UNIVERSIDAD DE ANTIOQUIA

Arquitectura de Referencia para IA Generativa Empresarial: Blueprint para el despliegue de soluciones escalables, seguras y agentiles

Juan Sebastián Ramírez Solarte

Proyecto de Investigación para optar al título de Ingeniero de Sistemas

Modalidad de Práctica
Proyecto de Investigación
Grupo de Investigación ITOS

Asesora
Diana Margot López Herrera (MsC en Ingeniería)

Universidad de Antioquia
Facultad de Ingeniería
Ingeniería de Sistemas
Medellín
2026

UNIVERSIDAD DE ANTIOQUIA

Cita	(Ramírez Solarte, 2026)
Referencia **Estilo APA 7 (2020)**	Ramirez Solarte, J. (225). *Arquitectura de Referencia para IA Generativa Empresarial.: Blueprint para el despliegue de soluciones escalables, seguras y agentiles* [Seleccione modalidad de grado]. Universidad de Antioquia, Medellín.



Grupo de Investigación ITOS Ingeniería y Tecnologías de las Organizaciones y la Sociedad.

Repositorio Institucional: http://bibliotecadigital.udea.edu.co

Universidad de Antioquia - www.udea.edu.co

El contenido de esta obra corresponde al derecho de expresión de los autores y no compromete el pensamiento institucional de la Universidad de Antioquia ni desata su responsabilidad frente a terceros. Los autores asumen la responsabilidad por los derechos de autor y conexos.

Dedicatoria

A mis padres, por su apoyo incondicional y por ser los cimientos sobre los cuales he construido mi vida.

A mi esposa, quien no solo ha sido mi soporte diario y mi compañera incondicional, sino la razón fundamental y la inspiración que me motivó a iniciar y culminar esta carrera. Este logro es tan tuyo como mío.

Agradecimientos

En primer lugar, expreso mi más sincero agradecimiento a mi asesora, la profesora Diana Margot López Herrera, quien me acogió desde el inicio de mi formación académica. Su orientación no se limitó a las aulas; el impulso y la confianza que brindó a mi trayectoria laboral fueron determinantes para mi crecimiento como Ingeniero de Sistemas. Gracias por su guía constante en la realización de este proyecto.

A mis mentores y colegas que me han acompañado a lo largo de mis años de experiencia en la industria TI. Sus enseñanzas prácticas y el conocimiento compartido han enriquecido mi visión profesional y se ven reflejados en la madurez técnica de este trabajo.

Finalmente, a mi esposa y a mi familia, por su paciencia infinita y por creer en mí durante las largas jornadas de estudio y trabajo. Gracias por estar siempre ahí.

Tabla de contenido

Resumen..... 10
Abstract..... 11
1. Introducción..... 12
2. Objetivos..... 14
2.1 Objetivo general..... 14
2.2 Objetivos específicos..... 14
3. Marco teórico..... 15
3.1 Fundamentos de la IA Generativa y la Necesidad de Estandarización..... 15
3.2 Evolución del Ciclo de Vida: De MLOps a GenAIOps..... 15
3.3 Interoperabilidad y Ecosistema de Orquestación..... 16
3.4 Patrones Arquitectónicos y Metodologías de Desarrollo..... 18
3.5 Seguridad, Gobernanza y Cumplimiento Normativo..... 20
3.6 Servicios en la Nube y Arquitecturas Agnósticas..... 22
3.7 Observabilidad y Criterios de Evaluación..... 22
4. Metodología..... 23
4.1 Fase 1: Revisión y Fundamentación del Estado del Arte..... 24
4.2 Fase 2: Análisis Comparativo y Síntesis de Hallazgos..... 25
4.3 Fase 3: Diseño y Modelado de la Arquitectura de Referencia..... 25
4.4 Fase 4: Propuesta de validación mediante Prueba de Concepto (PoC)..... 25
4.5 Resultados Esperados y Conclusiones del Trabajo..... 26
5. Análisis de resultados..... 27
5.1. Análisis de la Literatura Seleccionada y Criterios de Evaluación..... 27
5.1.1. Estructura de la Matriz de Calidad Metodológica y Académica..... 27
5.1.2. Estructura de la Matriz de Relevancia Arquitectónica y Técnica..... 29
5.1.3. Contraste de Resultados: El Dilema entre Rigor y Relevancia Técnica..... 31
5.1.4. Listado de Referencias Evaluadas..... 31
5.2 Fundamentación Metodológica de la Arquitectura: Sinergia TOGAF y arc42..... 33
5.2.1 Alineación Estratégica..... 33
5.2.2 Correspondencia de Artefactos y Trazabilidad..... 34
5.3 Visión de Arquitectura..... 34
5.3.1 Organización Funcional por Capas..... 35
5.3.2 Proposición de Valor y KPIs por Stakeholder..... 36
5.3.2 Capacidades de Negocio Habilitadas..... 37
5.4 Stakeholders y Preocupaciones..... 37
5.5 Objetivos de Calidad y Requisitos..... 38
5.5.1 Requisitos No Funcionales (RNF)..... 39
5.6 Vista de Contexto..... 40

