

Herramienta jMDA-Módulo CIM-PIM 3.0

JMDA tool- CIM-PIM module version 3.0

Rosendo Moreno Rodríguez ^{1*} <https://orcid.org/0000-0002-2918-0207>

Suleiny Rodríguez Aguila ² <https://orcid.org/0009-0001-5188-9749>

¹ Universidad Central “Marta Abreu” de las Villas, Cuba

² Oficina Territorial de Normalización de Villa Clara, Cuba.

* Autor para la correspondencia. (rosendojmr@gmail.com)

RESUMEN

El presente trabajo se corresponde a una nueva versión del módulo CIM-PIM de la Herramienta CASE jMDA que permita construir a través de una secuencia de preguntas y respuestas, la base inicial del Modelo Independiente de la Computación (CIM) y transformarlo de manera automática a los Diagramas de Casos de Uso, Clases y Actividades a nivel del Modelo Independiente de la Plataforma (PIM); todo ello a través de una interfaz intuitiva y amigable donde no sea necesario que el usuario tenga conocimientos básicos de “Ingeniería de Software”, pero en donde se destaca como novedad el uso de herramientas de inteligencia artificial, en este caso ChatGPT.

La interfaz de la versión anterior (2.0), tenía una lista de preguntas con posibles respuestas, pero fijas correspondientes a la experiencia del autor en el desarrollo de Sistemas de Información. En esta versión se propicia que el cliente describa en lenguaje natural la idea que tiene sobre el software que desea, y esa información es analizada por ChatGPT de OpenAI, para determinar los elementos necesarios (funcionalidades, entidades y sus características, pasos de secuencias de acciones, etc.) para los diagramas UML detallados antes, los cuales son interpretados posteriormente con PlantUML para crear dichos diagramas en ambiente PIM.

Palabras clave: Ingeniería del Software; Arquitectura Dirigida por Modelos; Módulo CIM-PIM; ChatGPT.

ABSTRACT

The present work corresponds to a new version of the CIM-PIM module of the CASE jMDA Tool that allows building, through a sequence of questions and answers, the initial basis of the Computing Independent Model (CIM) and automatically transforming it into Diagrams of Use Cases, Classes and Activities at the Platform Independent Model (PIM) level; all of this through an intuitive and friendly interface where it is not necessary for the user to have basic knowledge of “Software Engineering”, but where the use of artificial intelligence tools, in this case ChatGPT, stands out as a novelty.

The interface of the previous version (2.0) had a list of questions whit possible but fixed answers corresponding to the author’s experience in the development of Information Systems. This version encourages the client to describe in natural language the idea they have about the software they want, and this information is analyzed by OpenAi’s ChatGPT, to determine the necessary elements (functionalities, entities and their characteristics, action sequence steps, etc.) for the UML diagrams detailed above, which are subsequently interpreted whit PlanUML to create said diagrams in PIM environment.

Keywords: Software Engineering; Model Driven Architecture; CIM-PIM Module; ChatGPT.

Recibido: 09/06/2024

Aceptado: 01/10/2024

Introducción

Arquitectura Dirigida por Modelos y Lenguaje Unificado de Modelado

Los sistemas de software raramente son estáticos y necesitan iterar varias veces en busca de errores o para satisfacer nuevos requerimientos. Un enfoque popular es el uso del modelado desde etapas tempranas de análisis de requisitos (Uzun and Tekisnerdohgan, 2018). La obtención de diagramas de alto nivel a partir del análisis de requisitos basado en historias de usuario, no solo ayuda a la productividad en el desarrollo del software sino también en sus pruebas, definidas como una secuencia de acciones que ayuda a verificar si el software bajo prueba cumple con los requerimientos (Alégroth, E. et al, 2022 y Garousi V. et al, 2021).

El proceso central de MDA es la transformación de modelos que parten del espacio del problema (CIM) hasta modelos específicos de la plataforma (PSM), pasando por modelos que describen una solución independientemente de la computación (PIM).

En MDA, el modelo de objetos del dominio se llama PIM (*Platform Independent Model* o modelo independiente de la plataforma). El PIM está constituido por un conjunto de elementos cuyo diseño debe hacerse de forma independiente a cualquier lenguaje de programación o tecnología (Korchi and Khachouch, 2024 y (Debrauwer and Van der Heyde, 2016).

