

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>



Original article

Models, simulators and Software Engineering

Modelos, simuladores e Ingeniería del Software

Modelos, simuladores e Engenharia de Software

Sonia Itati Mariño¹



<https://orcid.org/0000-0003-3529-7003>

Pedro Luis Alfonzo¹



<https://orcid.org/0000-0001-5447-8518>

¹National University of the Northeast.
Argentina.



simarinio@yahoo.com;
plalfonzo@hotmail.com

Received: July 14, 2023

Accepted: December 6, 2024

ABSTRACT

An interdisciplinary approach is presented, linking models, simulators and software engineering. In particular from the Software Engineering are addressed and adapted agility issues and the SWEBOK knowledge guide. The methodology, for the achievement of the proposal deals with the exploration of literature and an adaptation of SCRUM to assist in the

management of educational projects and contribute to the teaching-learning process. In addition, the practices developed in a degree subject are explicit with the aim of achieving meaningful learning, focused on students oriented to project-based learning. It is concluded that the SWEBOK guide provides areas of knowledge that are addressed from the modelling and simulation of systems, contributing to significant learning through the appropriation and integration of disciplinary knowledge. In the future, other areas of knowledge in Software Engineering that would be linked to the proposed approach in the subject will be identified and analyzed focusing on the profile of students.

Keywords: SCRUM; higher education; models; simulators; Software Engineering.

RESUMEN

Se presenta un enfoque interdisciplinario que relaciona los modelos, los simuladores y la Ingeniería del Software. En particular desde la Ingeniería del Software se abordan y adaptan cuestiones de agilidad y la guía de conocimiento SWEBOK. La metodología, para el logro de la propuesta trata la exploración de la literatura y una adaptación de SCRUM para asistir en la gestión de proyectos educativos y contribuir al proceso de enseñanza-aprendizaje. Además, se explicita las prácticas desarrolladas en una asignatura de grado con la finalidad de lograr aprendizajes significativos, centrados en los estudiantes orientados al aprendizaje basado en proyectos. Se concluye que la guía SWEBOK proporciona áreas de conocimiento que son tratadas desde la modelización y simulación de sistemas, contribuyendo con aprendizajes significativos a través de la apropiación e integración de conocimientos disciplinares. A futuro, se identificarán y analizarán otras áreas de conocimiento de la Ingeniería del Software que se vincularían con el abordaje propuesto en la asignatura centrándose en el perfil de los estudiantes.

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

Palabras clave: SCRUM; educación superior; modelos; simuladores; Ingeniería del Software.

RESUMO

É apresentada uma abordagem interdisciplinar que relaciona modelos, simuladores e Engenharia de Software. Principalmente na Engenharia de Software, questões de agilidade e o guia de conhecimento SWEBOK são abordados e adaptados. A metodologia, para concretização da proposta, trata da exploração de literatura e de uma adaptação do SCRUM para auxiliar na gestão de projetos educacionais e contribuir no processo de ensino-aprendizagem. Além disso, são explicadas as práticas desenvolvidas em uma disciplina de graduação com o objetivo de alcançar uma aprendizagem significativa, voltada para alunos orientados para a aprendizagem baseada em projetos. Conclui-se que o guia SWEBOK disponibiliza áreas de conhecimento que são tratadas a partir da modelagem e simulação de sistemas, contribuindo com uma aprendizagem significativa através da apropriação e integração de conhecimentos disciplinares. Futuramente serão identificadas e analisadas outras áreas do conhecimento da Engenharia de Software que estariam vinculadas à abordagem proposta na disciplina, com foco no perfil dos alunos.

Palavras-chave: SCRUM; ensino superior; modelos; simuladores; Engenharia de Software.

INTRODUCTION

The complexity and emergence of various phenomena in today's world highlight the need to model abstractions of reality for further experimentation by means of software simulators. Simulation models are used in a wide variety of contexts, including commerce, industry, engineering, production, transport and academia, among others. The construction of

these simulators, representative of the models conceived, uses areas of knowledge from software engineering (IS), one of the disciplines of informatics.

