

Article

Not peer-reviewed version

Competence measurement of software projects in the degree course in Computer Engineering in Matanzas

[Walfredo González Hernández](#) *

Posted Date: 20 September 2024

doi: 10.20944/preprints202409.1625.v1

Keywords: competence, measurement in software projects, training of a computer engineer, competence training, higher education



Preprints.org is a free multidiscipline platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This is an open access article distributed under the Creative Commons Attribution License which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Article

Competence Measurement of Software Projects in the Degree Course in Computer Engineering in Matanzas

Walfredo González Hernández

Universidad Central "Marta Abreu" de las Villas, Santa Clara, Cuba; wghernandez@uclv.cu

Abstract: Measurement in software projects is one of the activities that define engineering. In this article, it is defined as the competence of the computer engineer and its dimensions and indicators are determined. Subsequently, a pre-experiment is carried out where different empirical methods are applied in three moments to monitor the development of the competence.

Keywords: competence; measurement in software projects; training of a computer engineer; competence training; higher education

Introduction

The training of a computer engineer is considered of a strategic nature due to the importance of the development of computerisation processes in organisations in today's world. Hence, there is a concern to determine the components of their preparation. There are several criteria about the activity of a computer engineer in terms of competences (González-Hernández, 2016; González Hernández, 2019) and it is assumed by a group of authors that they are made up of a system of knowledge, a system of actions, the values that sustain it and the metacognition that regulates its performance (Llerena-Ocaña and González-Hernández, 2017; Molina Hernández *et al*, 2021) [3,4].

The curriculum design of a computer engineer is a complex process in which three fundamental elements are in tension: the advances in computer science, the needs of the employing organisations and the aspirations of the students. For the most part these tensions converge towards the training of a professional who can play any role during a software development process and the measurement processes. One set of measurements allows decisions to be made about the outcome of a process such as quality, while others obtain anticipated measurements about the project such as estimations. Hence the importance of determining the structure of the measurement competence in software projects, as well as the ways for its development.

Measurement in Software Projects Competence

Competences

The determination of competences during the training of a professional must take into account training objectives that are geared towards meeting the needs of the employing organisations. A study carried out by Accenture (2007) shows that there may be divergences between what the university considers teaching and what organisations need. The tension derived from the divergence of the previous sentence and taking into account student aspirations allows the formulation of the training objective of the professional that leads to the conformation of the system of subjects that make up part of the professional's curriculum. The subjects, in the case of Cuban training, have a general objective that responds to the career objective.

Professional competences, as in the case of this research, have an initial training period in which skills are developed and basic knowledge is acquired. Once this process is completed, the

professional competence is developed during a period of undergraduate training, and from there, those directly related to it are developed on the job.

The competence to measure in software projects is one of those that will be directly related to the employment of an IT professional in any of the roles. Whether analyst, designer, programmer or tester, professionals in this science need to measure certain parameters in order to achieve the quality of the product of their activity. This is why it is important for computer engineering students to develop this competence.

Measurement in Software Projects Competence

An organizational computerization project poses certain problems to the development team and most of them are related to the quality of the process and the product (Ahmed Ismaeel and Nasser Al-Mulhim, 2021; Sunil Deshpande *et al*, 2020). Within the processes of computerisation of organisations, development technology plays an essential role and object orientation is a paradigm that has come to prevail. For Misiko (2020) there is a set of metrics that can be used in design as well as in object-oriented analysis and programming that would be interesting to assume as part of the computer engineer's curriculum. Another important process for measurement in these professionals is the estimation of projects and for this the paradigm does not play an essential role. Estimation as a process is a future-oriented measurement in which error plays an essential role. There are several models that can be used for project estimation and three are currently the most widely used: function points, COCOMO and SLIM (Abdelali *et al*, 2018).

