

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



**Departamento:**

Departamento de ingeniería e investigaciones tecnológicas  
(DIIT)

**Programa de acreditación:**

PROINCE

**Programa de Investigación<sup>1</sup>:**

**Código del Proyecto:**

C-256

**Título del proyecto**

Framework para explotación de modelos de IA generativos multimodales

**PIDC:**

Elija un elemento.

**PII:**

Elija un elemento.

**Informe Final**

**Director:**

Ierache, Jorge Salvador

**Director externo:**

**Codirector:-**

---

<sup>1</sup> Los Programas de Investigación de la UNLaM están acreditados con resolución rectoral, según lo indica la Resolución HCS N° 014/15 sobre **Lineamientos generales para el establecimiento, desarrollo y gestión de Programas de Investigación a desarrollarse en la Universidad Nacional de La Matanza**. Consultar en el departamento académico correspondiente la inscripción del proyecto en un Programa acreditado.

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



## A. Resumen del proyecto<sup>2</sup> (Desarrolle en no más de dos páginas.)

### Problema de investigación

Bajo la creciente tendencia de creación de nuevas técnicas para generar prompts e integrar hechos fácticos para extender el conocimiento de un LLM es cada vez más complejo integrar estos enfoques en aplicaciones para ingenieros sin conocimientos técnicos de IA. Por lo tanto, se requiere herramientas que asistan a la creación de prompts para mejorar la integración y efectividad de los LLMs en sistemas informáticos.

### Grado de ejecución de los objetivos planteados

En el primer año del proyecto, debido al crecimiento exponencial de desarrollos de las tecnologías alrededor del desarrollo de aplicaciones con LLM se planteó profundizar en la búsqueda del estado del arte de las técnicas y desarrollos de la disciplina de prompt engineering. El estado del arte reflejaba principalmente el uso de modelos GPT 3 y 4 [1], ChatGPT [2], Llama 2 (LLM Meta AI) [3] y el modelo de inteligencia artificial creado por Google que denominó "Gemini IA" [4]; modelos aplicados a tareas de propósito general multimodales (Ejemplo: texto e imágenes). Actualmente, se entrenaron hasta el momento diversos modelos con diferentes niveles de inteligencia, por ejemplo, la familia de Llama 3/4[5] que mejoran las capacidades de texto, sumando capacidades multimodales. En el transcurso del último trimestre de 2024 y principios de 2025 se entrenaron modelos GPT o1, o3[6], Deepseek R1[7] que ofrecen una simulación en razonamiento de manera estadística. Por otro lado Nvidia[8] y Anthropic[9] impulsaron la introducción de agentes bajo un nuevo paradigma de aprendizaje automático llamado Inteligencia Artificial Agéntica (Agentic IA) que se diferencia de las bases del paradigma de agentes simbólico impulsado por Michael Wooldridge[10], Stuart J. Russell and Peter Norvig[11].

Para que un LLM resuelva problemas de un sistema efectivamente se necesita como input un Prompt, representan un comienzo de texto que debe continuar a modo de órdenes. En el estado del arte se encuentra la disciplina llamada Ingeniería de Prompt (En inglés Prompt engineering) que define un vocabulario, conjunto de técnicas para escribir prompts y mecanismos para ampliar el conocimiento paramétrico de un modelo (Conocimiento adquirido en el preentrenamiento) con conocimiento externo fáctico como contexto en el Prompt ingresado. A continuación, se define el siguiente vocabulario a tener en cuenta:

- **Prompt ingenuo (Naive prompt):** Es un prompt corto y rápido para obtener información de manera superficial. Suele usarse en asistentes para poder consultar información rápidamente.
- **Consulta de usuario (User Query):** Es el pedido de un usuario a un sistema potenciado por un LLM. Esto depende del dominio e intención de lo que desea hacer un usuario.
- **Prompt de sistema, (System Prompt):** Es una instrucción que guía la consulta del usuario hacia el resultado deseado. Suele agregar más detalles, aplicar restricciones del dominio de la aplicación, restricciones de seguridad para evitar inyecciones de prompt no deseadas

<sup>2</sup> Actualizar todos los ítems en el *Banco de datos de actividades de CyT del SIGEVA UNLAM* (del Director y de los integrantes del Proyecto), en especial "Antecedentes y Producciones y Servicios". Ver: [www.youtube.com/@cvtunlam](http://www.youtube.com/@cvtunlam)