Software Engineering and the SWEBOK Guide

Software Engineering is an engineering discipline oriented towards the production of software. It covers all stages of the process, from the initial stages of specification of the system to its maintenance once in use. Professional associations such as the IEEE Computer Society and the Association for Computing Machinery joined forces to develop the Software Engineering Knowledge Body guide, known as SWEBOK (Software Engineering Body of Knowledge This document is considered a fundamental guide to the discipline. Its features, highlighted by Bourque and Fairley (2014), include:

- Characterize the contents of Software Engineering.
- Provide access, through various topics, to the body of knowledge of Software Engineering.
- Promote a consistent vision of Software Engineering worldwide.
- Clarify the relationship between Software Engineering and other disciplines, such as Computer Science or Mathematics.
- Provide a basis for curriculum development and the creation of certification materials.

In the work of Bourque and Fairley (2014) 15 areas of knowledge are defined, which are: software requirements, software design, software construction, software testing, software maintenance, software configuration management, software engineering management, software engineering process, software engineering models and methods, software quality, software engineering professional practice, software engineering economics, computer fundamentals,

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

mathematical foundations and engineering foundations.

Other works addressing the SWEBOK guide in the field of higher education can be found in Alenezi and Akour (2022), Abran et al. (2022) and Kamthan and Washizaki (2022).

Agile project management

The software industry, in order to respond efficiently and effectively, must adopt agility in its projects. In the 15th Annual State of Agile, SCRUM was highlighted as the most popular agile approach, with 66% of respondents identifying it as their favorite. Additionally, 15% mentioned SCRUM derivations such as ScrumBan (9%) and Scrum/XP (6%) (Hernández Sola, 2021).

In higher education contexts, there is a growing trend to adopt and adapt agile methodologies. Between the most relevant contributions are the works of Vila Grau and Capuz Rizo (2021), Capuñay Uceda and Antón Pérez (2021), Arias, Tomaselli and Tortosa (2021), and Ruiz de Mendarozqueta, Bustos and Colla (2022).

A synthesis of these contributions shows the diversity of approaches in which SCRUM, as an agile methodology, is addressed from literature reviews, in teaching and learning processes, and as support in the management of educational projects. In addition, the reflective analysis in this area contributes both theoretically and methodologically to this proposal

Context

The optional subject "Models and Simulation" belongs to the Area of Computational Sciences and Methods of the Bachelor's Degree in Information Systems.

In this subject, a project is developed aimed at the articulation of teaching and applied research activities, in accordance with some of the missions of the University. In particular, the content focuses on solving abstractions of real-

world problems, with the aim of identifying and proposing alternative solutions. Likewise, the identification and systematization of experiences linked to R&D&I projects is also promoted.

To contribute to extension and applied research projects, through the transfer of didactic and specific knowledge on modelling and simulation techniques, real problems are identified which are abstracted and solved with these tools. In addition, because of the placement of the subject in the curriculum, students may be interns, fellows or work in a variety of contexts.

It should be noted that the modelling carried out encourages the generation of numerous experiments, their tabulation, graphical representation and subsequent analysis. This encourages skills such as analysis of results and argumentation, which are essential for the professional training of students.

- Highlighted activities of the subject:
- Problematization and abstraction: Real situations that can be modeled and simulated are addressed based on problems raised in the subject.
- Identification of methods and tools:
- Selection, study and evaluation of methods for modeling simulation problems.
- Identification and evaluation of software tools for generating simulators and abstract models of real problems. These tools are selected by the students, and the simulators are used as artifacts to run experiments, analyze results, and infer alternative solutions. Comparative studies could be proposed in the future.
- Building simulators: Development of software artifacts that enable experimentation.
- Verification: Confirmation of the correct implementation with respect to the conceptual model, that is, the correspondence with the specifications and acceptable assumptions according to the purpose of the model and the simulator.

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

- Experimentation: Evaluation of the performance of the simulation algorithms studied, validating the use of various programming languages.
- Knowledge articulation: Integration of topics such as probability, programming and project management. The selection of topics covered in previous subjects is encouraged, to which the simulation module is proposed to be incorporated.