This competence begins its formation during the study of mathematics that computer engineering students receive during their first year, which includes Mathematical Analysis and Algebra. During this same year they receive the subjects Introduction to Programming and Object-Oriented Programming. In them they calculate the order of the algorithms they program by facing their first measurement and use some metrics such as inheritance tree height and class complexity, although it is not explicitly stated as a measurement. Later, in the second year, they begin to study Data Structures where they study in depth the different orders that algorithms can have and how to calculate them, which continues the training process. From this second year until the fourth year, they study Software Engineering, where they study in depth the different development methodologies. In these subjects, point estimation of use cases and algorithmic complexity are very briefly addressed as examples of measurement processes. Up to this point, the initial training of the competence to measure in software projects is considered as the initial training, as students study all the necessary knowledge and skills that allow them to understand the measurement processes.

The training process closes with a subject that is taken in the fifth year of the course called Measuring in software projects. This subject is allocated a total of 50 hours and covers three fundamental topics. The first topic is dedicated to the formalisation of the concepts of measurement, measurement and metrics. A second topic is devoted to the metrics of object-oriented analysis and design, while the third topic is devoted to project estimation using Function Points and COCOMO II. The course is taught using project-based learning (González-Hernández, 2013; Härkki *et al*, 2021). Each of the contents studied is applied in their final degree project, which culminates in the presentation of software resulting from a process of computerisation of organizations in the territory.

Therefore, the competence to measure in software projects is a configuration of knowledge about measurement, measurement skills, associated values and the necessary elements of self-regulation oriented to the performance of the role in an efficient and effective way in a computerisation process in order to achieve a quality product.

From what has been stated so far, it is assumed that the structure of the competence to measure in software projects is as follows:

Cognitive:

- a) Broad mastery of the concepts associated with metrics in relation to the role they occupy.
- b) Mastered the representation of metrics using the appropriate symbol system.
- c) Has a good command of the meaning of the results of the metrics applied depending on the role that corresponds to student.

- d) Has mastered the metrics corresponding to their role in the computerisation processes.
- e) Masters the actions comprised in the models according to the conception of those in charge of these processes.

Abilities:

- a) Selects the software entities to be measured.
- b) Determines the appropriate metrics according to the selected entities.
- c) Executes each of the selected metrics.
- d) Correctly interprets the results obtained.
- e) Compares the results obtained with other similar projects.
- f) Argues the decision taken depending on the results obtained in the metrics and the decisions taken in other projects with similar results.

Self-regulators

- a) Design strategies for the implementation of metrics in the computerisation processes in which they participate.
- b) Establish feedback mechanisms for the actions implemented in the measurement process.
- c) Establishes goals for the execution of the actions contained in the measurement of the computerisation processes.
- d) Regulates the processes to be executed in order to detect the most appropriate metrics to represent the computerisation processes.

Values

- a) Assumes with responsibility the selection of the metrics to be used in the computerisation processes.
- b) Assumes with honesty the mistakes made during the measurement process in computerisation processes in order to eliminate them.
- c) Encourages the creation of a climate of dialogue and controversy that stimulates timely criticism.
- d) Institutionalises error as a learning opportunity for the entire systems development team.
- e) It demands compliance with agreed deadlines for the timely delivery of measurements and their appropriate representation.
- f) Assume measurement as an important function in each role in its social relationship with the rest of the team members.
- g) Assume as part of their performance the ethical dilemmas derived from the responsibility of the role, they occupy in the software development processes.

The configuration represents the way in which a complex system is restructured by establishing informational flows with other systems, so assuming competence as a configuration is a significant step in the understanding of the articulation of the elements proposed in the psyche of the individual. This means that the subject organises these components depending on the informatisation process to be carried out in tension with the role they play. Hence, each subject expresses their competence in a unique and unrepeatable way, and the demands of the computerisation project play an essential role in this. These considerations mean that the competence analysis process is unique and unrepeatable for each individual, and it is necessary to determine whether each student possesses these indicators.