5.7.1 Interacciones Clave....40
5.7 Vista de Contenedores y Datos....41
5.7.1 Componentes del Orquestador de Agentes....43
5.7.2 Componentes del Servicio RAG....44
5.7.3 Componentes del Model Gateway....46
5.7.4 Componentes de Security Guardrails....46
5.7.5 Componentes de Infraestructura y Datos....47
5.8 Vista de Despliegue y Patrones....48
5.8.1 Escalado elástico de inferencia....49
5.8.2 Conectividad privada (Egress control)....50
5.8.3 Persistencia políglota....50
5.9 Decisiones Arquitectónicas (ADRs)....50
5.10 Perspectivas de Calidad y Validación....51
5.10.1 Escenarios de Seguridad (Security Scenarios)....51
5.10.2 Escenarios de Fiabilidad en la Respuesta....52
5.10.3 Escenarios de Operación y Evolución Técnica....52
5.11 Plan de Transición: Riesgos, Gaps y Roadmap....53
5.11.1 Riesgos y Deuda Técnica....53
5.11.2 Análisis de Gaps y Hoja de Ruta....56
5.12 Matriz de Trazabilidad arc42 vs TOGAF....57
5.13 Propuesta de PoC: Asistente de Reclamaciones en Seguros....58
5.13.1 Propósito....58
5.13.2 Alcance....58
5.13.3 Componentes Implementados....59
5.13.4 Criterios de Éxito....59
5.13.5 Plan de Pruebas....59
5.13.6 Entregables....60
5.13.7 Valor Esperado....60
6. Conclusiones y recomendaciones....61
Referencias....63
Anexos....69

Lista de tablas

Tabla 1 Dimensión de Definición (PICO y Objetivos)..... 28
Tabla 2 Dimensión de Reproducibilidad y Búsqueda..... 28
Tabla 3 Dimensión de Protocolo y Selección (PRISMA)..... 28
Tabla 4 Dimensión de Calidad, Sesgo y Puntuación Final..... 28
Tabla 5 Dimensión de Diseño Arquitectónico y Estructural..... 30
Tabla 6 Dimensión de Patrones Modernos y Soporte Técnico..... 30
Tabla 7 Dimensión de Atributos de Calidad Empresarial..... 30
Tabla 8 Dimensión de Madurez, Implementación, Gobernanza y Puntuación Final..... 30
Tabla 9 KPI de interés para stakeholders..... 36
Tabla 10 Rol e interés de los stakeholders..... 38
Tabla 11 Registros de Decisión de Arquitectura..... 41
Tabla 12 Descripción de Contenedores..... 42
Tabla 13 Registros de Decisión de Arquitectura..... 51
Tabla 14 Riesgos de Seguridad..... 54
Tabla 15 Riesgos Operativos..... 55
Tabla 16 Mapeo arc42 - TOGAF..... 57
Tabla 17 PoC Componentes..... 59
Tabla 18 PoC Criterios de éxito..... 59

Lista de figuras

Figura 1 Blueprint de la Arquitectura de Referencia Empresarial..... 35
Figura 2 Diagrama de Contexto (C4)..... 40
Figura 3 Diagrama de Contenedores (C4)..... 42
Figura 4 Orquestador de Agentes..... 43
Figura 5 Servicio RAG..... 45
Figura 6 Model Gateway..... 46
Figura 7 Security Guardrails..... 47
Figura 8 Infraestructura y Datos..... 48
Figura 9 Vista de Despliegue y Patrones..... 48

