International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-14 Issue-6, May 2025

Agile Methodology Integrated Programming Planned (PIP) to Optimize the Development of Educational Software

Cesar Patricio-Peralta, Jesús Zamora Mondragon, Jacker Paredes Meneses, Jimmy Ramirez Villacorta, Adan Eugenio Contreras Angeles, Carlos Alberto Garcia Cortegano

Abstract*: *The new Agile Methodology, Planned Integrated Programming (PIP), is an efficient and robust alternative for manufacturers, developers, and students of Software Engineering. The study employs an experimental design that enables the evaluation of the impact of the Agile PIP Methodology on the development of educational software. The population consisted of 5 professionals in charge of creating and maintaining Educational Software, 15 teachers and 10 students from the School of Systems Engineering of the University of San Martín de Porres. The Results: Based on chi-square tests, they demonstrate that PIP has a significant impact. With a Continuity Correction statistic value of 8.643 and a p-value of 0.003, it is concluded that PIP significantly improves the development of educational software, reducing development times, testing errors, and improving satisfaction. The application of PIP in the School of Systems Engineering at USMP proved effective, with an 80% effectiveness rate according to the scale indicators. Conclusions: The Agile Planned Integrated Programming (PIP) methodology has proven to be a valuable resource for developing educational software in the university context, achieving substantial improvements in quality, delivery times, and user satisfaction, which validates its application in other similar projects within the academic sector.

Keywords: Methodology, Agile, Flexibility, Adaptability, Educational Software, Planned Integrated Programming (PIP).

Abbreviations:
PIP: Planned Integrated Programming
XP: Extreme Programming

Manuscript received on 24 February 2025 | First Revised Manuscript received on 06 March 2025 | Second Revised Manuscript received on 18 April 2025 | Manuscript Accepted on 15 May 2025 | Manuscript published on 30 May 2025.
*Correspondence Author(s)
Cesar Patricio-Peralta*, Universidad Nacional Federico Villarreal, Lima, Perú. Email ID: c22754@utp.edu.pe, ORCID ID: 0000-0002-4419-5378.
Jesús Zamora Mondragon, Universidad César Vallejo, Lima, Perú. Email ID: jezamoraz@ucvvirtual.edu.pe, ORCID ID: 0000-0001-6362-1603.
Jacker Paredes Meneses, Universidad Nacional de la Amazonia Peruana, Iquitos, Perú. Email ID: jackerpame@gmail.com, ORCID ID: 0000-0003-4381-6379.
Jimmy Max Ramirez Villacorta, Universidad Nacional de la Amazonia Peruana, Iquitos, Perú. Email ID: jimmy.ramirez.villacorta@hotmail.com, ORCID ID: 0000-0003-2923-2270.
Adan Eugenio Contreras Angeles, Universidad Tecnológica del Perú, Lima, Perú. Email ID: adan_21_8@hotmail.com, ORCID ID: 0009-0007-9487-6239
Carlos Alberto Garcia Cortegano, Universidad Nacional de la Amazonia Peruana, Iquitos, Perú. Email ID: cagaco177@gmail.com, ORCID ID: 0000-0003-3353-9566

I. INTRODUCTION

This research enables us to design and validate the Agile PIP Methodology (Planned Integrated Programming), specifically tailored to optimise educational software development. This methodology combines elements of Agile approaches such as Scrum [1] and Extreme Programming (XP) [2], in addition to being aligned with the ISO/IEC/IEEE 29119 standard [3], which ensures quality in the development process. The PIP Methodology aims to efficiently meet the demands of the educational environment, offering an adaptable and collaborative approach. An approach that increases flexibility and efficiency in delivering educational software solutions.

This methodology is validated at the University of San Martín de Porres, an institution that, given the dynamics of its academic programs and the central role that technology plays in its pedagogical strategies, is an ideal context for implementing PIP. The research focuses on evaluating how the application of the methodology impacts software quality, user satisfaction (among students and teachers), and the reduction of development cycle times.

The central hypothesis of this study proposes that the design and validation of the Agile PIP Methodology optimizes the development of educational software, improving the integration between pedagogical models and technological solutions. Throughout the thesis, it is analyzed how the incorporation of short iterations, constant feedback, and the active participation of users in the development process contributes to the creation of software that effectively adjusts to the changing demands of the educational environment.