Methods

This research is carried out during the year 2021 in the second semester of the fifth year of computer engineering. In this year there is only one group, so it is not possible to carry out a quasi-experiment and a pre-experimental is carried out to assess the formation of the competence under study. The following research methods are used:

- Student survey: the survey makes it possible to determine the students' assessment of the presence or absence of the indicators in them, as well as the way in which they are integrated.
- Teacher interview: this method yields the teachers' assessment of the presence or absence of the indicators that make up the configuration.
- Classroom observation: its application offers the researcher the opportunity to verify the presence of the indicators by analysing students' behaviour.

- Data triangulation: this method makes it possible to contrast the information obtained by the three previous methods, which allows us to obtain assessments with a higher level of veracity about the formation of the competence.

The methods are applied in 3 breaks during the teaching of the subject in the first semester of the 2021 academic year in the fifth-year computer engineering group. This frequency is important as the introduction of a hybrid training model leads to increased control over the development of educational processes. This makes it possible to establish short-term actions to ensure that students appropriate the necessary elements for the development of competence.

Teaching this year has been marked by the transition from covid 19 and the forms of organisation have been modified from a predominantly face-to-face form to a distance one. The course architecture has been supported by a media system. A repository where all the study materials were stored was set up at <http://repositorio.umcc.cu>, a virtual teaching and learning space that promoted synchronous and asynchronous interaction between student, group and teachers at <https://eva.umcc.cu> and WhatsApp groups for more immediate communication. The first two are accessed free of charge from students' mobile phones due to an agreement between the university and a telecommunications company mediated by the government.

Results and Discussion

The first cut is made at the end of the first month of teaching with the students at the end of the first subject dedicated to measurement. For this first moment, the students are placed in the role of designers since they have already received the subjects of requirements engineering, systems analysis in previous semesters and software design in this semester. The results of the triangulation are shown in Figure 1. However, it can be seen that the students consider that they do not have a mastery of the knowledge associated with metrics, although the teachers and the observation indicate that two students do. This is explained by the fact that they already know metrics from previous subjects such as programming, but they have not mastered the appropriate system of symbols for their representation. It is important to take into account that four of the students are considered to have mastered the actions included in the metrics, which could denote a predominance of the executing function of thought. This happens when the executing function prevails over the rest of the assessment and analysis functions.

The results of the cognitive indicators indicate that the initial formation of competence has not taken place in the desired manner. There is a rejection of Mathematics and its contents in the group, which has a negative impact on any computer content that uses it as a basis. This group has a representation of Mathematics as a subject made only for geniuses and expects more complex and difficult knowledge to be studied in relation to this subject due to its strong mathematical basis. This representation must be dismantled through the use of simple and easy-to-understand metrics that, at the same time, are useful for making decisions to validate object-oriented designs. The metrics proposed by the authors Sunil Deshpande et al. (2020) in the theoretical framework and in the results of their article have the simplicity and completeness necessary for the purpose of the previous sentence.

In the case of skills, the worst performers are those related to experience and the level of knowledge about the metrics, as it is not possible to select the right metrics and argue the decision if one has a poor command of the metrics. Likewise, it is not possible to interpret the results of the metrics without knowledge of what they measure.

For the self-regulatory processes, it can be seen that 4 of the students overvalue the processes of designing strategies with clearly identified goals and regulating the most appropriate processes for selecting metrics. Meanwhile, the teachers are critical of the students' performance, considering that none of the students achieve the indicator related to goal setting. The observation shows that there are 3 students with the presence of indicators related to self-regulation.

A situation similar to the three previous dimensions is presented by the dimension related to values in which a curious situation is detected. The students do not consider that they assume responsibility for the selection of the metrics, which is due to the lack of knowledge that is evident in

the indicators of the cognitive dimension, although the teachers consider that one student achieves this, and in the observation, the achievement of two students is evident. Two of the students consider that they assume the ethical dilemmas related to the processes of requirements and software design.

