MetricRA: Learning Software Metrics through Augmented Reality

Rebeca Motta[1], Mario Bonicenha[1], Claudia Susie Rodrigues[1], Cláudia Werner[1]

[1]Computer Systems Engineering Programm at Federal University of Rio de Janeiro
Rio de Janeiro, Brazil

{rmotta, mario.bonicenha, susie, werner}@cos.ufrj.br

Abstract. *Augmented reality creates a bridge between virtual and real world, providing stimulating resources for different purposes. This technology enables new teaching possibilities since it can bring more abstract concepts into reality and put the knowledge related to several areas, such as Software Engineering, into practice. MetricRA is a tool developed to help Software Engineering students to understand Cohesion and Coupling metrics. The solution was implemented with Augmented Reality technology, where the user can control a class diagram to observe the metrics transformation. This article describes MetricRA tool and presents a study conducted to evaluate its ability to contribute to the understanding of the concepts proposed.*

1. Introduction

With the technological evolution, we can see the increasing availability of products and applications using Augmented Reality (AR). This technology can be applied for different purposes, in education, for instance, enables a different interaction that can help improve the learning experience [Wu *et al.* 2013]. This new option has already been used in Software Engineering, mainly assisting the understanding of software systems [Rodrigues, Werner, & Landau 2016]. This work aims to contribute to teaching Software Metrics. As software becomes more pervasive, metrics concerns are critical, being an important step towards developing quality software [Fenton & Pfleeger 1997].

In this scenario, the MetricRA tool, proposed in this work, focuses on the student experience and aims to help his/her learning regarding the concepts of Cohesion and Coupling Metrics. We define Cohesion as the relative functional robustness of a module, and Coupling as the relative interdependence between the modules [Pressman 2014]. The internalization of these concepts by the students is essential, since their conscious use, throughout the development, results in systems that have a better modular organization. The choice of these metrics came from two perspectives: education and technology. From teaching, the choice was due to the importance of these metrics in software design and architecture, as they affect the maintenance and organization of the system. From the technology, the visualization of the metrics is presented in an interactive way where AR potential can be exploited. The MetricRA uses AR to apply the metrics in a practical way, allowing the student to modify the structures of a diagram, analyzing Cohesion and Coupling metrics, and visualizing the results of the changes in a direct way.

2. Theoretical Background and Related Work

In this session, a theoretical background is presented regarding AR and Software Metrics. These concepts are the foundation of this work and are interrelated as the basis for the solution proposed by the tool. We also present Related Work regarding software visualization solutions, showing how the MetricRA tool fits in this research area.

2.1. Augmented Reality (AR)

AR relies on user interaction to supplement the real world with virtual objects that seem to co-exist in the same space. This technology is known to increase the interest of the user, thus bringing more attention to what is being manipulated in the application, which provides a great educational value. By using new strategies for passing knowledge and stimulating other senses, it is possible to collaborate in the cognitive process of the student, bringing also practical experimentation of content [Jin & Yano 1997].

With this technology, the content is not limited to the textual format, and the connection to the real world can be through graphics, images and videos, broadening the student's perception. AR´s goal is to enrich the physical environment with real-time virtual information. Therefore, interaction is a fundamental part of this technology. In the development AR applications, an interface between the user and the virtual environment is exploited to provide that experience to the user. This interface can be done in several ways. A rather simple and widely used option is through markers, where the user points a camera (e.g. mobile phone) to the marker and the application displays its corresponding 3D information overlapped with the marker. From it, virtual elements are combined with real things enabling a different kind of the interaction. Another example of AR can be recognized through the successful game, Pokémon Go.

2.2. Software Metrics

Metrics represents an important concept in Software Engineering, defining a quantitative measure of the degree to which a system, component or process has for a given attribute [IEEE 1990]. These measures can be applied to the product, artifacts or documents produced during the software life cycle, which can be evaluated in different ways, as to their internal or external quality.