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



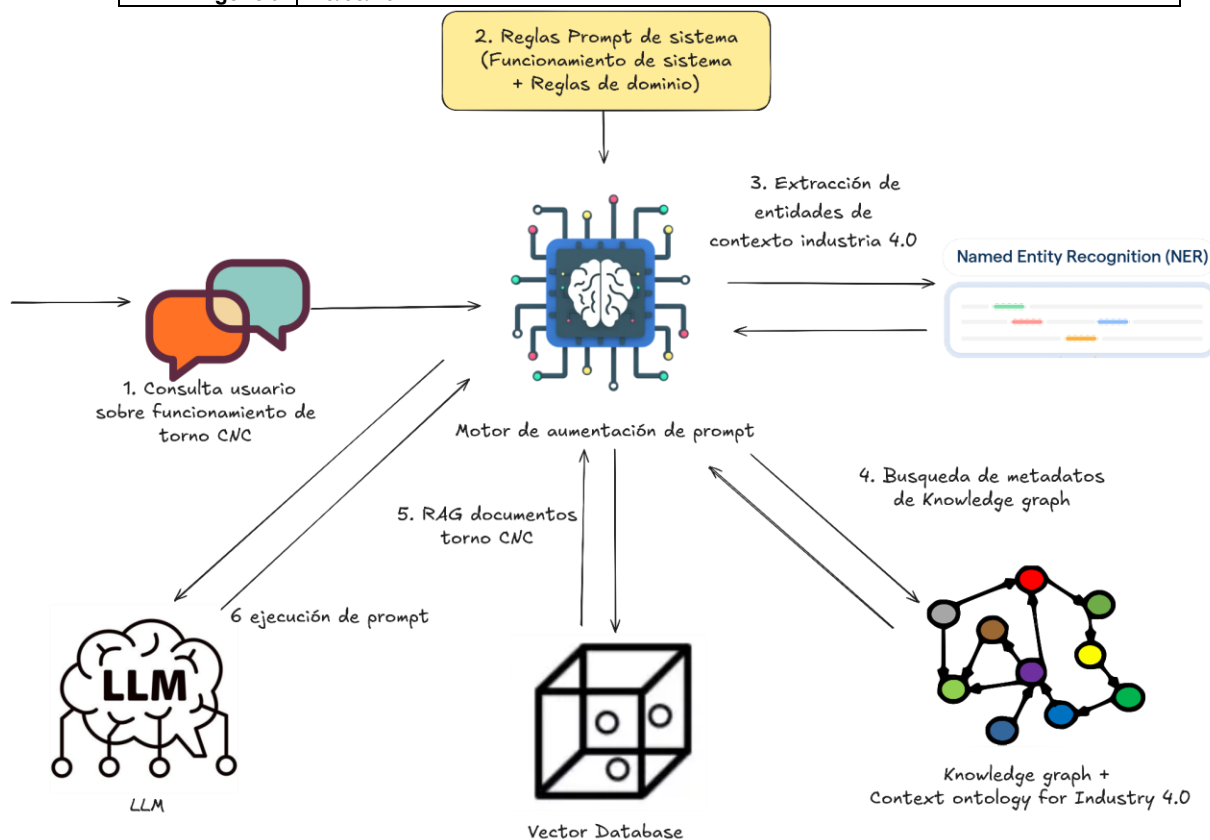
(Prompt injections) para realizar tareas maliciosas o vulnerar la integridad del modelo mediante técnicas jailbreaks prompts.

- **RAG (Retrieve Augmented Generation):** Es un mecanismo para incorporar conocimiento externo de documentos y bases de datos estructuradas para ampliar el conocimiento paramétrico del modelo obtenido en la fase de preentrenamiento. Ejemplos que se pueden citar RAG [12], Easy RAG[13], Auto RAG[14], GraphRAG[15].
- **Técnicas de prompt, (Prompt technique):** Son técnicas que buscan maximizar la calidad de los resultados, abstrayendo el armado complejo de prompts a partir de una consulta de usuario simple. Busca simplificar el proceso de escritura de query user prompts. Suelen trabajar a nivel de system prompt. Ejemplos de técnicas complejas son chain of thoughts[16], automatic chain of thoughts[17], tree of thoughts[18], graph of thoughts[19], skeleton of thoughts[20], thread of thoughts[21] y Multimodal chain-of-thought[22] entre otras.