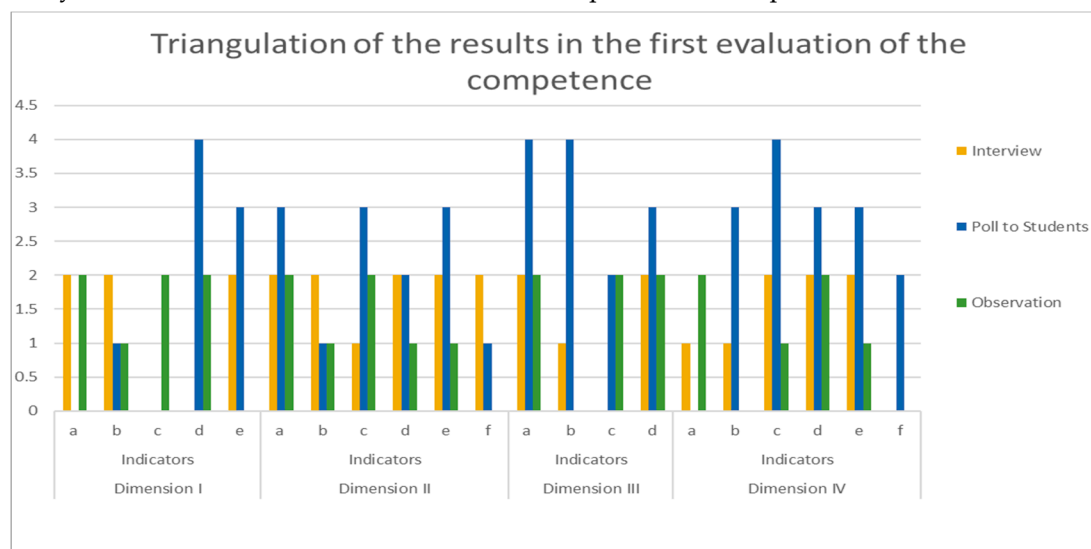


Figure 1. Results of the first triangulation. Source: author's elaboration.

The following strengths and weaknesses can be detected from the results of this cut:

Strengths:

- The teachers are committed to the training of students and guarantee the materials to achieve this.
- There is a technological infrastructure to support student access to resources in a repository and virtual space.
- There are communication channels to improve the infrastructure and alternative access routes for students with difficulties.

Weaknesses:

- There is weak initial training for students, which complicates the processes that should occur at this stage.
- Learning a subject with a strong mathematical component is made more complex by the students' representation of the subject as difficult, complicated and made only for geniuses.
- Emphasis has been placed on the assessment of students as a process that prioritises the use of the contents to apply them to problems and not to make decisions regarding engineering processes. This emphasis makes the subject more complex, as its content is precisely for decision making in terms of the quality of the processes carried out.

The results of the diagnosis are presented to the Collective of the Year made up of the teachers of all the subjects taught in that semester and the representatives of the student organisations. This is the governing body that focuses the most important methodological decisions of the year and coordinates the teachers' efforts to organise student learning. Once the results of the diagnosis have been discussed, it is decided to integrate the subjects of the year and to address integrative problems that are tackled from the different subjects. In the same way, the final evaluations of the subjects are carried out in an integrated manner, which allows the students to prepare for the exercise of their profession. The problems they will encounter in working life do not normally depend on a single subject, but must be solved by applying the competences acquired during their undergraduate studies.

Immediately after the first cut-off the students are told that their final assessment will be a measurement project applied to their diploma work. The diploma project is aimed at solving problems related to the computerisation of the territory where they live and involves community actors as potential clients. This indication states that the project approach will be used in the classes and each of them will apply the knowledge studied in class on a real development project that will

allow them to graduate as computer engineers. They must apply the metrics of object-oriented design and estimate the time, cost and effort for the development of the system. In the case of estimation, being one of the last contents to be studied, they will be applied post-development and it could be the case that they are calculated together with the implementation of their system. If this is the case, then the students compare the results obtained using function point estimation and COCOMO 2000 with the time actually taken. This allows them to assess the margins of error in these estimation processes or to correct errors in the calculations.