- ✓ CIM. Representa los modelos independientes de la computación (*Computation Independent Model*) que caracterizan el dominio del problema. Este tipo de modelos surge ante todo en procesos de modelado de negocio e idealmente se conciben antes del levantamiento de requisitos para una aplicación particular. (Osis J., Asnina, E. 2015 y Osis J., y otros. 2007).
- ✓ PIM. Representa los modelos que describen una solución de software que no contiene detalles de la plataforma concreta en que la solución va a ser implementada, de ahí su nombre de modelos independientes de la plataforma (*Platform-Independent Models*). Estos modelos surgen como resultado del análisis y diseño (Liu, Huang, Liu, 2024).

Como todo proceso la arquitectura MDA tiene ventajas y desventajas, pero sin duda el hecho de estar centrada en el negocio, soportar un proceso de desarrollo incremental (sea complejo como RUP, o ágil), el hecho de facilitar los cambios, incluso hacerlos reversibles de manera automática al modelo previo, la posibilidad de portabilidad y sobre todo la productividad, hacen muy atractiva esta forma de trabajo.

Los procesos y roles esenciales de MDA se pueden observar en la siguiente figura, donde se enmarca en rojo el área de esta investigación.

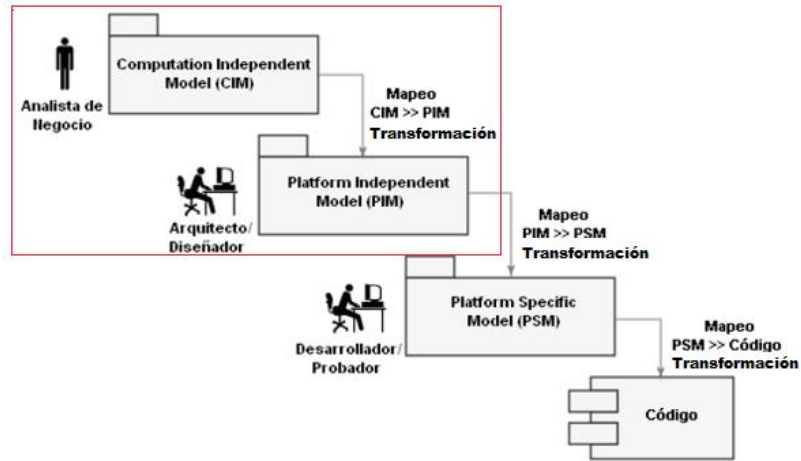


Fig. 1 - Procesos y roles en MDA.

OMG ha definido un conjunto de estándares para el desarrollo de modelos a través de MDA de los cuales se destacan: UML (*Unified Model Language*), MOF (*Meta Object Facility*), QVT (*Query View Transformation*), OCL (*Object Constraint Language*) y XMI (*XML Metadata Interchange*) (Gaitán Peña, 2017).

El Lenguaje Unificado de Modelado (“Unified Modeling Language” o UML, por sus siglas en inglés), constituye el lenguaje de modelado para desarrollar sistemas de software más usado actualmente; está respaldado por el OMG. Es un lenguaje gráfico para el desarrollo de sistemas que permite modelar la arquitectura, los objetos, las interacciones entre objetos, datos y aspectos del ciclo de vida de una aplicación, así como otros aspectos más relacionados con el diseño de componentes incluyendo su construcción y despliegue. En la práctica, los modelos UML fueron usados primeramente para el análisis y la documentación, sin tener en cuenta una variedad de herramientas que soportan ejecutables de UML (Ciccozzi, F., Malavolta, I., Selic, B. 2019).

De los 13 diagramas establecidos hasta la UML 2.5, en la investigación se ha determinado que para el análisis de requisitos conjuntamente con el cliente en el modelo CIM es básico lograr al menos una

transformación a PIM con Diagramas de Casos de Uso (como modelo de negocio), de Clases (como modelo de negocio) y de Actividades (Melouk, M., y otros. 2020).