II. DEVELOPMENT OF THE NEW AGILE PIP METHODOLOGY

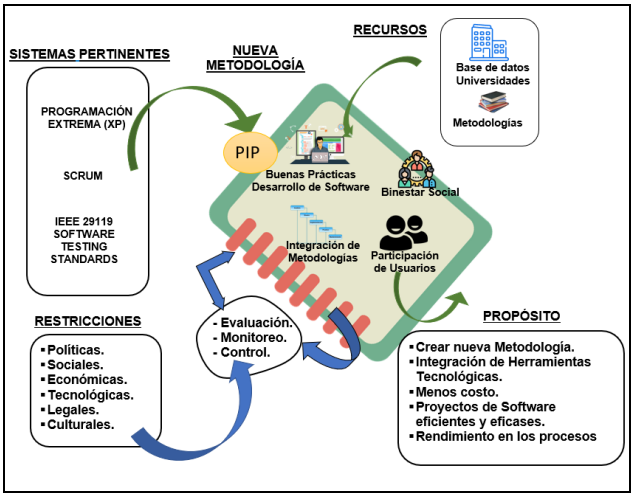
To develop the new Agile Integrated Planned Programming Methodology (PIP), a review of the available evidence from other current studies and related topics was conducted. This review analyses best practices in educational software development to design and validate the Agile PIP Methodology, aiming to optimise the educational software development process at the School of Systems Engineering of the University of San Martín de Porres.

A. Review of Existing Methodologies

The proposed Planned Integrated Programming (PIP) methodology will allow the control and monitoring of the development of each phase of the Educational Software project, applying best practices. The new method was created from the phases of Extreme Programming (XP), SCRUM, and ISO/IEC/IEEE 29119 methodologies.

B. Development of the New Planned Integrated Programming (PIP)

The New Planned Integrated Programming (PIP) proposal is an Agile approach that optimises the development of educational software at the University of San Martín de Porres, integrating principles of collaboration, flexibility, and adaptation. Based on Extreme Programming (XP), Scrum and the ISO/IEC/IEEE 29119 standards, PIP. This methodology is based on principles of agility, integrating various practices that facilitate adaptation to educational and technological needs. The phases of PIP are (Planning and Estimation, Implementation, Control and Monitoring, Launch).



[Fig.1: Pictorial Table of the Planned Integrated Programming (PIP)]

C. Details of the New Integrated Planning Programming (PIP) Methodology

For the implementation of the Agile Planned Integrated Programming (PIP) Methodology, four phases are considered, which are mentioned below:

- Phase I: Planning and Estimation: Includes defining clear objectives aligned with the University of San Martín de Porres' needs. Requirements identification is done by collecting and documenting the desired functionalities through stakeholder meetings. Establishing priorities and classifying the criteria by importance and urgency is essential. An estimate is made of the time and resources needed for development to ensure adequate planning. Additionally, a communication plan is developed to provide seamless interaction among all parties involved.
- Phase II: Implementation: Educational Software Development using the Agile PIP Methodology, divided into short iterations (sprints). The following activities are carried out: Meeting Minutes: Documentation of follow-up meetings to record decisions and agreements. Create Prototype: Development of a functional prototype that allows the application to be visualised and feedback to be obtained. Optimise Source Code: Refactoring the code to improve its quality and performance, ensuring it meets established standards. Constant feedback enables real-time adjustments, improving the quality of the software.
- Phase III: Control and Monitoring: The control and monitoring phase enables the creation of a set of activities to verify the correct implementation of the project activities outlined in the planning. The purpose is to explain the project's progress so that it can be appropriately corrected when the project implementation deviates from the schedule.

III. RESULTS OF THE INDICATORS

Below are the values obtained for each indicator in the GC Post and Ge Post Test.