In this paper, we explore the Cohesion and Coupling metrics, which are measures related to the software design that contribute to its internal quality. The concepts of Cohesion and Coupling, although having a great impact on the internal quality of systems, end up being overlooked by beginners and also by users who are often interested mainly in the interface and utility of the system [Fenton & Pfleeger 1997]. Therefore, teaching these concepts, in a direct and practical way, can contribute to software quality and to the understanding of their importance. The Cohesion metric measures how focused the class responsibilities are in meeting a specific goal. A highly cohesive class has clear and well-defined responsibilities, while a class with low Cohesion has many different and unrelated responsibilities. The Coupling metric refers to how much a functional unit depends on another, that is, the relationship between the parts of a system. The development of software with high Cohesion and low Coupling contributes to, among other things, maintenance and reuse [Chidamber & Kemerer 1992].

2.3. Related Work

The purpose of this work is to introduce a tool for visualizing software, especially regarding software metrics. As an example of related work we have a survey of visualization techniques, both in 2D and 3D, representing the static aspects of software [Caserta & Zendra 2011], and a review which identifies different solutions to visualize software metrics, including an analysis of the visual attributes and interaction mechanisms [Titang & Dur 2015]. Through these works, it is possible to observe the advances and current challenges in the area, and how our work fits in that context. A similar work to MetricRA is presented by Churcher *et al*. [Churcher, Irwin, & Kriz 2003], where the concept of Cohesion of a class is presented by a 3D image, enriched by size, shape and color. As a result, the visualization of the class takes an abstract format, like chemical molecules, representing the relationship of its elements. The differential of our work regarding the others is given by the technology used and the purpose. With AR, we achieve visual expressiveness and the capability to adapt to the user's interactions in real time, unlike a 2D image. This proposal offers more interactivity than the current related work and has a concern with education alongside with visualization.

3. MetricRA Tool

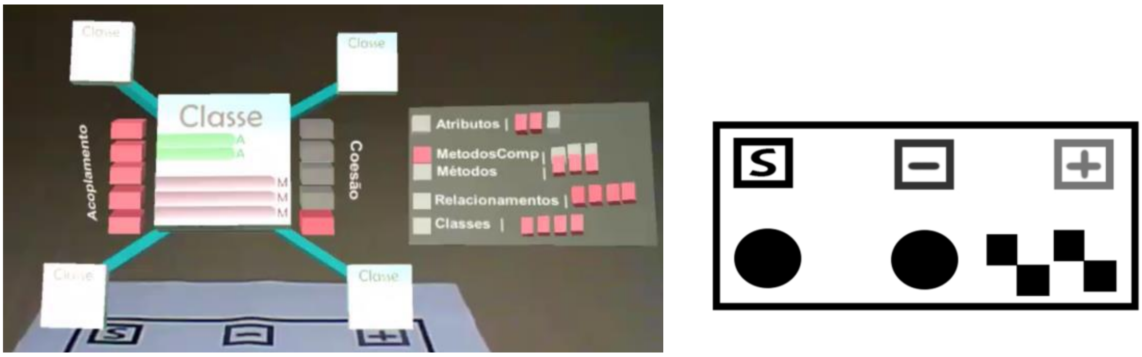
MetricRA consists of an application that uses AR to help students understanding the concept of Cohesion and Coupling metrics. The main objectives of the tool are to (i) help students internalize the concepts; (ii) reduce the gap between theory and practice through the use of contextual examples; (iii) provide immediate feedback for a rapid assimilation of knowledge; (iv) provide an exploratory and intuitive environment through students' interaction; (v) be simple and easy to use. To understand a concept, the best way is to see it in practice and AR can contribute to this. AR is widely applied to simulation and makes the learning process more dynamic. Among its main advantages are immediate feedback and continuous evaluations [Wu *et al*. 2013]. For the development of the tool, Unity5 3D[1] engine was used, as the development platform; and Vuforia[2], an extension to develop AR applications. The resources needed to run the tool is a computer equipped with a webcam. The tool was implemented in C# language.