En la segunda parte del proyecto (2025) se decidió reemplazar las tareas de pruebas de concepto multiagentes debido a costos operativos y complejidad alta de interacciones, razón por la cual se enfoca el proyecto en su objetivo primario en el contexto de LLM individual. Por lo tanto, se realizó una prueba de concepto de un prototipo de Prompt augmentation Engine que facilitara el manejo del prompt por parte de los desarrolladores de aplicaciones potenciadas por LLMs. En esta prueba un usuario realiza consultas sobre el funcionamiento de un torno CNC a un LLM integrado en el prototipo de engine propuesto. Este permite incorporar reglas de funcionamiento de sistema que permite especificar cómo se comportará el sistema a nivel general (estilo de respuestas, identificar el nivel de expertise del usuario entre otras) y reglas de dominio que son proporcionadas por expertos de dominio de aplicación de la industria 4.0.

Para las pruebas realizadas se desacoplan las extracciones de entidades, la búsqueda de metadatos en Knowledge graphs y la técnica RAG debido a la complejidad e imposibilidad de lograr una solución general que funcione para diferentes configuraciones de sistemas. De esta manera, cada implementador del prototipo de engine aporta estos pasos para que se ajusten a sus casos de uso en particular. Adicionalmente al planteo original se incorporaron el uso de Guardrails desacoplados del engine para entradas y salidas del sistema para que apliquen controles de seguridad a las entradas y salidas del modelo de lenguaje (LLM). En la figura 1 se puede observar el funcionamiento del engine, En el paso 1 el usuario realiza una pregunta del funcionamiento del torno CNC en la que se le aplica el guardrail de input para aplicar controles de seguridad. En caso que falle algún guardrail, se detiene la ejecución mediante una excepción y en caso contrario se avanza con la ejecución. En el paso 2 se arma el prompt de sistema a partir de la consulta del usuario, reglas de prompt de funcionamiento del sistema + reglas de dominio. Sumadas a las entidades encontradas (Paso 3) y sus metadatos encontrados (Paso 4). En el siguiente paso 5 se incorpora contexto de documentos mediante la técnica RAG utilizada por el implementador del engine y los metadatos que enriquecen la búsqueda de chunks de documentos relevantes. Por último se ejecuta el prompt en el LLM y en la salida es evaluada por el guardrail de salida para verificar que sea correcta antes de devolver el resultado al usuario. En el artículo [23] se puede observar en detalle el funcionamiento de la PoC y resultados obtenidos en el anexo presentado.

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



**Fig 1.** Diagrama conceptual del prototipo de prompt augmentation engine propuesto

### Resultado obtenido: Template para motor de aumentación de prompts para large language models

El resultado final del proyecto es un template de pasos a realizar por un motor de aumentación de prompts para aplicar guardrails que controlen los inputs de usuario y outputs de LLMs, integrar conocimiento de expertos de dominio para resolver un problema mediante el uso de reglas en lenguaje natural y aplicar metadatos extraídos de un knowledge graphs[24] para enriquecer el prompt de usuario antes de la aplicación de técnicas RAG. La utilidad del template es informar pasos necesarios a considerar para la tarea de aumentación de prompts independientemente de la arquitectura y/o framework que se utilicen en la aplicación potenciada por LLMs dejando a libre elección los elementos dependientes de dominio (Guardrails, reglas de negocio, Knowledge graphs), modelos LLM, técnicas de prompt engineering y técnicas RAG para la incorporación de contextos. En el caso de aplicación propuesto, se aumentaron los prompts de un usuario de un sistema de preguntas y respuestas sobre el funcionamiento de un torno CNC bajo el contexto de la industria 4.0. En la próxima sección se explicará el estado del arte investigado para destacar las vacancias encontradas en el estado de arte a cubrir por el trabajo propuesto.

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



## Estado del arte

Debido al gran crecimiento y complejidad de la disciplina de ingeniería de prompts surgieron frameworks orquestadores como Langchain[25] para facilitar y organizar transformaciones, inserciones de datos en forma de cadenas que son ejecutadas en LLMs. Luego, surgieron frameworks modulares como DSPy[26] que permiten plantear tareas estructuradas y específicas compiladas como código para los LLM y que ofrecen un valioso módulo de evaluación de resultados que destaca sobre los demás. Bajo esta línea Google lanzó una biblioteca orientada a investigadores llamada OneTwo[27] que permite ejecutar experimentos para poder obtener diferentes métricas. Por último, surgieron framework como AdalFlow[28] que optimizan la ejecución de prompts a bajo nivel optimizando pesos del modelo LLM mediante finetuning.