In this subject, students, individually or in teams of up to 2 or 3 members, must develop an integrative work that synthesizes, through a methodology specific to the simulation of systems, an abstraction of a case study. These simulators mostly apply inventory models or queues, and relate to the active methodology known as project-based learning (Botella Nicolás y Ramos Ramos, 2020; Villanueva Morales, Ortega Sánchez and Díaz Sepúlveda, 2022; Soriano, Rosero, Guzmán y Nieves, 2024).

The article aims to present an interdisciplinary approach that relates models, simulators and Software Engineering.

MATERIALS AND METHODS

The methodological approach to achieving the results comprised the following phases.

Phase 1. Exploratory review of the literature:

The focus was on the selection and documentary analysis of articles related to agile methods and the SWEBOK knowledge guide. Following Romillo and Polaino (2019, p. 4), this phase "made it possible to identify, through the use of the analysis and synthesis method, the main elements and their interrelation, which were decisive for decision-making" in relation to the present research and development.

Phase 2. Definition and delimitation of the context in which the R&D is situated:

The subject Models and Simulation of a degree course in systems was established as the context for applied research, articulating with some lines of R&D projects accredited by the university.

Phase 3. Adaptation of the agile SCRUM methodology:

Its choice was based on the design of a strategy to systematize the production of empirical knowledge supported by theory.

Phase 4. Identification of SWEBOK knowledge areas involved in the development:

In the design and development of the research, theoretical level methods were applied, such as analytical-synthetic, in this case to identify antecedents around areas of Software Engineering and determine which ones are articulated with the activities developed in the subject, specifically the construction of software simulators.

The inductive-deductive method was chosen, with the aim of advancing from general to specific knowledge, especially to analyze and reflect on the link between some areas of knowledge of the SWEBOK guide and the design and development of simulators of models abstracted from reality to solve problems identified by academia.

At an empirical level, supported by theoretical methods, the analysis of observations and notes made during the period 2021 was used, in a context of preventive and mandatory social isolation (ASPO). During this period, various situations were addressed according to the institutional regulations in force. The simulators built within the framework of the subject were a key source of information to support the proposal.

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

RESULTS

The adequacy of the SCRUM agile methodology is described below. Figure 1 presents SCRUM roles, artifacts and practices included in the proposal.

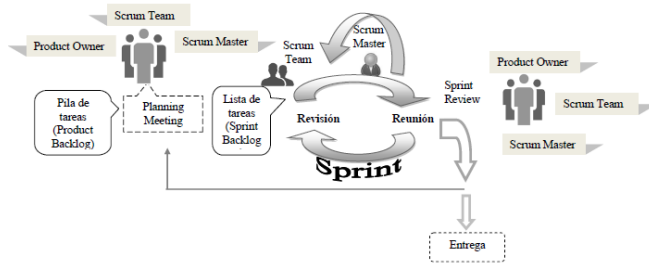


Fig. 1- SCRUM roles, artifacts and practices

Definition of roles

In a higher education context and adapting Scrum, the following roles were established:

Scrum roles

Product Owner: The owner of the product, represented by the responsible professor and the subject teachers. He defines and prioritizes the task list based on the project requirements and classroom planning to achieve the established learning outcomes.

Scrum Master: This is the project leader. He ensures that the activities are carried out and resolves problems that arise in order to achieve the teaching-learning objectives defined in the planning. This role is assumed by the teachers, who guide the execution of the tasks.

Scrum Team: This is the team responsible for executing the proposal, made up of students, under the guidance of the Scrum Master.

Scrum Practices and Artifacts

Product Backlog: Represents the requirements of each project in a list of tasks associated with the model and its simulator.

Risk management: It is carried out continuously through meetings scheduled in each class, Planning Meetings and Sprint Reviews, aimed at monitoring, tracking and feedback according to planning.

Planning Meeting: Sprint planning meeting, based on the Product Backlog. Participating in the Product Backlog Owner, the Scrum Master and the Scrum Team. These meetings take place during the classes scheduled in the schedule and may be complemented with specific tutorials or consultations.