Figure 1A shows an image of the MetricRA tool execution. In the center of the image, there is the main class that relates to other classes. On the right, there is a menu of options and diagram structures that the student can interact with. With the help of AR, it displays a class diagram overlapped on a marker. Through the menu, one can change the structure of the diagram, for example: change the number of attributes and methods. According to the chosen option, the tool changes the diagram, recalculates and displays Cohesion and Coupling, indicating their satisfaction level that varies from satisfactory to unsatisfactory, in green and red colors respectively.

In Figure 1A, the main class has low Cohesion (a red cube on the right) meaning that the class is not converging only to one main goal (unsatisfactory). The coupling, on the other hand, is high level (five red cubes on the left), meaning that it is not satisfactory since it is depending directly on many other classes. In the example, the class has two attributes and three non-shared methods.

[1] https://unity3d.com
[2] https://developer.vuforia.com/

In MetricRA, a marker was used (Figure 1B). The marker is used to change the structures and to select the options in the menu. It is possible to increase or decrease the number of structure items. The marker has three buttons: "+" which adds the selected item in the right menu, "-" to decrease and "S" to select one of the menu options.



A B

Figure 1. A - Image of the MetricRA tool execution. B - Tool Marker.

For didactic reasons, the metrics used simple formulas calculated according to the diagram structure: methods, shared methods, relationships and number of classes. The Cohesion metric was calculated according to LCOM1 and LCOM2 (Lacks of Cohesion Metrics) [Chidamber & Kemerer 1992], which considers the information shared between the methods. The formula used for Cohesion was: 1-(SM)/(TM*SA), where SM stands for Shared Methods, TM means the number of Total Methods and SA, the number of Shared Attributes between the methods. For Coupling, a C/R ratio was used, where C is the number of Classes and R is the number of Relationships of that class. Since it is a classical approach for metrics, we chose the LCOM CK suite. Different works redefine them but remain based on the original proposal. In the tool is possible to add other metrics and redefine the current ones, by adjusting the formulas.

4. Evaluation

The effectiveness of new tools to support teaching and learning should be evaluated in order to observe their effects on users [Von Wangenheim, Savi, & Borgatto 2012]. In addition, since AR is a recent technology that allows unconventional interaction, it is essential to analyze its quality, mainly regarding the usability of its applications [Martins, Paulo, & Correa 2013], before being applied in the classroom. Thus, a feasibility study was initially conducted with the objective of analyzing the tool support to understanding the Cohesion and Coupling metrics, through the real-time interactivity offered by AR, regarding to motivation effectiveness, user experience and learning aid.

4.1 Planning

In the study, we aim to observe the ability of the MetricRA to support the understanding of Cohesion and Coupling metrics in a different experience through the use of AR. Based on GQM (Goal-Question-Metric) [Basili, Caldeira, & Rombach 1994], the objective of the evaluation is: to analyse *the MetricRA tool* with the purpose of *characterizing it,* in relation to *its motivation effectiveness, user experience and learning aid*, from the point of view of *undergraduate students,* in the context of *teaching metrics in Software Engineering.*

As a basis for the evaluation material, the methodology proposed by Savi *et al*. [Savi, Wangenheim, & Borgatto 2011] was used. This methodology is focused on the evaluation of the quality of educational games, however, the material includes contents such as the evaluation of training [Kirkpatrick & Kirkpatrick 2009], motivational strategies [Keller 1987] and, also consider user experience, of great importance in the AR context. The adaptation of the methodology for the evaluation of the tool considered three main aspects: **Motivation**, **User Experience** and **Learning**, which seek to reflect the evaluation goals. From the original proposal [Savi *et al*. 2011], some aspects outside the scope of the tool, such as Challenge and Social Interaction, were not considered.

The evaluation form presented 15 statements that considered the three aspects to be evaluated. To indicate their experience, they evaluated these statements on a 5-point Likert scale: (-2) strongly disagree, (-1) disagree, (0) neither agree nor disagree, (+1) agree and (+2) strongly agree, with the freedom to include comments on each item, if necessary. As for the learning aspect, the participant evaluated his/her perception of the concepts covered by the tool before and after the experience. The last part of the evaluation had space for the participant to express the positive and negative points of the tool so that it was also possible to identify improvement opportunities.