En el estado del arte las bibliotecas indicadas anteriormente DSPy, OneTwo y AdalFlow establecieron un conjunto de funcionalidades principales para trabajar a más bajo nivel con LLMs. En la tabla 1 se puede observar diferentes funcionalidades como la orquestación de prompts (Opt.) en donde se definen nodos/ transformaciones que van a ser ejecutadas en una secuencia de tareas. Evaluación (Eval.) que permite establecer criterios para poder evaluar el output para poder controlar el correcto funcionamiento del LLM en tiempo real o en intervalos de tiempo según complejidad y/o costo del control. Guardrails para poder interceptar entradas y/o salidas del LLM para poder controlar lo que se recibe e/o informa al usuario.

Existe una vacante en el campo emergente ingeniería de prompts aumentada (En inglés, Augmented Prompt engineering) que se define [29] como proceso interactivo entre los sistemas AI y los humanos en la que la AI genera outputs a partir de datos y reglas que aportan la intuición que tienen los humanos del conocimiento específico en un campo de dominio de aplicación que no tiene la AI. Bajo esta línea y contexto, en el presente trabajo se busca aportar una estructura de reglas con conocimiento de experto para aumentar el prompt Aumentación de prompt basados en reglas, (Au. Reglas para poder guiar al LLM mediante criterio e intuición proporcionado por expertos. Una sugerencia es adaptar y aplicar la metodología IDEAL[30] para sistemas expertos, que permite organizar un conjunto de reglas para resolver problemas orientados a metas lo que se alinea a mecanismos de aprendizaje por refuerzo aplicados en el entrenamiento de LLMs para poder resolver problemas con lenguaje natural.

**Tabla 1. Comparación de funcionalidades**

	Orq.	Au. Reglas	Guard rails	Eval.	Opt.
Presente trabajo	✓	✓	✓	✗	✗
DSPy	✓	✗	✗	✓	✓
OneTwo	✗	✗	✗	✓	✓
AdalFlow	✓	✗	✗	✗	✓

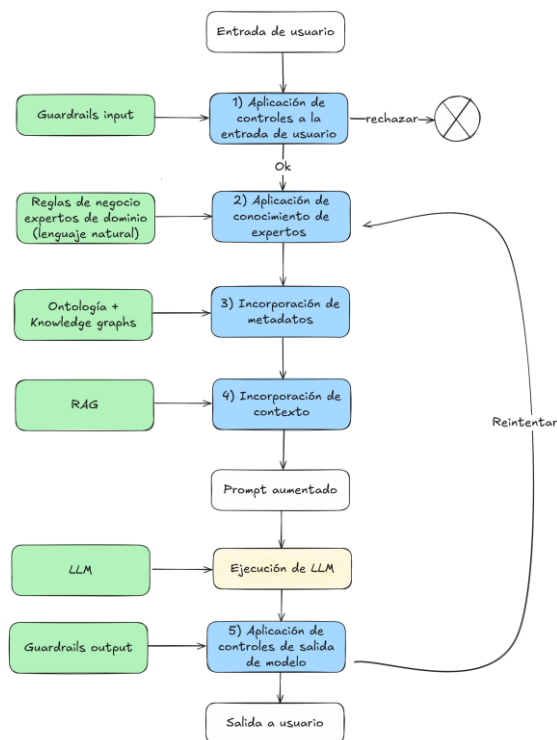
Las funcionalidades faltantes de Evaluación y optimización se podrían integrar como futuras líneas de trabajo mediante el uso de la biblioteca DSP en el prototipo de nuestro framework. En la próxima sección se explicará en detalle el template del prototipo de framework propuesto.