Sprint Backlog: Contains the selected tasks from the Product Backlog that will be addressed in each Sprint.

Sprint Backlog Management: At the beginning of each iteration, requirements are selected and the effort for each task is estimated. These iterations are linked to practical and laboratory work guides that, incrementally, contribute to the development of the Final Integrating Work (TFI).

Sprint: Each Sprint is estimated to last two weeks and produces a functional module of the final work. For example: building an artificial sample, a queuing theory simulator, or an inventory simulator.

Meetings: They are carried out in person, virtually or asynchronously to ensure the achievement of objectives. The active participation of all team members is essential for feedback and continuous improvement.

Sprint Review: At the end of each Sprint, the Scrum Team presents progress through oral presentations, demonstrations of deliverables, reports, or other resources. The Product Team participates Owner, the Scrum Master and the Scrum Team.

Burn-down charts: These charts allow you to visualize and analyze the team's progress, providing a graphical representation of completed and pending tasks. This ensures that

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

deliveries are made within the established deadlines, which influences the promotion or approval of the subject.

Sprints included in the proposal

Sprint 1: Delimitation of the proposal, designing a case study adapted to the methodological strategy outlined in the "Context" sub-section.

Sprint 2: Analysis of the practical and laboratory work guides proposed by the subject, as well as the student productions of the 2021 academic year.

Sprint 3: Deepening knowledge areas in the SWEBOK guide. This analysis allowed us to identify connections between Software Engineering and the construction of software for experimentation and simulation, creating a portfolio that supports the Final Integrative Work.

Sprint 4: Determination of the SWEBOK guide knowledge areas represented in the practical and laboratory work. The simulators developed include topics related to "Mathematical Foundations", "Software Construction" and "Professional Practice of Software Engineering".

DISCUSSION

The area of knowledge of Mathematical Fundamentals (Bourque and Fairley, 2014) helps software engineers to understand logic, which is translated into code in programming language. Logic and reasoning are essential in the mathematics that a software engineer must master. In addition, techniques are addressed that allow the representation and carrying out of reasoning and judgment in a precise and therefore mathematical way. It is considered that a computer program can only be written for a given problem if a coherent logic is followed. Thus, the objective of this area of knowledge is

to facilitate the development of skills for identifying and describing such logic.

The term "software development" refers to the detailed creation of software and to the area of knowledge called Software Construction, according to the guide by Bourque and Fairley (2014). This guide presents a detailed description of how to build it. It states that building software is an activity in which the software engineer must face the constraints of the real world, which is chaotic and changing, and must do so accurately. Due to the influence of these real-world constraints, software construction is more driven by practical considerations than other areas of knowledge. Development languages include all forms of communication through which a person can specify an executable solution to a given problem. In this subject, the construction of software simulators is promoted, using the tools previously studied and, preferably, those of the students themselves.

As for the knowledge area "Professional Practice of Software Engineering" (Bourque and Fairley, 2014), it deals with the knowledge, skills and attitudes that engineers must possess in order to practice software engineering in a professional, responsible and ethical manner. Since software products have widespread applications in social and personal life, their quality can have a profound impact on our personal and social well-being. In particular, the development of simulators is a strategy to support decision-making abstracted from the context, which allows inferring possible results without directly intervening with the environment. Therefore, it is considered that activities such as this contribute to articulating the interaction between the university, the company and the context.

Although the studies by Abran et al. (2022), Alenezi and Akour (2022), and Kamthan and Washizaki (2022) mention the use of the SWEBOK guide in curricular issues, their approaches are not related to an applied mathematics subject within the curriculum of the Computer Science discipline. This fact supports

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

the differentiating approach of the present proposal. One area of knowledge of the SWEBOK guide is testing. Alenezi and Akour (2022) propose a course that contemplates the criteria and standards of SWEBOK and ABET in order to coincide with specialized topics in Software Engineering and contribute to the formation of the skills required by the industry. In the proposed subject, testing is a relevant activity, although it has not been made explicit in this proposal, it could be the subject of development in future courses.