4.2 Execution

All participants signed a Consent Term and a Characterization Form (completed before interacting with the tool). The participants received a numeric code for anonymization and received the same instructions in video format. After that, participants had the opportunity to interact with the tool and then carry out the proposed tasks, individually and without any contact with each other.

Pilot Study. Performed by a member of the Virtual and Augmented Reality laboratory, to observe the suitability of the study. From the results, we made some improvement, such as a better explanation, as well as clarity in the presented virtual elements.

Proof of Concept. The participants were chosen for convenience and were students of the Computer Science course. Seven participants performed the assessment and they reported that the video was fast and with poor quality audio. From these comments, some details were added to the tool and the video updated.

Observational Study. The study was carried out in April 2017 with the 10 undergraduate students in computing, who participated voluntarily in response to invitations published in students mail list. The analysis of results, presented in the next subsection, refers to the data collected in this study. The explanatory video used is available[3], as well as the executable file and evaluation package[4].

4.3 Analysis and Results

Characterization. All participants had graduation in progress, chosen by convenience, aiming at a group that would have greater gain with the tool. Among the participants, 60% reported having experience with research and 60% reported from 1 to 3 years of experience in the market, with an overlap of 2 participants with experience in both. Most participants reported a low level of experience with software metrics (70%), with

[3] https://goo.gl/UaUEyk
[4] https://goo.gl/nYWDKj

30% having no experience at all (Figure 2). When asked specifically about Cohesion and Coupling metrics, 50% of participants reported having little knowledge about it. As for Class Diagrams, 40% reported that the experience was obtained through classroom projects, and 30% in personal projects.

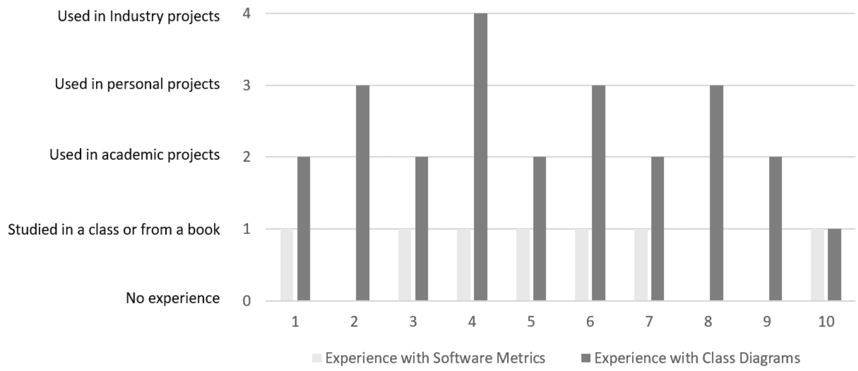


Figure 2. Participants' experience in Metrics and Class Diagrams.

Tasks Execution. To encourage interaction, the participants performed 5 tasks with the support of the tool. Tasks seek to contribute to the sedimentation of related concepts learned before. The participants were asked to interact and answer the following questions while using the tool: a) cite what information is shown in the main class and which structures are, in fact, significant for the Cohesion and Coupling metrics; b) list two ways of achieving high cohesion and two ways of achieving low coupling; c) describe how the metrics can affect the quality of a system. Participants then answered each question and assessed their difficulty level on a scale ranging from 1 (very easy) to 4 (very difficult), as shown in Figure 3. Each participant is represented by their code (from 1 to 10), showing the level of difficulty considered for each task (from very easy to very difficult) and the index of success (the incorrect tasks are marked with "wrong"). The average time for the execution was 28 minutes. The hit rate was very high, with 82% of the tasks performed correctly. This step is necessary so participants can experience the tool with the same orientation, enabling a uniform observation.

