importance of usage, popularity of usage, and rate of use as criteria for assessing overall usefulness.

2.3   Assessing the Creativity of Products

Considering that novelty and usefulness of products should be consider as the main influences on creativity, it is necessary to express creativity as a function of these two factors. Basically, any expression that try to measure creativity should consider it as the product of these two factors: novelty and usefulness.

3   Conclusions

It was exposed that according to the definition of creativity, it should be measured directly in terms of novelty and usefulness of the results. Then, assessing creativity requires assessment of novelty and usefulness. Some methods for assess novelty, usefulness, and creativity were briefly presented. It is clear that a better formalization of these influences is an area of research in developing.

References

1. Crawford, B., de la Barra, C.L.: Enhancing creativity in agile software teams. In: Concas, G., Damiani, E., Scotto, M., Succi, G. (eds.) XP 2007. LNCS, vol. 4536, pp. 161–162. Springer, Heidelberg (2007)
2. Crawford, B., Soto, R., de la Barra, C.L., Crawford, K., Olguín, E.: Agile software teams can use conflict to create a better products. In: Stephanidis [21], pp. 24–29 (2014)
3. Crawford, B., Soto, R., de la Barra, C.L., Crawford, K., Olguín, E.: The influence of emotions on productivity in software engineering. In: Stephanidis [21], pp. 307–310 (2014)
4. de la Barra, C.L., Crawford, B.: Fostering creativity thinking in agile software development. In: Holzinger, A. (ed.) USAB 2007. LNCS, vol. 4799, pp. 415–426. Springer, Heidelberg (2007)
5. de la Barra, C.L., Crawford, B., Soto, R., Misra, S., Monfroy, E.: Agile software development: it is about knowledge management and creativity. In: Murgante, B., Misra, S., Carlini, M., Torre, C.M., Nguyen, H.-Q., Taniar, D., Apduhan, B.O., Gervasi, O. (eds.) ICCSA 2013, Part III. LNCS, vol. 7973, pp. 98–113. Springer, Heidelberg (2013)
6. Gero, J.S., Kannengiesser, U.: The situated function–behaviour–structure framework. Des. Stud. **25**(4), 373–391 (2004)
7. Glass, R.L.: Software Creativity. Prentice-Hall Inc., Upper Saddle River (1995)
8. Gu, M., Tong, X.: Towards hypotheses on creativity in software development. In: Bomarius, F., Iida, H. (eds.) PROFES 2004. LNCS, vol. 3009, pp. 47–61. Springer, Heidelberg (2004)
9. John, M., Maurer, F., Tessem, B.: Human and social factors of software engineering: workshop summary. SIGSOFT Softw. Eng. Notes **30**(4), 1–6 (2005)

10. Kotler, P., TríasdeBes, F.: Marketing Lateral. Editorial Pearson/Prentice Hall, Spain (2004)
11. Kruchten, P.: Casting software design in the function-behavior-structure framework. IEEE Softw. **22**(2), 52–58 (2005)
12. Maiden, N., Gizikis, A., Robertson, S.: Provoking creativity: imagine what your requirements could be like. IEEE Softw. **21**(5), 68–75 (2004)
13. Maiden, N., Robertson, S.: Integrating creativity into requirements processes: experiences with an air traffic management system. In: 13th IEEE International Conference on Requirements Engineering (RE 2005), 29 August–2 September 2005, Paris, France, pp. 105–116. IEEE Computer Society (2005)
14. Makins, M.: Collins English Dictionary. HarperCollins, Glasgow (1991)
15. Mich, L., Anesi, C., Berry, D.M.: Applying a pragmatics-based creativity-fostering technique to requirements elicitation. Requir. Eng. **10**(4), 262–275 (2005)
16. Piffer, D.: Can creativity be measured? An attempt to clarify the notion of creativity and general directions for future research. Think. Skills Creativity **7**(3), 258–264 (2012)
17. Robertson, J.: Requirements analysts must also be inventors. IEEE Softw. **22**(1), 48–50 (2005)
18. Sarkar, P., Chakrabarti, A.: Studying engineering design creativity-developing a common definition and associated measures (2008)
19. Sarkar, P., Chakrabarti, A.: Assessing design creativity. Des. Stud. **32**(4), 348–383 (2011)
20. Shah, J.J., Smith, S.M., Vargas-Hernandez, N.: Metrics for measuring ideation effectiveness. Des. Stud. **24**(2), 111–134 (2003)
21. Stephanidis, C. (ed.): HCI 2014, Part I. CCIS, vol. 434. Springer, Heidelberg (2014)