Kamthan and Washizaki (2022) synthesize the papers presented at the International Conference on Software Engineering and Knowledge Engineering (SEKE, 2022) around the SWEBOK guide. The value of these papers lies in their ability to foster the integration of guidance into software-intensive organizations and educational institutions. In this sense, although this proposal is limited to an elective course of degree, it is considered that it contributes to the awareness of the need for interdisciplinary approaches, especially in subjects which provide alternative methods to support decision making, as the techniques of system simulation which address abstractions of reality.

In the 21st century, interdisciplinary approaches are valued in which knowledge, experiences and know-how from various disciplines contribute significantly to training. Therefore, the abstractions of modeled and simulated problems, by integrating Software Engineering topics, strengthen the link between academic and professional fields, largely focused on the Software Industry.

Following what was presented by Vila Grau and Capuz Rizo (2021), the findings of this article contribute with evidence generated in higher education spaces to the so-called "expansion of Agility and Scrum in classrooms" (p. 2276). This is another example of how to address agile project management issues in educational contexts in an interdisciplinary manner.

Another example of an interdisciplinary approach is the articulation of a systems degree from an optional subject in the area of Computational Sciences and Methods, and its relationships with areas of knowledge of Software Engineering covered in the SWEBOK guide. Specifically, work was done with problem-based learning (PBL), mentioning examples of what was discussed in Botella Nicolás and Ramos Ramos (2020). In addition, Villanueva Morales, Ortega Sánchez and Díaz Sepúlveda (2022) point out that PBL involves collaborative work, a modality that is seen in agility.

From a didactic-pedagogical approach, PBL ensures the appropriation and integration of disciplinary knowledge, evidencing new alternatives to generate knowledge oriented towards decision-making. Therefore, it is considered that articulating the aspects of the discipline in PBL strengthens the significant learning of students who are inserted in a complex working world. That is, it is an active learning (Soriano et al., 2024) that, based on the abstraction of real problems, generates experiences that enrich real and significant learning. Coinciding with Soriano et al. (2024), it is stated that students reinforce both their specific knowledge and their professional skills.

As future work, it is proposed to explore other links following what is set out in the course program. In this way, the studies and findings of works that integrate subjects in the area will be visualized and explained, as well as other areas of knowledge of the degree, such as programming, information systems, among others.

Using student-centered approaches to achieve meaningful learning, a survey will be designed to assess perceptions about the areas of knowledge of Software Engineering, focusing on the profile of the participants. In addition, as part of a continuous improvement cycle, it is proposed to develop an instrument to systematize and measure student motivation when solving real problems that reflect the perceptions of teachers.

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

ACKNOWLEDGEMENTS

Research and development included in the PI Computer systems: models, methods and tools (PI 19F014, Res. No. 1015/19 CS.) and Computer systems and knowledge management. Models, methods and tools (PI 22F025, Res. No. 0931/22 CS UNNE). Accredited by the General Secretariat of Science and Technology, National University of the Northeast.

BIBLIOGRAPHIC REFERENCES

Abran, A., Yurkov, A., Khalin, V. y Shilova, O. (2022). Quantitative analysis of informational significance of SWEBOK knowledge areas in IEEE/ACM curriculum guidelines. In *System Analysis in Engineering and Control* (pp. 561-573). Cham: Springer International Publishing. https://www.researchgate.net/publication/359998231_Quantitative_Analysis_of_Informational_Significance_of_SWEBOK_Knowledge_Areas_in_IEEEACM_Curriculum_Guidelines

Alenezi, M. y Akour, M. (2022). Methodical Software Testing Course in Higher Education. *International Journal of Engineering Pedagogy*, 12(1). https://www.researchgate.net/publication/358504537_Methodical_Software_Testing_Course_in_Higher_Education

Arias, M., Tomaselli, G. P. y Tortosa, N. (2021). Computación afectiva y agilidad en entornos virtuales de educación: una revisión sistemática de la literatura. *VIII Simposio Argentino sobre Tecnología y Sociedad (STS 2021) - JAIIO 50* (Modalidad virtual). Argentina. <https://www.sciencedirect.com/science/article/abs/pii/S0360131519302027>