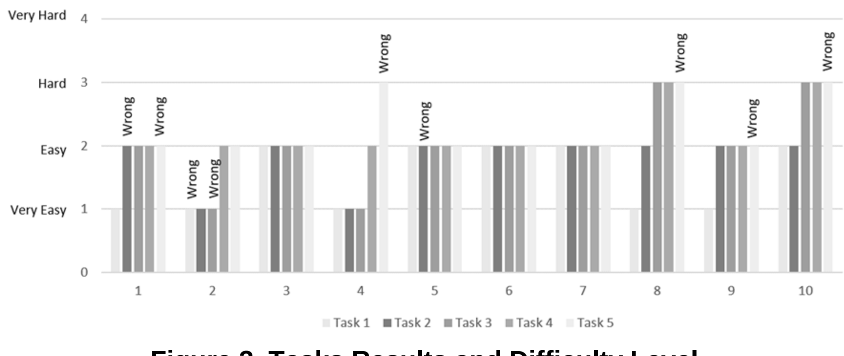


Figure 3. Tasks Results and Difficulty Level.

MetricRA Evaluation. The evaluation of the tool was performed by the 10 participants through a questionnaire adapted from [Savi *et al*. 2011]. The evaluated aspects were divided into three categories: **Motivation**, **User Experience** and **Learning**, according to the research objectives. We understand that these concepts are abstract and may have more than one interpretation, however in this article each category, and their sub-characteristics has been defined based on the protocol used.

The **Motivation** to learn is essential for teaching and the use of motivational strategies seeks to bring an engagement. The Motivation aspect, based on the model used, counts on the sub-characteristics of **Satisfaction**, **Relevance** and **Attention**. **Satisfaction** refers to a positive feeling about the effort in the learning experience. The **Relevance** tries to observe the consistency between the content taught with its utility. **Attention** is related to the cognitive responses to what is being taught. These aspects were translated into seven statements assessed as presented in Figure 4.

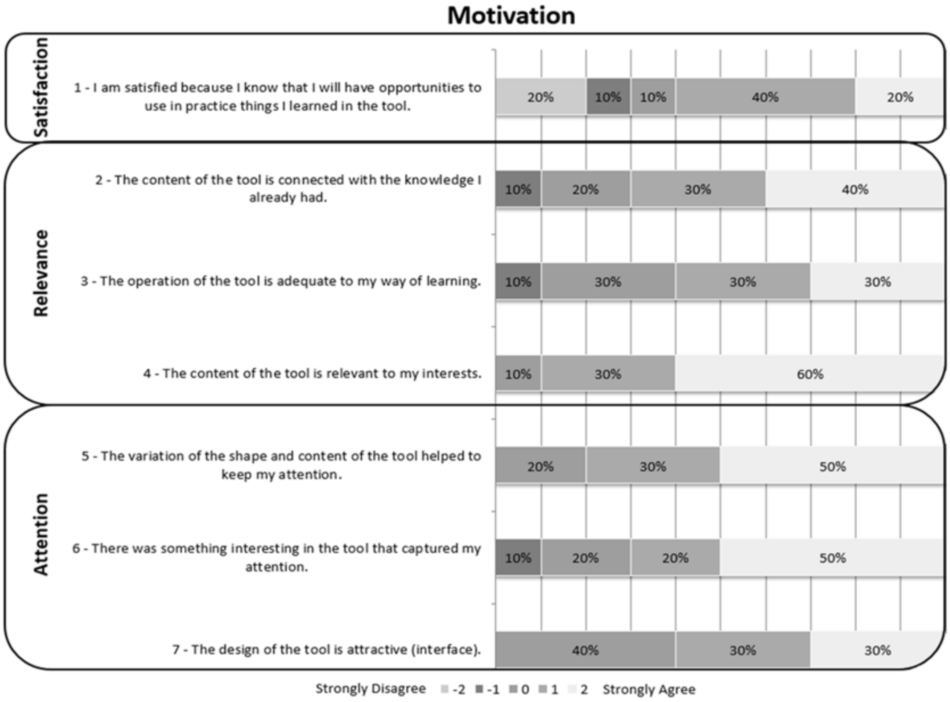


Figure 4. Evaluation of the Motivation category.