Table- I: Post-Test Gc and Post-Test Ge Results

N°	I1: Development Time (Hours)		I2: Number of Errors in Tests		I3: User Satisfacción	
	Gc post-test	Ge Posttest	Gc post-test	Ge Posttest	Gc post-test	Ge Posttest
1	43	36	52	35	2	5
2	46	26	60	50	3	4
3	42	27	40	32	1	3
4	26	16	45	15	1	5
5	35	22	59	30	3	5
6	40	28	46	10	2	5
7	36	24	60	40	2	5
8	39	14	50	35	2	5
9	33	14	35	21	2	3
10	48	29	57	40	1	5
11	35	20	60	50	4	4
12	26	20	56	50	3	5
13	47	28	50	40	4	3
14	38	28	40	25	4	4
15	48	31	35	12	4	5
16	42	25	48	32	4	5
17	31	20	56	40	3	3
18	41	22	55	43	4	4
19	44	34	60	52	4	4
20	23	15	46	35	2	5
21	46	28	40	30	4	4
22	27	24	30	10	2	5
23	28	15	38	15	4	3
24	26	13	55	20	4	5
25	41	32	45	22	2	5
26	40	10	34	19	4	4
27	30	28	38	15	4	4
28	44	32	29	14	2	4
29	29	17	55	12	2	5
30	39	33	35	15	1	4

Retrieval Number: 100.1/ijitee.D107514040325
DOI: 10.35940/ijitee.D1075.14060525
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

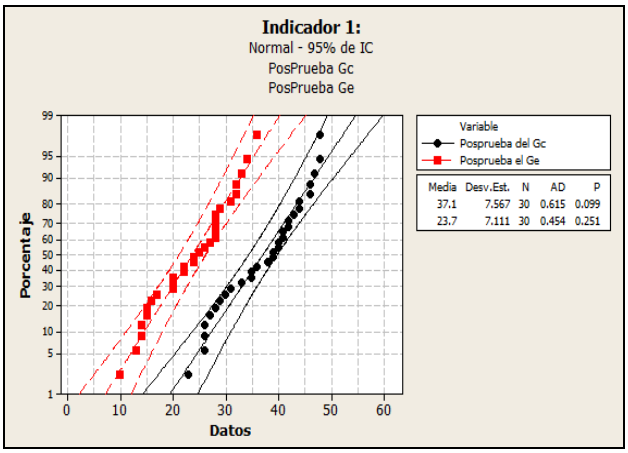
Table- II: Average of the Indicators of the Post-Test Gc and Post-Test Ge

Indicator	Averages	
	Post GC Test	Post Test Ge
I01: Development Time	32.27	23.7
I02: Number of Errors in Tests	26.47	18.9
I05: User Satisfaction	2.8	4.33

A. Normality Tests

The normality test is performed on each indicator to see if the data behaves normally.

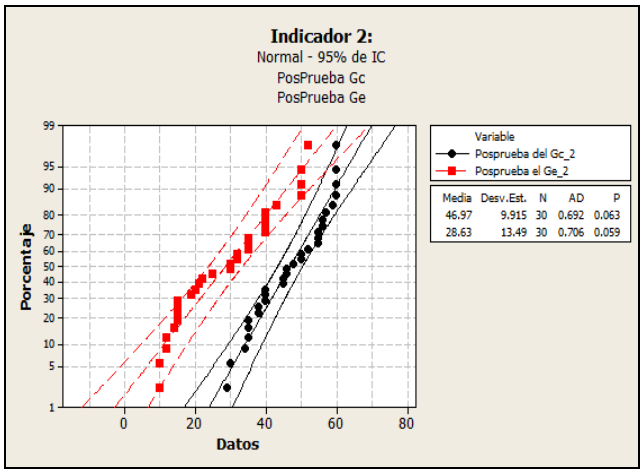
Indicator 1: Average Time in Educational Software Development per hour.



[Fig.2: Normality tests Indicator 1]

Conclusion: Figure 2 displays the Anderson-Darling test results, indicating that the p-values are greater than 0.05 for both PostTest Gc (p = 0.099) and PostTest Ge (p = 0.251). This confirms that they exhibit a normal distribution.

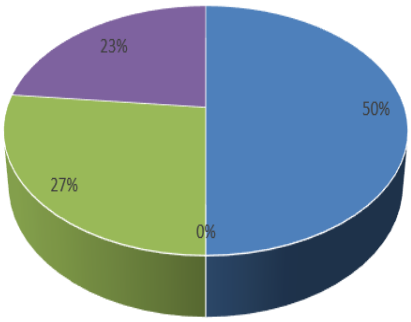
Indicator 2: Number of Errors in Educational Software Testing.