Inteligencia Artificial y ChatGPT

Los avances de unas tecnologías y la obsolescencia de otras marchan a una velocidad inimaginable, especialmente en este siglo XXI. La inteligencia artificial (IA) se ha desarrollado vertiginosamente e impacta la vida de la sociedad. Los expertos en IA creen que esta tecnología seguirá avanzando y transformando la sociedad en los próximos años (Angelov, PP., y otros. 2021). Si bien es difícil predecir con certeza cuándo se alcanzará un hito importante en la IA, algunos sugieren que el año 2023 podría ser significativo debido a los avances en áreas como el aprendizaje automático, la robótica y la computación cuántica. Sin embargo, es importante recordar que la IA es un campo en constante evolución y siempre habrá nuevos desafíos y oportunidades para explorar, como su uso en la Ingeniería del Software (Shehab, M., y otros. 2020 y Sofian, H., y otros. 2022). Algunas definiciones se resumen en el siguiente gráfico:



Fig. 2 - Definiciones de varios autores sobre la IA.

Una de las expresiones de la IA más recientes es el ChatGPT que debe su nombre a las siglas en inglés de *Generative Pre-trained Transformer* o Transformador Preentrenado Generativo. ChatGPT es un modelo de lenguaje generativo desarrollado por OpenAI que utiliza el aprendizaje profundo para generar respuestas

coherentes y naturales en conversaciones con humanos. Esto ha permitido la creación de asistentes virtuales más avanzados y personalizados, así como la automatización de servicios de atención al cliente y soporte técnico.

El ChatGPT funciona utilizando un modelo de lenguaje generativo basado en técnicas de aprendizaje profundo. Este modelo es entrenado con grandes cantidades de datos de conversaciones humanas, lo que le permite aprender patrones y estructuras del lenguaje natural (Cuartiellas y otros. 2023). Cuando un usuario interactúa con el Chat GPT, el modelo utiliza la información proporcionada por el mismo para generar una respuesta coherente y natural. El modelo también es capaz de aprender continuamente a medida que interactúa con los usuarios, mejorando su capacidad de respuesta y personalización. Puede ser entrenado en diferentes idiomas y contextos, lo que lo hace adaptable a diferentes necesidades y situaciones. Además, puede ser personalizado para adaptarse a las necesidades específicas de cada empresa o usuario, mejorando su experiencia y la eficacia del servicio. La aplicación facilita la inclusión de personas con limitaciones físicas al acceso de la información, (Diego Olite, y otros. 2023). Tanto en ChatGPT como en el teclado predictivo del móvil se utilizan algoritmos para tratar predecir texto (Sánchez-Vera, 2023).

Algunas características de ChatGPT son: Generación de respuestas naturales, Capacidad de aprendizaje, Adaptabilidad, Automatización de servicios, y Personalización. También tiene efectos en el análisis lingüístico (Rafique, H., y otros. 2024).

ChatGPT ha evolucionado de manera vertiginosa. Su funcionamiento se estructura en la integración de capas y parámetros. Se nutre del procesamiento de millones de informaciones existentes en internet y que van desde las científicas y fidedignas hasta otras que aparecen de modo no verificable en las redes, como algunos contenidos de Wikipedia (Diego Olite, y otros. 2023).

Métodos o Metodología Computacional

El presente trabajo es una investigación aplicada. Se usaron métodos y técnicas de la Ingeniería del Software como el modelado UML y la programación en Java, para el desarrollo de la herramienta. Se consideraron experiencias prácticas en el desarrollo de Sistemas de Información, para establecer un orden lógico de preguntas-respuestas, así como las ayudas o propuestas que brinda el software a partir de su interacción con ChatGPT.

Resultados y discusión

Análisis y diseño de la interfaz gráfica de usuario

La anterior versión del módulo CIM-PIM (Moreno-Rodríguez, R. 2020) presenta una interfaz muy rígida de preguntas y respuestas, aunque su énfasis estuvo en la generación automática de los diagramas UML necesarios. Al igual que en esa ocasión, ahora se desarrolló el modelado, usando específicamente los diagramas de Casos de Uso, Clases y Artefactos.