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



## Template de framework

El template de framework propuesto busca proponer pasos para estructurar la aumentación de prompts en la integración de LLMs pequeños con sistemas de software existentes como se puede observar en la figura 2. La secuencia de pasos consiste en: Paso 1 Incorporar input guardrails para aplicar controles de protección al input recibido desde el usuario para evitar inyecciones de prompts y jailbreaks. Paso 2 Incorporar reglas de negocio de expertos de dominio y de funcionamiento como reglas de prompt de sistema. Paso 3. Incorporación de metadatos mediante el uso de knowledge graph para enriquecer la aumentación de contexto del paso siguiente. Paso 4. Incorporación de información relevante de contexto mediante el uso de técnicas RAG. Paso 5. La aplicación de guardrails para controlar el output del LLM. En la próxima sección se explicará el caso de estudio utilizado.



**Fig. 2.** Pasos para la aumentación de prompts

## Caso de estudio

En el caso de aplicación, se trabajó en una prueba de concepto de un motor de aumentación integrado a un sistema de preguntas y respuestas sobre manufactura de nueva generación en el contexto de industria 4.0, particularmente en manufactura integrada por computadora aplicando CNC siguiendo los siguientes pasos. El caso consiste en aumentar las consultas del usuario (paso 1) mediante el uso de 10 reglas: 5 reglas que guían al LLM para emitir una respuesta y 5 reglas de conocimiento de experto (Paso 2). Luego, se extrae entidades encontradas en el prompt del usuario mediante la técnica NLP Named entity recognition (paso 3) para poder realizar una búsqueda de metadatos (paso 4) de contexto de un knowledge graph estructurado con ontologías existentes como

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



Context Ontology for Industry 4.0 (COInd4)[31][32][33] con datos creados de manera sintética para este caso de aplicación que permitan recuperar fragmentos relevantes para el contexto mediante la aplicación de RAG (naive RAG) (paso 5). Los fragmentos de texto vectorizados se obtuvieron de fuentes como [34] que guía cómo manejar un torno CNC. Por último, se realiza la ejecución del prompt en el LLM (paso 6). La aplicación de guardrails se deja indicada como mejor práctica indicados en las secciones 2.1 y 2.5, quedando fuera del alcance actual de la prueba de concepto realizada. El progreso de este caso de aplicación se puede ver en el anexo en [35].

**B. Actualizar Publicaciones, Desarrollos, Formación de RRHH, y demás actividades en el Banco de datos de actividades de CyT dentro de SIGEVA UNLAM, para que se vea reflejado en el informe dentro de la plataforma.**

Tipo de Producción	Trabajos en eventos C-T publicados
Título	Línea de investigación: Framework para explotación de modelos de IA generativos
Autor/es	Dr. Martin Becerra, Dr. Jorge Ierache, Mg. Pablo Soligo, Ing. Diego Sanz
Editorial	Comodoro Rivadavia: Universitaria de la Patagonia -EDUPA
Fecha	Abril, 2024
Situación	Publicado
DOI y/o Enlace/link (solo si está publicado)	<a href="#">ISBN 978-987-8352-57-2</a>

Tipo de Producción	Trabajos en eventos C-T publicados
Título	Avances en la Línea de investigación: Motor de aumentación de prompt modelos de IA generativos.
Autor/es	Dr. Martin Becerra, Dr. Jorge Ierache, Ing. Diego Sanz, Mg. Pablo Soligo,
Editorial	Universidad Nacional de Cuyo. Secretaría de Investigación, Internacionales y Posgrado
Fecha	Abril, 2025
Situación	Publicado
DOI y/o Enlace/link (solo si está publicado)	ISBN <a href="#">978-987-575-267-2</a>

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



Tipo de Producción	Trabajos en eventos C-T publicados
Título	Template de Motor de Aumentación de Prompts para Large Language Models
Autor/es	Dr. Martin Becerra, Dr. Jorge Ierache, Ing. Diego Sanz, Mg. Pablo Soligo,
Editorial	Universidad Nacional de Rio Negro
Fecha	Octubre, 2025
Situación	En prensa
DOI y/o Enlace/link (solo si está publicado)	

**C. Vinculación<sup>3</sup>:** Indicar conformación de redes, intercambio científico, etc. con otros grupos de investigación; con el ámbito productivo o con entidades públicas. Desarrolle en no más de dos (2) páginas.