Siglas, acrónimos y abreviaturas

ACL	Access Control List (Lista de Control de Acceso).
ADM	Architecture Development Method (Método de Desarrollo de Arquitectura - TOGAF).
ADR	Architecture Decision Record (Registro de Decisión de Arquitectura).
AI / IA	Artificial Intelligence / Inteligencia Artificial.
AIMS	Artificial Intelligence Management System (Sistema de Gestión de IA).
APA	American Psychological Association.
API	Application Programming Interface (Interfaz de Programación de Aplicaciones).
AWS	Amazon Web Services.
C4	Model for visualizing software architecture (Modelo de visualización de arquitectura).
CISO	Chief Information Security Officer (Director de Seguridad de la Información).
Cms.	Centímetros.
CRM	Customer Relationship Management.
DLP	Data Loss Prevention (Prevención de Pérdida de Datos).
DoS	Denial of Service (Denegación de Servicio).
ERIC	Education Resources Information Center.
ERP	Enterprise Resource Planning (Planificación de Recursos Empresariales).
Esp.	Especialista.
GDPR	General Data Protection Regulation.
GenAI	Generative Artificial Intelligence (AI Generativa).
GenAIOps	Generative AI Operations.
gRPC	Google Remote Procedure Call.
HPA	Horizontal Pod Autoscaler.

HTTP	Hypertext Transfer Protocol.
IdP	Identity Provider (Proveedor de Identidad).
ISO	International Organization for Standardization.
KPI	Key Performance Indicator (Indicador Clave de Desempeño).
LLM	Large Language Models (Modelos de Lenguaje Extensos).
MLOps	Machine Learning Operations.
MP	Magistrado Ponente.
MSc	Magister Scientiae (Magíster en Ciencias).
NIST	National Institute of Standards and Technology.
OIDC	OpenID Connect.
OWASP	Open Web Application Security Project.
Párr.	Párrafo.
PBQ-SF	Personality Belief Questionnaire Short Form.
PhD	Philosophiae Doctor (Doctorado).
PII	Personally Identifiable Information (Información de Identificación Personal).
PoC / POC	Proof of Concept (Prueba de Concepto).
PostDoc	Postdoctorado.
RAG	Retrieval-Augmented Generation (Generación Aumentada por Recuperación).
RBAC	Role-Based Access Control (Control de Acceso Basado en Roles).
REST	Representational State Transfer.
ROI	Return on Investment (Retorno de la Inversión).
SAML	Security Assertion Markup Language.
SLO	Service Level Objective (Objetivo de Nivel de Servicio).
SQL	Structured Query Language.
TCO	Total Cost of Ownership (Costo Total de Propiedad).

TOGAF	The Open Group Architecture Framework.
TTFT	Time to First Token (Tiempo hasta el primer token).
UdeA	Universidad de Antioquia.
VPC	Virtual Private Cloud.
VPN	Virtual Private Network.
WSS	WebSocket Secure.

Resumen

Esta investigación propone una arquitectura de referencia estandarizada (blueprint) para el diseño y operación de soluciones de IA Generativa Empresarial que sean escalables, seguras e interoperables. El estudio aborda la problemática actual donde la adopción acelerada de esta tecnología requiere de marcos técnicos sólidos, lo que genera silos tecnológicos, riesgos de seguridad (como la inyección de prompts) y dependencia de proveedores (vendor lock-in).

La metodología sigue un enfoque aplicado en cuatro fases:

- Fase 1: Revisión del estado del arte y ecosistemas como LangChain y LlamaIndex.
- Fase 2: Análisis comparativo de patrones arquitectónicos.
- Fase 3: Modelado de la arquitectura bajo el estándar arc42, integrando componentes de Agentic AI y servicios RAG (Retrieval-Augmented Generation).
- Fase 4: Propuesta de validación mediante una Prueba de Concepto (PoC).

Los resultados destacan una plataforma modular que utiliza un "Model Gateway" para garantizar el agnosticismo tecnológico y "Security Guardrails" para la protección de datos sensibles. El trabajo concluye que esta arquitectura permite transformar la innovación disruptiva en una capacidad operativa técnica y económicamente sostenible para las organizaciones.