El Diagrama de Casos de Uso incluye 3 actores, con varios casos de uso específicos, aunque algunos son *extend* o *include* de otros principales, como se puede apreciar en la siguiente figura:

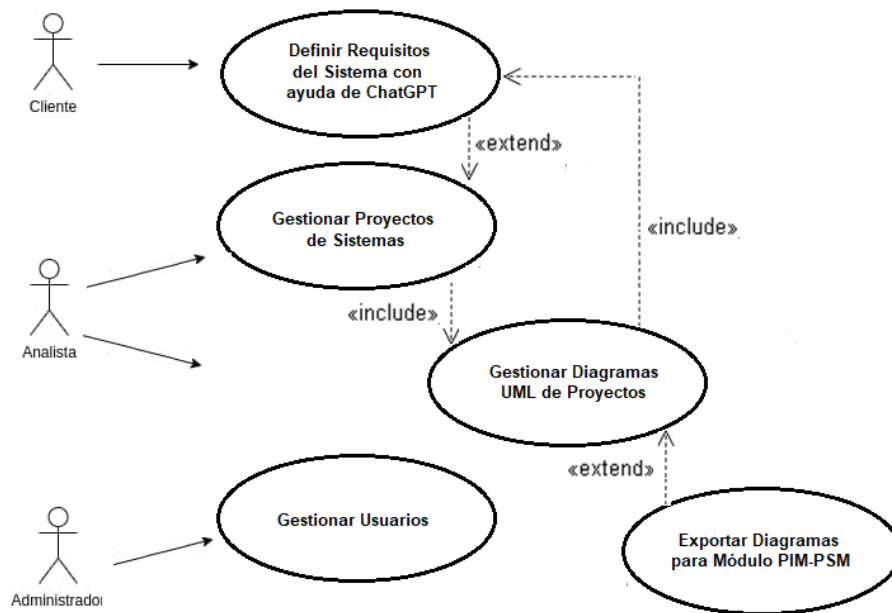


Fig. 3 - Casos de Uso del Módulo CIM-PIM.

Tabla 1 - Descripción del caso de uso Definir Requisitos del Sistema con ayuda del ChatGPT.

Caso de uso del sistema		Definir Requisitos del Sistema	
Actor	Cliente		
Propósito	Responder secuencia de preguntas que permitirán ir incorporando los requisitos que debe cumplir el nuevo sistema.		
Resumen	Inicia cuando el cliente entra al proyecto que le fue asignado y comienza a introducir requisitos del sistema apoyándose en las sugerencias que le brinda el asistente a partir del ChatGPT para llegar a generar los diagramas deseados.		
Responsabilidades	Introducir los requisitos del sistema a través de una interfaz intuitiva y amigable a partir de la secuencia de pasos y empleando las sugerencias de preguntas y respuestas que los son proporcionadas con ayuda de ChatGPT.		
Casos de uso asociados	Gestionar Proyectos, Gestionar Diagramas, Exportar Diagramas.		
Flujo normal de los eventos			
Acción del actor		Respuesta del sistema	
1- Entrar al proyecto asignado y accionar el botón cargar sugerencias (Modelo de Requisitos)		2- Mostrar sugerencias de preguntas y respuestas que puede seleccionar el cliente para introducir como requisitos en el editor de texto.	
3- Pasar a la opción del Modelo de casos de uso y Accionar el botón <i>cargar sugerencias</i> (del Modelo de casos de uso) y así sucesivamente con los modelos de Clases y de Actividades.		4- Mostrar sugerencias de preguntas y respuestas que puede seleccionar el cliente para introducir en el editor de texto y finalmente mostrar el diagrama de Casos de Uso, Clases y Actividades en formato PNG y en su código con PlantUML.	

Breve descripción de la herramienta

Al iniciar la herramienta el analista debe crear un proyecto dentro del cual se establecerán requisitos acordes al intercambio de preguntas y respuestas que puede posteriormente dar lugar a los modelos de análisis del sistema requeridos. Para esto se presenta la siguiente pantalla:

Crear proyecto

Nombre

tienda de ventas de electrónicos

vender productos de electrónica como pjs, celulares etc

CERRAR GUARDAR

Fig. 4 - Ventana para crear un proyecto.

Durante la secuencia de pasos a seguir para responder las preguntas que permiten obtener los requisitos del sistema deseado, la herramienta proporciona sugerencias a partir de ChatGPT que pueden ser tomadas en cuenta o no, y de serlo basta con marcar y copiar y después pegar en la zona de descripción del problema como en un editor de texto.



Fig. 5 - Sugerencias de ChatGPT para los requisitos.