[Fig.3: Normality Tests Indicator 2]

Conclusion: Figure 3 shows that the Anderson-Darling test results indicate that the p-values are greater than 0.05 for both PostTest Gc (p = 0.063) and PostTest Ge (p = 0.059), confirming that they present a normal distribution.

Indicator 3: User Satisfaction with Educational Software.



Conclusion: 50% of the time the Requester Satisfaction was rated as strongly agree by customers, 0% of the time the Requester Satisfaction Level was rated as strongly disagree by customers, 27% of the time the Requester Satisfaction Level was rated as neither agree nor disagree by customers, 23% of the time the Requester Satisfaction Level was rated as somewhat agree by customers.

IV. RESULT AND DISCUSSION

Results In this section, the results obtained within the framework of implementing the agile PIP methodology for developing educational software are interpreted and analysed in relation to the proposed key indicators: reduction in development time, decrease in the number of errors, and increase in user satisfaction. The results are compared with previous studies and the theoretical framework that supports this research.

A. Reducing Time in the Development of Educational Software

One of the main objectives was to verify whether the agile PIP methodology could reduce the development time of educational software. The results show a significant decrease in delivery times, in line with the findings [3] highlighted those agile methodologies, such as Extreme Programming (XP), allow for short iterations and continuous deliveries, speeding up development cycles. Similarly [4], creators of Scrum, underline that agile development delivers products faster thanks to the dynamic prioritization of tasks and flexibility in the face of changing requirements. In the case of the agile PIP methodology, the iterative approach and incremental deliveries allowed for faster times than traditional approaches.

B. Reducing the Number of Errors in Educational Software Testing

The results were consistent with the existing literature, which supports the use of the agile approach in terms of error reduction. According to [5], agile methodologies improve software quality by promoting frequent testing and continuous functionality integration, facilitating early detection of failures. In this research, errors were identified and corrected at early stages, thanks to the implementation of tests in each iteration, aligned with the quality principles proposed by the ISO 29119 standard (International Organisation for Standardisation, 2013). This standard emphasises the importance of continuous testing to ensure software quality. This resulted

Fewer errors were reported in the final phases of development, validating the hypothesis that the PIP methodology increases software stability [6].

C. Increase in User Satisfaction

The active involvement of end users, both teachers and students, during the development process was key to increasing their satisfaction [7]. The agile PIP methodology, by encouraging constant feedback and adaptation of the software to users' changing needs and expectations, significantly improved their perception of the final product. Survey and interview data show a high level of satisfaction, reflecting the importance of involving users in key phases of development. These results are consistent with previous studies, which indicate that user participation in the development process is a crucial factor in increasing software satisfaction and adoption.

V. CONCLUSION

After verifying the hypothesis, it can be concluded that the development times of educational software, software quality and user satisfaction improve significantly with the help of the agile PIP methodology compared to traditional methods, which supports what is stated in the hypothesis of this research; Therefore, the following conclusions can be drawn from the proposed objectives:

The Agile PIP Methodology reduces the time spent developing Educational Software, which results in the ability to manage changes and emerging priorities. According to scale indicators, the application at the Professional School of Systems Engineering of the USMP was handy 80% of the time.

The Agile PIP Methodology reduces the number of errors in Educational Software testing, resulting in greater efficiency and improved project outcomes. According to scale indicators, the application at the Professional School of Systems Engineering at the USMP was handy, with a rating of 80%.

The application of the Agile PIP Methodology increases the satisfaction of end-users who are actively involved in the development process of educational software. According to scale indicators, the application at the Professional School of Systems Engineering of the USMP was handy 80% of the time.

ACKNOWLEDGMENT

César Patricio-Peralta deeply thanks his family for their unconditional support and for being a constant source of inspiration. He also expresses his gratitude to the colleagues and collaborators who, with their valuable guidance, contributed to the development of this work. He also thanks institutions and entities that provided the resources and access to the information necessary to complete this research.

DECLARATION STATEMENT

After aggregating input from all authors, I must verify the accuracy of the following information as the article's author.