Botella Nicolás, A. M. y Ramos Ramos, P. (2020). La relación con los demás y la

motivación en un Aprendizaje Basado en Proyectos. *Estudios pedagógicos (Valdivia)*, 46(1), 145-160. <https://dx.doi.org/10.4067/S0718-07052020000100145>

Bourque, P. y Fairley, R. (2014). SWEBOK Guide to the Software Engineering Body of Knowledge, Version 3.0. IEEE Computer Society. <https://www.computer.org/education/bodies-of-knowledge/software-engineering>

Capuña y Uceda, O. E. y Antón Pérez, J. M. (2021). Influencia de SCRUM en los plazos de entrega y rendimiento en los proyectos de las asignaturas de Desarrollo de Software. *Revista Iberoamericana de Tecnología en Educación y Educación en Tecnología*, no. 29, 36-42. https://www.scielo.org.ar/scielo.php?script=sci_arttext&pid=S1850-99592021000200005

Hernández Sola, G. (2021). Nuestro análisis de la agilidad en 2021 a partir del 15th Annual State of Agile Report. <https://www.Scrum.org/resources/blog/nuestro-analisis-de-la-agilidad-en-2021-partir-del-15th-annual-state-agile-report>

Kamthan, P. y Washizaki, H. (2022). Educational and Professional Implications of SWEBOK. *Proceedings of the 34th International Conference on Software Engineering and Knowledge Engineering. SEKE 2022*. KSIR Virtual Conference Center Pittsburgh, USA. <https://worldscientific.com/doi/10.1142/S0218194022500590>

Romillo, A. y Polaino, C. (2019). Aplicación del Modelo de Gestión Pirámide del Desarrollo Universitario en la Universidad de Otavalo, Ecuador. *Formación Universitaria*, 12(1), 3-12. <https://www.scielo.cl/scielo.php?script=>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>

[sci_arttext&pid=S071850062019000100003](https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599)

Ruiz de Mendarozqueta, A., Bustos, F. y Colla, P. (2022). Relationship between mature software engineering practices and agility practices. *Electronic Journal of SADIO*. 21(2). 79-109.
https://sedici.unlp.edu.ar/bitstream/handle/10915/142904/Documento_completo.pdf-PDFA.pdf?sequence=1

Soriano, K., Rosero, P., Guzmán, J. y Nieves, Z. (2024). Implementación en el aprendizaje basado en proyectos (ABP) en la educación universitaria: impacto en la motivación y el rendimiento de los estudiantes. *Revista Social Fronteriza*. 4. e45456.
<https://www.revistasocialfronteriza.com/ojs/index.php/rev/article/view/475>

Vila Grau, J. L. y Capuz-Rizo, S. (2021). La extensión de la agilidad al ámbito docente a través de Scrum & EduScrum. *In 25th International Congress on Project Management and Engineering At: Alcoi (Spain)*. 1, 2276-2290, Project: PhD research Project of Juan Luis Vila Grau.
<http://dspace.aepro.com/xmlui/handle/123456789/3047>

Villanueva Morales, C., Ortega Sánchez, G. y Díaz Sepúlveda, L. (2022). Aprendizaje Basado en Proyectos: metodología para fortalecer tres habilidades transversales. *Revista de estudios y experiencias en educación*, 21(45), 433-445.
<https://doi.org/10.21703/0718-5162.v21.n45.2022.022>

Conflict of interests:

The authors declare not to have any interest conflicts.

Authors' contribution:

The authors participated in the design and writing of the article, in the search and analysis of the information contained in the consulted bibliography.

Cite as

Mariño, S. I., Alfonzo, P. L. (2024). Models, simulators and Software Engineering. *Mendive. Journal on Education*, 22(4), e3599.
<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>



This work is licensed under a [Creative Commons Attribution-NonCommercial 4.0 International License](https://creativecommons.org/licenses/by-nc/4.0/)

<https://mendive.upr.edu.cu/index.php/MendiveUPR/article/view/3599>