The second cut shows the results of Figure 2 and demonstrates that the use of simple metrics in their mathematical formulation but applicable to object-oriented design was not an unwise decision. The number of students who mastered the concepts of measurement, metrics and measurement in the observation increased by 6, while teachers and students agree that only 4 students mastered these concepts. This is evidence of the students' misgivings about the solidity and validity of the mathematical knowledge they possess, as well as of the teachers' strong demand that they demonstrate their knowledge. However, the students have a better understanding of the metrics of object-oriented design, as well as of the results obtained, and this is highlighted by the teachers in indicator c and d. Observation corroborates this data with slightly lower results in terms of the number of students who demonstrate the indicators.

In the second dimension, there is a lower number of students with the presence of the indicators, which shows that the skills depend on the knowledge system. The observation corroborates the students' assessment in indicators a, b and c, demonstrating the formation of the simpler skills, while in the case of the rest of the skills it corroborates the teachers' assessment in the case of the more complex skills. This is due to the fact that the students successfully cope with those skills that require a higher level of reproduction, while they fail in the skills that require intellectual production, as is the case with d and f. Nevertheless, these results show a substantial improvement in the students who present the skills necessary for the formation of competence.

The third dimension is the one with the lowest level of development and it is predictable because it depends on a higher degree of development of the two dimensions that precede it, especially the first one, as evidenced in other studies (Chen, 2020; Jaramillo *et al*, 2017). Although it has not been determined in the studies reviewed, the results obtained seem to indicate that values also intervene in the self-regulatory processes, so that competence is formed by first achieving the acquisition of content, then the development of skills, followed by values and finally the processes of self-regulation. This is explained by the assumed indicators themselves. It is not possible to establish feedback mechanisms if mistakes are not honestly assumed, or if a climate of dialogue is not created, or if ethical dilemmas are not assumed as part of their actions. This inference is possible after the analysis of two cuts and the observation of the students during the execution of a project.

The fourth dimension makes it possible to assess the development of the value of responsibility in more students and is due to the introduction of the project to be solved by each of them. Since the introduction of project-based learning as a teaching approach (González Hernández, 2021; Hujjatusnaini *et al*, 2022), a greater development of the values associated with responsibility for the software project to be discussed has become evident, as well as an increased demand for respecting the deadlines for revising the system delivered in each iteration.