- **Conflicts of Interest/Competing Interests:** Based on my understanding, this article does not have any conflicts of interest.
- **Funding Support:** This article has not been funded by any organizations or agencies. This independence ensures that the research is conducted with objectivity and without any external influence.
- **Ethical Approval and Consent to Participate:** The content of this article does not necessitate ethical approval or consent to participate with supporting documentation.
- **Data Access Statement and Material Availability:** The adequate resources of this article are publicly accessible.
- **Author's Contributions:** The authorship of this article is contributed equally to all participating individuals.

REFERENCES

1. Sutherland. Jeff. (2023). SCRUM. https://scrumguides.org/jeff.html
2. Beck, K. (1999). Aceptar el cambio con programación extrema. Computer, 32(10), 70-77. DOI: https://doi.org/10.1109/2.796139
3. Condor S, García C, Dávila A. Adopción de prácticas de pruebas de software de la ISO/IEC 29119-2. Applications in Software Engineering - Proceedings of the 5th International Conference on Software Process Improvement, CIMPS 2016.
   DOI: https://doi.org/10.1109/CIMPS.2016.7802802
4. Muñoz-Arteaga, J., López-Torres, G. C.., & Muñoz-Zavala, Ángel E. (2023). An Agile Learning Methodology to Support Inclusive Education: Una metodología de aprendizaje ágil como apoyo a la educación inclusiva. Edutec, Revista Electrónica De Tecnología Educativa, (86), 116–136.
   DOI: https://doi.org/10.21556/edutec.2023.86.3013.
5. Abrahamsson. (2002) Agile Methodologies Improve Software Quality. DOI: https://doi.org/10.48550/arXiv.1709.08439
6. Rodríguez, A., Gómez, L., y Herrera, Agile methodologies in educational software development: A systematic review. P. 2023.
   DOI: https://doi.org/10.3390/info10100314.
7. Samar A. (2020), Fundamentals of Agile Methodologies. Journal of Software Development.
   DOI: https://doi.org/10.3991/ijim.v14i11.13269.

AUTHOR'S PROFILE

Cesar Patricio-Peralta, Doctor Honoris Causa, PhD in Systems Engineering from the Federico Villarreal National University, Master and Systems Engineer. Professor at the Technological University of Peru and the Continental University. Advisor and Jury of Engineering Theses. Specialisation in Software Development, Big Data, Blockchain, Machine Learning, Business Intelligence, and Databases. International reviewer of scientific articles and author of scientific articles indexed in Scopus.

Jesús Zamora Mondragon, Systems and Computer Engineer, Doctor in Administration, highly qualified professional in ICT, management and administration; accustomed to working as a team with leadership and institutional commitment and faculty advisor at the Universidad César Vallejo.

Jacker Paredes Meneses has been a teacher trainer for English, Spanish, and French since 2000. Mr. Paredes holds a Doctor's Education from the University of Cesar Vallejo. Also, a Master's Degree in Education from the University of Cesar Vallejo. And a Master's Degree in Public

Management from the National University of the Peruvian Amazon and Diplomas. Mr. Paredes works for the National University of the Peruvian Amazon as a teacher in the FACEN Faculty with teenagers, Young adults, and adults. Also works in various prestigious institutions in Peru, including MINEDU and Anglo. His special interests include research, teaching oral communication skills, incorporating TikTok and media into ELT classes, and online professional development for international teachers.

Jimmy Max Ramirez Villacorta, Graduate and Holder of a degree from the National University of the Peruvian Amazon, from the Faculty of Systems Engineering and Informatics. Master of the Postgraduate School of Cesar Vallejo University in Educational Administration, a doctoral candidate in Systems Engineering at the Federico Villarreal National University, and a researcher for RENACYT.

Carlos Alberto Garcia Cortegano, holds a Doctorate in Systems Engineering and a Master's degree in Administration, with a distinguished academic career at the National University of the Peruvian Amazon (UNAP), where he serves as a Principal Professor and was Dean of the Faculty of Systems Engineering and Informatics (2016-2019). As a RENACYT researcher, he has conducted studies in artificial intelligence, cybersecurity, and mobile applications, with notable work in disease detection using neural networks, the revitalisation of native languages, and real-time surveillance systems. His contributions have been instrumental in the technological development of the Amazon region, and his scientific publications can be found in the CTI Vitae of CONCYTEC.

Adan Eugenio Contreras Angeles, Doctor in Systems Engineering and Master in Public Management, professor at the Technological University of Peru.