Palabras clave: IA Generativa, arquitectura de referencia, RAG, agentes inteligentes, seguridad LLM, interoperabilidad.

Abstract

This research proposes a standardized reference architecture (blueprint) for the design and operation of Enterprise Generative AI solutions, ensuring they are scalable, secure, and interoperable. The study addresses the current gap where accelerated AI adoption lacks solid technical frameworks, leading to technological silos, security risks (such as prompt injection), and vendor lock-in.

The methodology follows an applied approach in four phases:

- Phase 1: Review of the state-of-the-art ecosystems like LangChain and LlamaIndex.
- Phase 2: Comparative analysis of architectural patterns.
- Phase 3: Architectural modeling using the arc42 standard, integrating Agentic AI components and RAG (Retrieval-Augmented Generation) services.
- Phase 4: Validation proposal using a Proof of Concept (PoC).

The results highlight a modular platform utilizing a "Model Gateway" to ensure technological agnosticism and "Security Guardrails" for protecting sensitive data. The work concludes that this architecture enables the transformation of disruptive innovation into a technically and economically sustainable operational capability for organizations.

*Keywords***:** Generative AI, reference architecture, RAG, intelligent agents, LLM security, interoperability.

1. Introducción

La Inteligencia Artificial Generativa (IA Generativa) se ha consolidado como un paradigma disruptivo en la transformación digital, permitiendo a las organizaciones de desarrollo de software y áreas de TI reducir costos operativos y el tiempo de entrega al mercado mediante la automatización de flujos de trabajo complejos. No obstante, esta adopción acelerada ha superado la capacidad de estandarización técnica y arquitectónica, lo que conduce frecuentemente a implementaciones aisladas que crean silos tecnológicos y limitan la integración con los sistemas existentes. Puesto que muchas de estas iniciativas surgen como respuestas tácticas a necesidades específicas, la necesidad de un marco de referencia sólido compromete la escalabilidad y la sostenibilidad de las soluciones a largo plazo, obligando a las empresas a enfrentar una constante reinvención de soluciones en cada nuevo proyecto.

Ahora bien, la necesidad de un blueprint estructurado no solo impacta la interoperabilidad, sino que expone a las organizaciones a riesgos críticos de seguridad y gobernanza de datos, tales como la inyección de prompts, la fuga de información sensible y el envenenamiento de datos de entrenamiento. Asimismo, la dependencia excesiva de proveedores tecnológicos o vendor lock-in y la dificultad para transitar desde prácticas de MLOps hacia procesos formales de GenAIOps impiden que las implementaciones superen la etapa experimental o de prueba de concepto. Teniendo en cuenta lo anterior, el problema central de este trabajo radica en la necesidad de una arquitectura de referencia estandarizada que permita diseñar y operar soluciones de IA Generativa de manera segura y agnóstica, integrando las mejores prácticas de ingeniería de software con la gestión de modelos de lenguaje.

En este contexto, el presente trabajo tiene como objetivo proponer una arquitectura de referencia que sirva como un plano guía para el desarrollo de soluciones empresariales de IA, proporcionando lineamientos claros para la orquestación de agentes inteligentes y la mitigación de vulnerabilidades técnicas. El estudio se orienta a resolver interrogantes sobre los desafíos de gobernanza predominantes, los principios arquitectónicos que garantizan la fiabilidad y la manera en que un marco de referencia puede facilitar la transición hacia plataformas operativas robustas. Finalmente, la justificación de esta propuesta reside en la necesidad de transformar la innovación disruptiva en una capacidad operativa técnica y económicamente sostenible, cerrando la brecha entre el rápido avance tecnológico y la falta de modelos arquitectónicos consolidados mediante un enfoque que articula el diseño teórico con la validación empírica.

Como cierre del proceso, se llevará a cabo el diseño de una Prueba de Concepto aplicada a un caso de uso empresarial práctico, con el fin de identificar la viabilidad y efectividad de la arquitectura diseñada en un entorno real, encontrando un valor tangible en la resolución de desafíos técnicos complejos.