**D. Otra información. Incluir toda otra información que se considere pertinente.**

### Referencias

1. GPT-4 Technical Report. Disponible en <https://arxiv.org/abs/2303.08774>. Accedido el 18/12/2025
2. OpenAI, ChatGPT. Disponible en <https://openai.com/blog/chatgpt>. Accedido el 18/12/2025
3. Meta, Llama 2: Open Foundation and Fine-Tuned Chat Models. Disponible en <https://arxiv.org/abs/2307.09288>. Accedido el 18/12/2025.
4. Google, Gemini. Disponible en <https://deepmind.google/technologies/gemini/#introduction>. Accedido el 18/12/2025.
5. Meta, The Llama 3 Herd of Models. Disponible en <https://arxiv.org/abs/2407.21783>. Accedido el 18/12/2025.
6. OpenAI GPT o3, Deliberative Alignment: Reasoning Enables Safer Language Models. Disponible en <https://arxiv.org/abs/2412.16339>. Accedido el el 18/12/2025.
7. Deepseek, R1 DeepSeek LLM: Scaling Open-Source Language Models with Longtermism. Disponible en <https://arxiv.org/abs/2401.02954>. Accedido el 18/12/2025.
8. Nvidia, What is Agentic IA?. Disponible en <https://blogs.nvidia.com/blog/what-is-agentic-ai/>. Accedido el 18/12/2025.
9. Anthropic, Building effective Agents. Disponible en <https://www.anthropic.com/research/building-effective-agents>. Accedido el 18/12/2025.
10. Wooldridge. M, Jennings N. Agent Technology: Foundations, Applications, and Markets. Springer; 1998. 2nd Printing 2002 ed. edición (1 mayo 2002). ISBN-13 : 978-3540635918
11. Russell, S. J., & Norvig, P. (2021). Artificial Intelligence: A Modern Approach (4th ed.). Pearson.

<sup>3</sup> Entendemos por acciones de “vinculación” aquellas que tienen por objetivo dar respuesta a problemas, generando la creación de productos o servicios innovadores y confeccionados “a medida” de sus contrapartes.

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



12. Lewis, P., et. al. (2021). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks.. Disponible en <https://arxiv.org/abs/2005.11401>. Accedido el 18/12/2025.
13. Feng, Z. et. al. (2024). EasyRAG: Efficient retrieval-augmented generation framework for automated network operations. Disponible en <https://arxiv.org/abs/2410.10315>. Accedido el 18/12/2025.
14. Yu, T. et. al. (2024). Auto-RAG: Autonomous retrieval-augmented generation for large language models. Disponible en <https://arxiv.org/abs/2411.19443>. Accedido el 18/12/2025.
15. Edge, D., et al. (2024). From Local to Global: A Graph RAG Approach to Query-Focused Summarization. Disponible en <https://arxiv.org/abs/2404.16130>. Accedido el 18/12/2025.
16. Wei, J. et. al. (2023). Chain-of-thought prompting elicits reasoning in large language models Disponible en <https://arxiv.org/abs/2201.11903>. Accedido el 18/12/2025.
17. Zhang, Z., et. al.. (2022). Automatic chain of thought prompting in large language models. Disponible en <https://arxiv.org/abs/2210.03493>. Accedido el 18/12/2025.
18. Yao, S. et. al. (2023). Tree of thoughts: Deliberate problem solving with large language models. Disponible en <https://arxiv.org/abs/2305.10601>. Accedido el 18/12/2025.
19. Besta, M. et al. (2023). Graph of thoughts: Solving elaborate problems with large language models (arXiv:2308.09687v2). Disponible en <https://arxiv.org/abs/2308.09687v2>. Accedido el 18/12/2025.
20. Ning, X. et. al. (2024). Skeleton-of-thought: Prompting LLMs for efficient parallel generation. Disponible en <https://arxiv.org/abs/2307.15337>. Accedido el 18/12/2025.
21. Zhou, Y. et. al. (2023). Thread of thought: Unraveling chaotic contexts. Disponible en <https://arxiv.org/abs/2311.08734>. Accedido el 18/12/2025.
22. Zhang, A. et. al. (2023). Multimodal chain-of-thought reasoning in language models disponible en <https://arxiv.org/abs/2302.00923>. Accedido el 18/12/2025.
23. Becerra M., Ierache J, Ing Sanz Diego, Mg. Soligo Pablo. Template de Motor de Aumentación de Prompts para Large Language Models. XXXI Congreso Argentino de Ciencias de la Computación. En prensa.
24. IBM, Knowledge graphs. Disponible en <https://www.ibm.com/topics/knowledge-graph>. Accedido el 18/12/2025.
25. Langchain, The platform for reliable agents. Disponible en <https://www.langchain.com/>. Accedido el 18/12/2025.
26. DSPy, DSPy is a declarative framework for building modular AI software. Disponible en <https://dspy.ai/>. Accedido el 18/12/2025.
27. Google Deepmind, OneTwo Python library designed to simplify interactions with LLM, primarily aimed at researchers. Disponible en <https://github.com/google-deepmind/onetwo>. Accedido el 18/12/2025.
28. SylphAI-Inc, AdalFlow is a PyTorch-like library to build and auto-optimize any LM workflows. Disponible en <https://github.com/SylphAI-Inc/AdalFlow>. Accedido el 18/12/2025.
29. Anand Nayyar, et. al. Chapter 11 - Future trends in large language models and prompt engineering,