Satisfaction was met by 60% reflecting the opportunities to use the concepts treated in the tool in practice. For abstract concepts, such as metrics, aligning practice with theory is a good strategy since it brings the student closer to the concepts. The **Relevance** aspect obtained a high evaluation in all items (with 70%, 60% and 90% in items 2, 3, and 4, respectively). The last item shows the content relevance as to its suitability to the student's academic interests. For **Attention**, the evaluation was also high in all items (with 80%, 70% and 60% in items 5, 6, and 7, respectively). By analyzing the reported comments, participants thought the interface was interactive and attractive (participants 3, 4 and 8), but that there was some instability in the interaction, "deconcentrating" (participant 7), identified as an improvement opportunity.

Figure 5 presents the results of students' responses related to the **User Experience** category. The concept of experience is individual and somewhat abstract, which may hinder its description and evaluation. However, it was of interest to contemplate this concept through the sub-characteristics of **Competence**, **Amusement** and **Immersion**. To provide a good experience, the tool seeks to support the understanding of the concepts by recognizing the participants' previous knowledge, affirming their **Competence**. The **Amusement** aspect seeks to reflect feelings such as distraction and satisfaction. **Immersion** is an important sub-characteristic for an AR tool because it proposes a deviation of focus from real to virtual world.

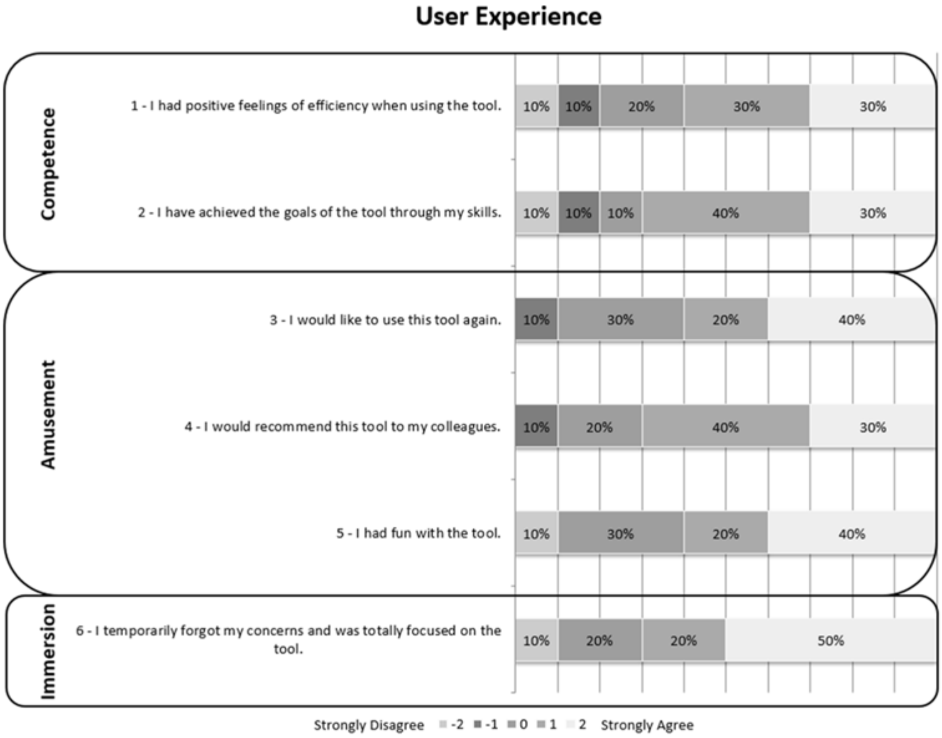


Figure 5. Evaluation of the User Experience category.

In the **Competence** aspect, the evaluation had in average 60% and 70% in the evaluated items. Participants with discordance ratings (10% in each item) left no comment. In **Amusement**, the rating was high on all items (with 60%, 70% and 60% in items 3, 4, and 5, respectively) with positive feedbacks in the comments, such as the tool was "fun and very well illustrated" (participant 8). As for **Immersion**, 70% of the participants agreed that they were totally concentrated on the tool, with positive comments such as: "AR contributed to my interest and concentration" (participant 4).