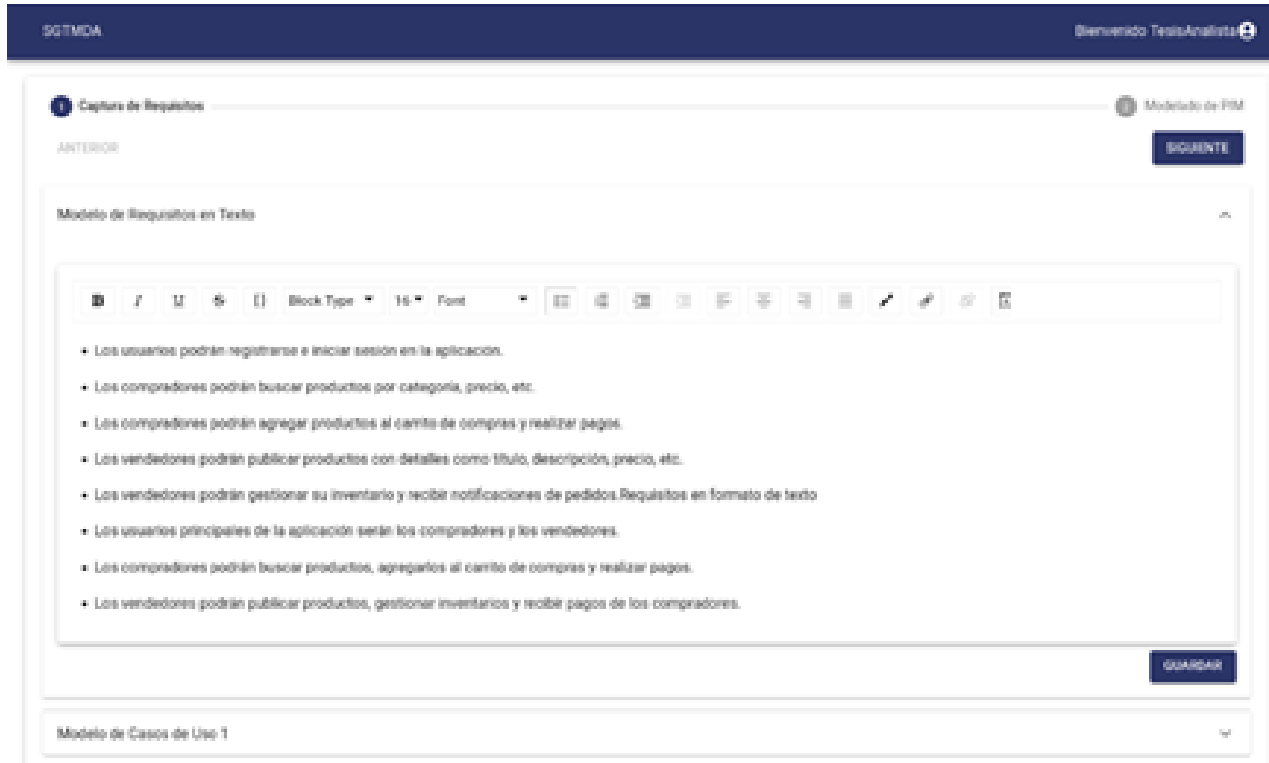


Fig. 6 - Requisitos introducidos en el editor de texto.



Fig. 7 - Pseudocódigo PlantUML y diagrama de casos de uso en formato PNG obtenidos a partir de requisitos.

De la misma manera otra serie de ventanas se presentan durante el proceso de uso del módulo CIM-PIM para crear los diagramas de clases y de actividades propuestos para el sistema de información solicitado a partir de requisitos determinados con ChatGPT.



Fig. 8 - Fragmentos de las ventanas de Requisitos para Clases, pseudocódigo PlantUML y Diagrama de Clases parcial.

Conclusiones

1. Se desarrolló una nueva versión del módulo CIM-PIM de la Herramienta jMDA que la UCLV está proponiendo como un CASE viable y soberano con la idea de aumentar la productividad del trabajo de desarrollo de Sistemas de Información de las diferentes entidades que se dedican a esta tarea en el país.
2. La versión 3.0 del módulo CIM-PIM de la herramienta jMDA tiene como novedad el uso de una herramienta libre (ChatGPT) lo que implica el uso de conceptos de la inteligencia artificial, y como otro elemento fundamental, el usuario no necesita conocimientos básicos de “Ingeniería de Software” para responder la secuencia de preguntas que dan lugar a los requisitos del sistema.