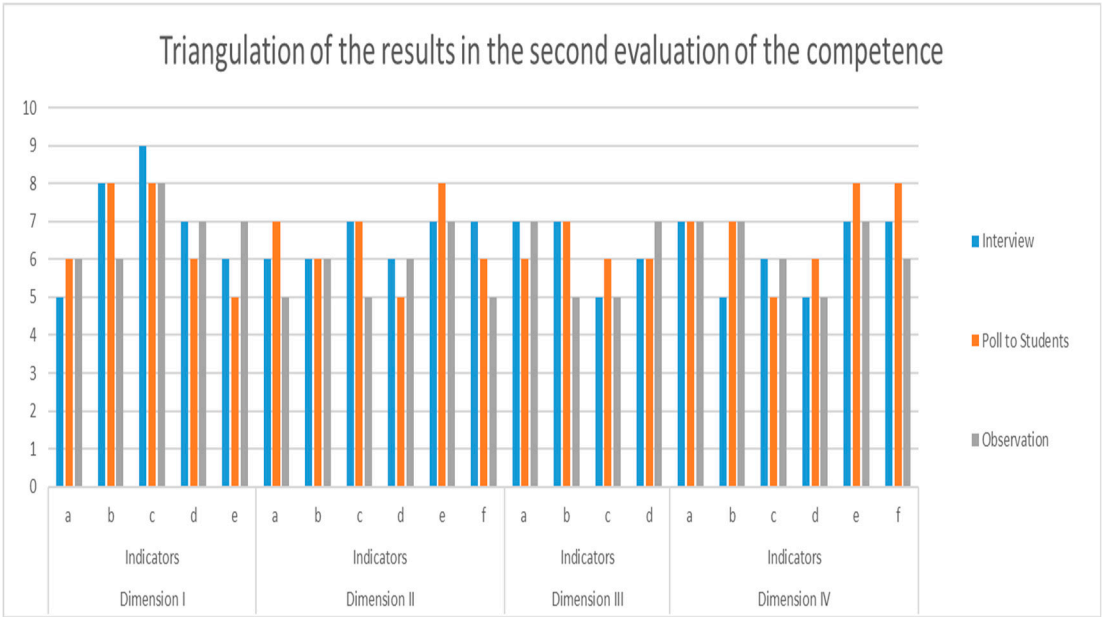


Figure 2. Results of the second triangulation. Source: author’s elaboration.

The following strengths and weaknesses can be detected from the results of this cut:

Strengths:

- The strengths detected in the initial diagnosis are maintained.
- All students have a thesis topic related to software development for computerisation processes in organisations.
- The linking of the subject project to the diploma work, which guarantees organisations a reliable estimate of time, cost and development effort with which they can work and make decisions.
- The integration of the subjects in order to respond to interdisciplinary processes in the resolution of problems linked to their thesis where they apply the knowledge of the year's subjects.
- Unity of criteria of the teaching staff.

Weaknesses:

- The representation of Mathematics as a complex and difficult subject to learn is maintained, although they recognise that metrics are not among the most complex mathematical representations they have learnt.

The third cut shows the results of Figure 3. The use of the project as a teaching approach has been a dynamic element for the development of competence as shown in the results of the last cut-off 5 months after the start of the course and three months after having applied this approach. More than 80% of the students have developed the indicators, which indicates the formation of the competence to measure in software projects. In most of the indicators, similarity of the results obtained in the three methods is observed, which indicates that the students' self-assessment of their development is more adequate, while the teachers' demands have not diminished. There is even a tendency for the students' evaluation of their performance to be more rigorous than that of the teachers. This is corroborated by the observation results and argues for the development of criticism and self-criticism so necessary in a project environment. The need for good practice workshops and comparisons between student projects is evident from the low results of indicator e of the second dimension.

Work has continued on the treatment of the mathematical content that underpins the measurement processes. Each moment of understanding of the mathematical representations of the measurement processes is accompanied by a positive emotional expression on the part of the teacher who gratifies the achievement obtained. In the same way, every moment in which they do not understand the mathematical processes that take place in the measurements is helped by reaffirming that this is a normal process in all learners. Mistakes made are treated in a similar way, where the positive aspects of the error are reaffirmed and the negative aspects are minimised, which eliminates

the students' fear of making mistakes and socialising errors. All students can help each other to correct mistakes and a climate of trust and security is built up, which is essential in cooperative work when integrating development teams.