Finally, items related to **learning** were evaluated, where the items received 70% and 90% of agreement, respectively (Figure 6). One can observe the positive results of the evaluation regarding the efficiency in learning the concepts of metrics.

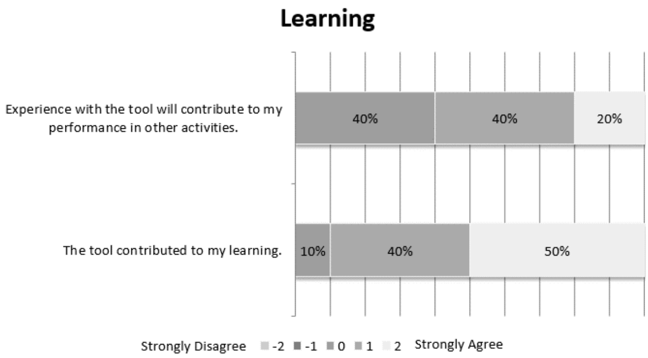


Figure 6. Evaluation of the Learning category.

Self-evaluation. A self-evaluation to assess the learning, comparing how it was before and after the tool (Figure 7). The evaluation consists of assigning a grade within a scale ranging from 1 (little) to 5 (very) the level of knowledge of the concepts presented: Coupling, Cohesion, Main Class, Attributes and Methods.

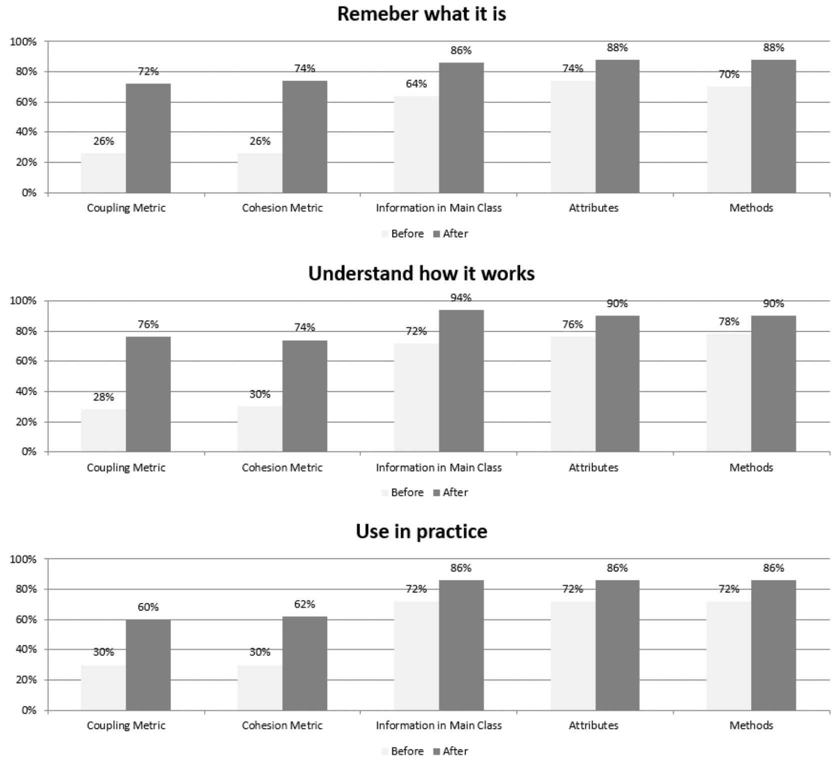


Figure 7. Self-evaluation measures on learning objectives.

It concerns three perspectives related to the concepts: (a) remember what it is; (b) understand how it works; (c) use in practice. Some comments such as "it is clear how to calculate the metrics" (participant 3) show the contribution of the tool. They had a positive result in the tasks (73% accuracy) and, in the self-evaluation, all reported some improvement. Figure 7 shows an increase in the concepts presented (gains of 60%).