3. Con ayuda de otro software libre, PlantUML, los elementos se organizan y se logra el diagramado UML propuesto: Diagramas de Casos de Uso, Clases y Actividades.

Referencias

- Usun, B.; Tekinerdogan, B. 2018. Architecture Conformance Analysis Using Model-Based Testing: A Case Study Approach. *Softw: Pract Exper.* 2018;1–26. <https://doi.org/10.1002/Spe.2667>. [Pdf]
- Alégroth, E., Karl, K., Rosshagen, H., Helmfridsson, T., Olsson, N. 2022. Practitioners' Best Practices To Adopt, Use Or Abandon Model-Based Testing With Graphical Models For Software-Intensive Systems. *Empirical Software Engineering* 27(5), 103, 2022. [Ttps://doi.org/10.1007/S10664-002-10145-2](https://doi.org/10.1007/S10664-002-10145-2). [Pdf]
- Garousi V, Keles, Ab, Balaman Y, G'Uler Zo, Arcuri A. 2021. Model-Based Testing In Practice: An Experience Report From The Web Applications Domain. Arxiv:2104.02152.
- Korchi, A., Khachouch, Mk., Lakhrissi, Y. 2024. A Model-Driven Architecture Solution For Multi-Platform Mobile App Development. 2024. *Journal Of System And Management Sciences.* Vol. 14 (2024) No. 5, Pp 1-13. Doi: 10.33168/Jsms.2024.0501. Issn 1816-6075 (Print), 1818-0523 (Online). [Pdf]
- Debrauwer, L. & Van Der Heyde, F. 2016. Uml 2.5: Iniciación, Ejemplos Y Ejercicios Corregidos. 5ta. Edición. Ediciones Eni. Octubre De 2021. Isbn: 978-2-409-00372-1.
- Liu, Z., Huang, Y., Liu, L. 2024. Shaderperformer: Platform-Independent Context-Aware Shader Performance Predictor. *Proceedings Of The Acm On Computer Graphics And Interactive Techniques.* Vol 7, No. 1, Pp 1-7. Mayo-2024. <https://doi.org/10.1145/3651295>.
- Osis J., Asnina, E. 2015. Is Modeling A Treatment For The Weakness Of Software Engineering? *Handbook Of Research On Innovations In Systems And Software Engineering.* Cap. 17. Doi: 10.4018/978-1-4666-6359-6.Ch017. [Pdf]

- Melouk, M., Rhazali, Y., Youssef, H. 2020. An Approach For Transforming Cim To Pim Up To Psm In Mda. *Procedia Computer Science*. Vol. 170, Pp 869-874. Elsevier. [Www.Sciencedirect.Com](http://www.sciencedirect.com). [Https://Doi.Org/10.1016/J.Procs.2020.03.122](https://doi.org/10.1016/j.procs.2020.03.122). [Pdf]
- Osis J., Nazaruka, E., Grave, A. 2007 Computation Independent Modeling Within The Mda. *Ieee Xplore*. [Doi:10.1109/Swste.2007.20](https://doi.org/10.1109/Swste.2007.20). [Www.Researchgate.Net](http://www.researchgate.net). [Pdf]
- Ciccozzi, F., Malavolta, I., Selic, B. 2019. Execution Of Uml Models: A Systematic Review Of Research And Practice. *Software & Systems Modeling* (2019) 18:2313–2360. [Https://Doi.Org/10.1007/S10270-018-0675-4](https://doi.org/10.1007/S10270-018-0675-4). [Pdf]
- Gaitán Peña, C. A. 2017. Líneas De Productos Software: Generando Código A Partir De Modelos Y Patrones. [Pdf] *Revista Scientia Et Technica*. Año Xxii, Vol. 22, No. 2, Junio De 2017. Universidad Tecnológica De Pereira. Colombia. Issn 0122-1701 Disponible En [Http://Www.Redalyc.Org/Artículo.Oa?Id=84953103009](http://www.redalyc.org/articulo.oa?id=84953103009).
- Uml-Omg 2012. Omg Unified Modeling Language. [En Línea]. Digitum: Repositorio Institucional De La Universidad De Murcia. Junio 2023. [Consultado El: 12 De Octubre De 2023] 664-678 P. Disponible En: [Http://Www.Omg.Org/Spec/Uml/2.5](http://www.omg.org/spec/UML/2.5)
- Moreno-Rodríguez, R. 2020. Case Jmda De Arquitectura Dirigida Por Modelos Para Sistemas De Información. *Rcci: Revista Cubana De Las Ciencias Informáticas. Uci*. Vol:14. No. 4 (2020). Issn-E: 2227-1899.
- Angelov, Pp., Soares, Ea., Jian, R., Arnold, Ni., Atkinson, Pm. 2021. Explainable Artificial Intelligence: An Analytical Review. *Wiley Interdisciplinary Reviews: Dta Mining And Knowledge Discovery* 11 (5), E1424, 2021. [Doi: 10.1002/Widm.1424](https://doi.org/10.1002/widm.1424). [Pdf]
- Shehab, M., Abualigah, L., Jarrah, Mi., Alomari, Oa., Daoud, Ms. 2020. (Aiam2019) Artificial Intelligence In Software Engineering And Inverse: Review, *International Journal Of Computer Integrated Manufacturing*, 33:10-11, 1129-1144, [Doi: 10.1080/0951192x.2020.1780320](https://doi.org/10.1080/0951192x.2020.1780320). [Https://Doi.Org/10.1080/0951192x.2020.1780320](https://doi.org/10.1080/0951192x.2020.1780320). [Pdf]