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



Mastering Prompt Engineering, Morgan Kaufmann, 2025, Pages 309-336, ISBN 9780443339042, <https://doi.org/10.1016/B978-0-443-33904-2.00009-4>.

30. Gómez, A., Juristo, N., Montes, C., Pazos, J. "Ingeniería del Conocimiento". Centro de Estudios Ramón Areces. S.A., Madrid, 1997.
31. Giustozzi F. et. Al(2018). Context Modeling for Industry 4.0: an Ontology-Based Proposal. Disponible en <https://www.sciencedirect.com/science/article/pii/S1877050918312791>. Accedido el 18/12/2025
32. 2. Giustozzi F. et. Al (2023). A semantic framework for condition monitoring in Industry 4.0 based on evolving knowledge bases. Disponible en [https://www.researchgate.net/publication/374512372\\_A\\_semantic\\_framework\\_for\\_condition\\_monitoring\\_in\\_Industry\\_40\\_based\\_on\\_evolutionary\\_knowledge\\_bases](https://www.researchgate.net/publication/374512372_A_semantic_framework_for_condition_monitoring_in_Industry_40_based_on_evolutionary_knowledge_bases). Accedido el 18/12/2025.
33. 3. Giustozzi F. et. al (2024). Context Ontology for Industry 4.0 (COInd4). Disponible en [https://gitlab.insa-rouen.fr/fgiustozzi/STeAMINg-Ontology/-/tree/master?ref\\_type=heads](https://gitlab.insa-rouen.fr/fgiustozzi/STeAMINg-Ontology/-/tree/master?ref_type=heads). Accedido el 18/12/2025.
34. Bueno D., Gonzalez J (2015). Guía de aprendizaje para manejo de torno cnc wabeco CC-D6000 E. Disponible en <https://core.ac.uk/download/pdf/71398373.pdf> . Accedido el 18/12/2025.
35. Becerra M., lerache J (2025). Avances caso de aplicación Industria 4.0. Disponible en <https://drive.google.com/drive/folders/1Og3lxB8Hv1k0EpCfS1gkXm-JEB-xLpjG?usp=sharing> . Accedido el 18/12/2025.

## E. Cuerpo de anexos:

- Anexo I:
  - FPI-013: Evaluación de alumnos integrantes. (si corresponde)
  - FPI-014: Comprobante de liquidación y rendición de viáticos. (si corresponde)
  - FPI-015: Rendición de gastos del proyecto de investigación acompañado de las hojas foliadas con los comprobantes de gastos.
  - FPI-038: Formulario de reasignación de fondos en Presupuesto.
- Anexo II: FPI 017<sup>4</sup> Alta patrimonial de los bienes adquiridos con presupuesto del proyecto
- Nota justificando baja de integrantes del equipo de investigación.

## F1. Observaciones

El Dr Martin Becerra cumple funciones de investigador formado, además realiza tareas de codirección del proyecto, con la finalidad de enriquecer su formación para futuras direcciones de proyectos (su categorización está pendiente de evaluación, si bien sus antecedentes resultan equivalentes para una Categoría III).

<sup>4</sup> Solo ante la presentación del Informe Final

<b>Código</b>	FPI-009
<b>Objeto</b>	Guía de elaboración de Informe de avance/final de proyecto
<b>Usuario</b>	Director de proyecto de investigación
<b>Autor</b>	Secretaría de Ciencia y Tecnología de la UNLaM
<b>Versión</b>	8
<b>Vigencia</b>	19/09/2024



Dr Jorge Ierache

Lugar y fecha : San Justo 22 de Diciembre 2025