4.4 Threats to Validity

It is important to highlight some threats to validity [Wohlin *et al.* 2012]. For conclusion validity, the sample size, chosen for convenience, and the limited experience in relation to the concepts are identified. As mitigation for the later, the instructions were in video, avoiding training differences. From the qualitative perspective, the participants' comments are worthy, revealing improvement opportunities. Despite that, they are opinions, a limitation on external validity, being not possible to generalize the results.

5. Conclusion

The MetricRA has aims to help in the understanding of Cohesion and Coupling metrics. To this end, the tool uses AR technology, allowing changing diagram structure, in real time, so the student can control and internalize the concepts. An observational study was performed with 10 participants. The results were positive with 82% of the tasks performed correctly and an increase of the level of knowledge, indicated by a self-evaluation performed by the students themselves. From this study and the presented results, we can observe that some objectives of the tool were achieved (help students internalize the concepts and provide an exploratory learning environment). This is an initial effort to evaluate the tool in relation to its motivation, user experience and learning aid, from the point of view of undergraduate students. As future activities we intend to complement it and improve the data significance by comparing gain aspects using a baseline scenario in comparison with the tool. For this paper, the focus was on the tool and the metrics proposed, not being addressed the perspective of structural complexity, an option for future work.

References

Basili, V. R., Caldeira, G., & Rombach, H. D. (1994). Goal Question Metric Paradigm.

Caserta, P., & Zendra, O. (2011). Visualization of the static aspects of software: A survey. *IEEE Transactions on Visualization and Computer Graphics*, *17*, 913–933.

Chidamber, S. R., & Kemerer, C. F. (1992). a Metrics Suite for Object Oriented Designa Metrics Suite for Object Oriented Design. *PhD Proposal*, *1*(6), 476–493.

Churcher, N., Irwin, W., & Kriz, R. (2003). Visualising Class Cohesion with Virtual Worlds. In *INVIS.AU* (Vol. 24, pp. 89–97).

Fenton, N. E., & Pfleeger, S. L. (1997). *Software Metrics: A Rigorous and Practical Approach*. *It Professional* (Vol. 2). International Thomson Publishing.

IEEE. (1990). IEEE Standard Glossary of Software Engineering Terminology (IEEE Std 610.12-1990). *IEEE Computer Society*.

Jin, Q., & Yano, Y. (1997). Design issues and experiences from having lessons in text-based social virtual reality environments. *Ieee SMC*, *2*, 1418–1423.

Keller, J. M. (1987). Development and use of the ARCS model of motivational design. *Journal of Instructional Development*, *10*(1932), 2–10.

Kirkpatrick, D. L., & Kirkpatrick, J. D. (2009). Evaluating: part of a ten-step process. *Evaluating Training Programs*, 3–20.

Martins, V. F., Paulo, S., & Correa, A. G. (2013). Usability Test for Augmented Reality Applications.

Pressman, R. S. . B. R. M. (2014). *Software Engineering A Practitioner's approach Eighth Edition*. Palgrave Macmillan.

Rodrigues, C. S. C., Werner, C. M. L., & Landau, L. (2016). VisAr3D. In *38th ICSE '16* (pp. 451–460). New York, New York, USA: ACM Press.

Savi, R., Wangenheim, C. G. von, & Borgatto, A. F. (2011). A Model for the Evaluation of Educational Games for Teaching Software Engineering. In *SBES* (pp. 194–203).

Titang, S., & Dur, A. (2015). The Visualization of Software Quality Metrics Bachelor of Science Thesis in Software Engineering and Management Dur Abuzaid.

Von Wangenheim, C. G., Savi, R., & Borgatto, A. F. (2012). DELIVER! - An educational game for teaching Earned Value Management in computing courses. *IST*, *54*(3), 286–298.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., & Wesslén, A. (2012). *Experimentation in software engineering*. Springer Science & Business Media.

Wu, H. K., Lee, S. W. Y., Chang, H. Y., & Liang, J. C. (2013). Current status, opportunities and challenges of augmented reality in education. *C&E*, *62*, 41–49.