- Sofian, H., Arzilawati, N., Ahmad, R. 2022. Systematic Mapping: Artificial Intelligence Techniques In Software Engineering. Ieee Access. 2022. Digital Object Identifier 10.1109/Access.2022.3174115. [Pdf]
- Cuartielles, R.; Ramon-Vegas, X.; Pont-Sorribes, C. 2023. “Retraining Fact-Checkers: The Emergence Of Chatgpt In Information Verification”. Profesional De La Información, Vol. 32, No. 5, E320515. <https://doi.org/10.3145/Epi.2023.Sep.15>. <https://revista.profesionaldelainformacion.com/index.php/Epi/article/view/87368>. [Pdf]
- Diego Olite, F. M., Morales Suárez, I. R. & Vidal Ledo, M. J. 2023. Chat Gpt: Origen, Evolución, Retos E Impactos En La Educación. Revista Educación Médica Superior, Vol. 37, No. 2. Habana Abr-Jun 2023. Issn 0864-2141. Disponible En: <http://scielo.sld.cu/scielo.php?script>
- Rafique, H., Nazeer, I., Rehman, J.. 2024. The Impact Of Chatgpt On Language Evolution: A Linguistic Analysis. Journal Of Education And Social Studies, 5(1) 2024. Pp 56-68. Issn: 2789-8075. <http://www.scienceimpactpub.com/jess>. [Pdf]
- Sánchez-Vera, M. 2023. Hasta Chat Gpt Y Más Allá: Una Breve Guía Reflexiva Sobre El Impacto De La Inteligencia Artificial En La Educación. [Pdf]. [Consultado El: 23 De Octubre De 2023] <http://hdl.handle.net/10201/131973>.

Conflicto de interés

Los autores autorizan la distribución y uso de s artículo.

Contribuciones de los autores

Conceptualización: Dr. Rosendo Moreno Rodríguez.

Investigación: Dr. Rosendo Moreno Rodríguez y M.Sc. Suleiny Rodríguez Aguila

Metodología: Dr. Rosendo Moreno Rodríguez y M.Sc. Suleiny Rodríguez Aguila

Administración del proyecto: Dr. Rosendo Moreno Rodríguez

Recursos: M.Sc. Suleiny Rodríguez Aguila

Software: M.Sc. Suleiny Rodríguez Aguila

Supervisión: Dr. Rosendo Moreno Rodríguez

Validación: M.Sc. Suleiny Rodríguez Aguila

Visualización: Dr. Rosendo Moreno Rodríguez y M.Sc. Suleiny Rodríguez Aguila

Redacción – borrador original: Dr. Rosendo Moreno Rodríguez

Redacción – revisión y edición: Dr. Rosendo Moreno Rodríguez