2. Objetivos

2.1 Objetivo general

Proponer una arquitectura de referencia que sirva como un blueprint estandarizado para la creación de soluciones empresariales de IA generativa, garantizando que sean escalables, seguras e interoperables al integrar las mejores prácticas de ingeniería de software y gobernanza de datos.

2.2 Objetivos específicos

- Identificar los principales desafíos técnicos y de negocio que enfrentan las empresas al implementar soluciones de IA generativa, enfocándose en integración, escalabilidad y gobernanza.
- Estudiar estándares internacionales (ISO, NIST, IEEE) para determinar su aplicabilidad y guiar el diseño de una arquitectura robusta y conforme a normativas.
- Modelar los componentes clave de la arquitectura de referencia, definiendo las capas funcionales, sus interacciones y las responsabilidades para asegurar un diseño modular.
- Establecer directrices de ingeniería (MLOps, DevOps) adaptadas al ciclo de vida de la IA generativa, para optimizar el desarrollo, despliegue y mantenimiento de las soluciones.
- Validar la viabilidad y eficacia de la arquitectura a través de una Prueba de Concepto (PoC) en un caso de uso empresarial práctico, para ser ejecutado en un posterior trabajo.

3. Marco teórico

3.1 Fundamentos de la IA Generativa y la Necesidad de Estandarización

La Inteligencia Artificial Generativa (IA Generativa) ha emergido como un paradigma disruptivo en la transformación digital, fundamentándose en modelos capaces de generar contenido nuevo y coherente —como texto, imágenes o código— a partir de patrones aprendidos de grandes volúmenes de datos. Este enfoque es un habilitador estratégico para la innovación empresarial, impulsando casos de uso en automatización, análisis avanzado y optimización de la toma de decisiones (McKinsey & Company 2024; Sengar et al. 2024; Stryker y Scapicchio 2024; Weiss 2023).

Sin embargo, su rápida adopción ha superado la consolidación de metodologías estandarizadas, para el desarrollo de aplicaciones robustas de gran impacto a nivel de las empresas, conservando los estándares de calidad ya establecidos como escalabilidad, mantenibilidad entre otros. La necesidad de una arquitectura de referencia, entendida como un blueprint estandarizado, y los escasos intentos de la misma conduce a implementaciones aisladas que crean silos tecnológicos y limitan la integración. La interoperabilidad, la escalabilidad y la seguridad se identifican como desafíos fundamentales que una arquitectura robusta debe abordar para maximizar el valor de estas tecnologías y garantizar un crecimiento sostenido más allá de una Prueba de Concepto (PoC) (Joshi 2025; Stryker y Scapicchio 2024).

3.2 Evolución del Ciclo de Vida: De MLOps a GenAIOps

La gestión del ciclo de vida de la IA generativa ha evolucionado desde los principios de MLOps (Machine Learning Operations, se refiere a un conjunto de prácticas que optimizan y automatizan el ciclo de vida de los modelos de aprendizaje automático) hacia GenAIOps (Generative AI Operations, es un marco emergente para operaciones con IA generativa), una especialización que aborda las características únicas de los modelos generativos. Mientras MLOps se centra en el ciclo de entrenamiento y despliegue, GenAIOps enfatiza la selección de modelos preentrenados, la ingeniería de prompts, el ajuste fino (fine-tuning) y patrones

avanzados como Retrieval Augmented Generation (RAG) (Stryker y Scapicchio 2024; World Economic Forum 2024).

Esta evolución introduce nuevos desafíos operacionales: la gestión y versión de prompts, la evaluación continua de las respuestas generadas, la optimización de modelos mediante técnicas como cuantización y destilación, y el monitoreo de la calidad y seguridad de la salida en tiempo real. Desafíos que con una arquitectura de referencia estarían en gran parte cubiertos ya que esta debe incorporar directrices de ingeniería que sistematizan estas prácticas (Mariani y Dwivedi 2024; Rios-Campos et al. 2023).