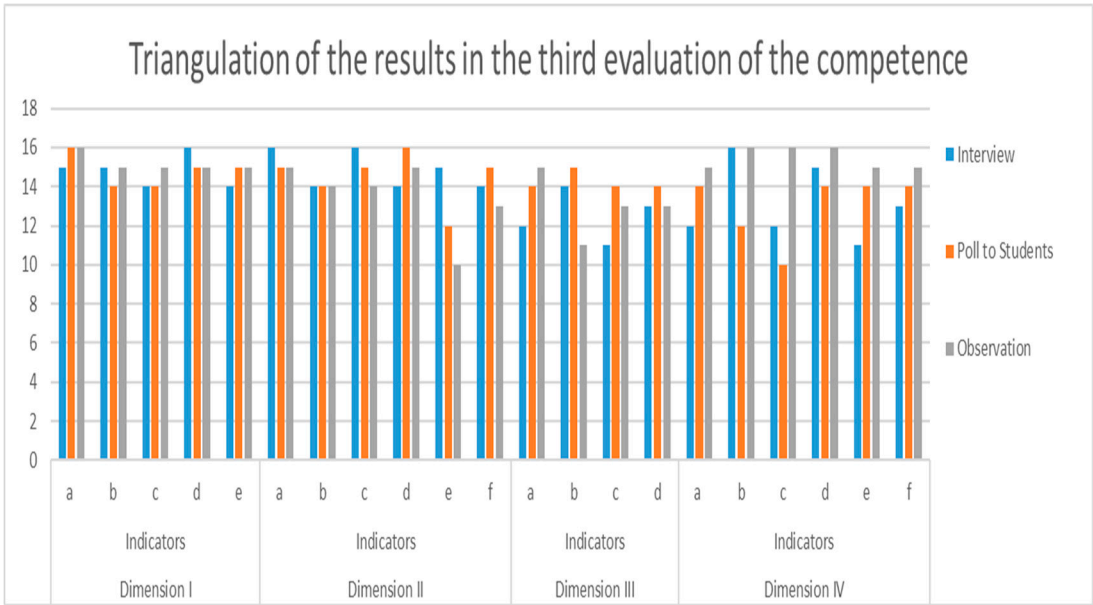


Figure 3. Results of the third evaluation. Source: author’s elaboration.

The following strengths and weaknesses can be detected from the results of this cut:

Strengths:

- Students like the subject and reflect in different spaces the importance it has for their professional future.
- The faculty has uniform criteria for the evaluation of integrative problems in order to award a grade to students.
- The forms of teamwork are developed, which is essential during their integration into development organisations.

Weaknesses:

- Non-existence of a strategy of post-graduate improvement of attention to competence that allows for a deepening of measurement processes once placed in an organisation that computerises processes.

Once all the results obtained during the execution of the pre-experimental have been analysed, the following general results are obtained:

- The computer project is the activity par excellence for the formation of the competences of a computer engineer as has been corroborated in this study and is demonstrated by previous studies with other related specialities (García-Llorente *et al*, 2020; Oliveira and Oliveira, 2019).
- Competence develops in stages that go from initial training and undergraduate training to its development in the workplace, and in each of these stages it goes from its simplest elements: knowledge to more complex ones such as self-regulation processes. In the intermediate stage, skills and value processes are formed. Each piece of knowledge that the student appropriates develops the necessary skills to operate with it, and from appropriation to its application, values and self-regulation are developed. From the simplest knowledge, skills, values and self-regulation, the foundations are laid for those of greater complexity in an ever-increasing spiral.
- Computer competences are formed through the integrated work of the subjects that make up the year and allow students to appropriate them through inter- and multidisciplinary work. In this way assessment through integrative projects takes on a formative character.

The study allows understanding the processes of formation of the competence to measure in software projects and the conditions that must be guaranteed for its achievement. This article also demonstrates that the competence studied requires a high level of integration of the IT contents of

the degree course, as it must include those related to Mathematics, Software Engineering and Programming among others.

Conclusions

The competence to measure in a software project is made up of a system of knowledge, skills, values and self-regulation elements related to the activity to be carried out. Indicators were obtained for each of its dimensions that enable it to be characterised and evaluated.

The joint work of teaching collectives to present projects and integrative problems whose solution requires the application of the contents of each subject is one of the ways in which competence is formed. Another of the ways is the introduction of the project approach that allows students to face real projects in which they must assume different roles and prove that measurement is an inherent process of engineering.

References

1. Abdelali, Z., Mustapha, H., & Abdelwahed, N. (2018). *Investigating the use of random forest in software effort estimation*. Paper presented at the Second International Conference on Intelligent Computing in Data Sciences (ICDS 2018).
2. Accenture, Y. U. (2007). Las competencias profesionales en los titulados. Contraste y diálogo Universidad-Empresa. (1-16). http://www.accenture.com/SiteCollectionDocuments/Local_Spain/PDF/Accenture-Resumen-Competencias-profesionales-enlos-titulados.pdf
3. Ahmed Ismaeel, D., & Nasser Al-Mulhim, E. (2021). The influence of interactive and static infographics on the academic achievement of reflective and impulsive students. *Australasian Journal of Educational Technology*, 37(1), 147-162.
4. Chen, F. (2020). Co-development of emotion regulation: shifting from self-focused to child-focused perezhivanie in everyday parent-toddler dramatic collisions. *Early Child Development and Care*, 1-14. doi:10.1080/03004430.2020.1762586
5. García-Llorente, H. J., Martínez-Abad, F., & Rodríguez-Conde, M. J. (2020). Evaluación de la competencia informacional observada y autopercebida en estudiantes de educación secundaria obligatoria en una región española de alto rendimiento PISA. *Revista Electrónica Educare*, 24(1), 1-17. doi:10.15359/ree.24-1.2
6. González-Hernández, W. (2013). Creativity Development in Informatics Teaching Using the Project Focus. *International Journal of Engineering Pedagogy*, 3(1), 63-70. doi:10.3991/ijep.v3i1.2342
7. González-Hernández, W. (2016). La implementación de procesos de informatización en organizaciones como competencia en la formación del profesional informático. *e-Ciencias de la Información*, 6(2), 3-19. doi:10.15517/eci.v6i2.25276
8. González Hernández, W. (2019). La competencia depurar sistemas informáticos en la Carrera de Ingeniería Informática de la Universidad de Matanzas. *ReiDoCrea*, 8(11), 119-132.
9. González Hernández, W. (2021). Los espacios de aprendizaje y las formas de organización de la enseñanza: una caracterización desde la subjetividad. *Revista de Estudios y Experiencias en Educación (REXE)*, 20(42), 17-27. doi:10.21703/rexe.20212042gonzalez18
10. Härkki, T., Vartiainen, H., Seitamaa-Hakkarainen, P., & Hakkarainen, K. (2021). Research paper Co-teaching in non-linear projects: A contextualised model of co-teaching to support educational change. *Teaching and Teacher Education*, 97, 103188 doi:10.1016/j.tate.2020.103188
11. Hujjatusnaini, N., Corebima, A. D., Prawiro, S. R., & Gofur, A. (2022). The effect of blended project-based learning integrated with 21st-century skills on pre-service biology teachers' higher-order thinking skills. *Jurnal Pendidikan IPA Indonesia*, 11(1), 104-118. doi:10.15294/jpii.v11i1.27148
12. Llerena-Ocaña, L.-A., & González-Hernández, W. (2017). La competencia desarrollar sistemas web en la formación de los profesionales informáticos: una aproximación a su estudio. *ReiDoCrea*, 6(2), 229-245.
13. M. Jaramillo, J., Rendón, M. I., Trommsdorff, G., Muñoz, L., & Weis, M. (2017). Children's Self-Regulation in Cultural Contexts: The Role of Parental Socialization Theories, Goals, and Practices. *Frontiers in Psychology*, 8, 1-9. doi:10.3389/fpsyg.2017.00923
14. Misiko, N. (2020). A Review of Metrics for Object-Oriented Design. *Global Journal of Computer Science and Technology: C Software & Data Engineering*, 20(2), 1-6.
15. Molina Hernández, C. R., González Hernández, W., & Cruz Lemus, G. (2021). Habilidad modelar procesos dinámicos de control automático. *Educación Química*, 32(1), 100-111. doi:10.22201/fq.18708404e.2021.1.75429

16. Oliveira, D. S. d., & Oliveira, N. R. C. d. (2019). Competência em Informação: mapeamento do uso de fontes de informação por discentes da área da saúde. *Transinformação*, 31. doi:10.1590/2318-0889201931e170074
17. Sunil Deshpande, B., Kumar, B., & Kumar, A. (2020). Object Oriented Design Metrics for Software Defect Prediction: An Empirical Study. *Test Engineering & Management*, March - April, 10501 - 10518.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.