3.3 Interoperabilidad y Ecosistema de Orquestación

La interoperabilidad constituye un eje fundamental en la implementación de soluciones empresariales basadas en IA generativa, ya que asegura la integración de diversos modelos y sistemas en arquitecturas híbridas y multi-cloud. Según Wolfenstein (Wolfenstein 2024), la colaboración entre modelos especializados de IA dentro de un mismo ecosistema incrementa el rendimiento y la capacidad de adaptación de las organizaciones, al permitir que cada modelo cumpla funciones específicas en sinergia con los demás. De igual manera, el IBM Institute for Business Value (IBM 2024) subraya que la adopción acelerada de IA generativa en las empresas requiere marcos de gobernanza y estándares de interoperabilidad para evitar fragmentación tecnológica y maximizar el retorno de inversión. Gartner (Brinson 2025) proyecta que para 2027, el 80% de los equipos de ingeniería de software incorporarán IA generativa en sus procesos, lo que evidencia que la interoperabilidad será indispensable para escalar estas soluciones y convertirlas en motores de innovación y transformación digital. En síntesis, la interoperabilidad no solo facilita la integración técnica, sino que también habilita la creación de soluciones empresariales sostenibles, seguras y competitivas en el mercado global.

La proliferación de soluciones de IA generativa está generando un paisaje fragmentado de ecosistemas incompatibles, con protocolos y formatos de datos distintos que crean barreras para la interoperabilidad entre agentes. Esta fragmentación impone costos de ingeniería, limita la reutilización de componentes y amenaza la escalabilidad. La propuesta de marcos como Web of Agents busca establecer un estándar mínimo para la interoperabilidad colaborativa, basado en mensajería, gestión de estado y descubrimiento de agentes, siguiendo el éxito de protocolos abiertos como HTTP (Azure Architecture Center 2025; Mariani y Dwivedi 2024).

Los frameworks de orquestación son marcos conceptuales y técnicos que permiten coordinar, automatizar y gestionar múltiples servicios, aplicaciones y procesos en entornos tecnológicos complejos, especialmente en la nube y arquitecturas distribuidas. Su propósito es garantizar que los recursos funcionen de manera integrada, eficiente y escalable. La orquestación en TI se entiende como la ejecución ordenada y coordinada de tareas de automatización, garantizando que procesos como la implementación, configuración y gestión de aplicaciones se realicen en el orden correcto y con mínima intervención humana .

Un framework de orquestación es un conjunto de herramientas, directrices y componentes que permiten integrar servicios heterogéneos, definir flujos de trabajo, monitorizar rendimiento y asegurar la interoperabilidad en entornos multi-cloud o híbridos. En el ámbito de la inteligencia artificial, los frameworks de orquestación también se aplican para gestionar modelos de lenguaje de gran tamaño (LLM), facilitando su encadenamiento, monitoreo y optimización en aplicaciones complejas. Algunas de sus características son: automatización avanzada, integración de servicios, escalabilidad, monitorización y control, flexibilidad. Lo cual facilita la transformación digital al reducir tiempos de despliegue y mejorar la eficiencia operativa, la gobernanza de datos y servicios, al asegurar trazabilidad y cumplimiento normativo y en el caso de la IA, permiten que modelos complejos se integren en aplicaciones reales con mayor confiabilidad.

Algunos de los frameworks de orquestación en el ámbito de la IA son:

- **LangChain**: Se distingue por su enfoque en cadenas de tareas secuenciales, ideal para flujos de trabajo complejos que requieren múltiples pasos lógicos y una gestión robusta del contexto.
- **LlamaIndex**: Se especializa en aplicaciones centradas en la recuperación de datos (RAG), ofreciendo técnicas avanzadas de indexación para integrar eficientemente datos privados y públicos.
- **Semantic Kernel:** Adopta un diseño modular donde las funciones de IA operan como bloques de código reutilizables (skills), facilitando la integración en aplicaciones existentes y flujos de trabajo dinámicos y adaptativos (Lamatic Labs 2025; Rios-Campos et al. 2023; Turing 2026).

3.4 Patrones Arquitectónicos y Metodologías de Desarrollo

Los patrones de software tradicionales son soluciones recurrentes y probadas para problemas comunes en el diseño y desarrollo de sistemas, que permiten mejorar la reutilización, la mantenibilidad y la escalabilidad del software. Entre los más reconocidos se encuentran el Singleton, Factory, Observer, Strategy, Adapter y Model-View-Controller (MVC), ampliamente documentados por Gamma, Helm, Johnson y Vlissides (Gamma 1995) en su obra seminal Design Patterns: Elements of Reusable Object-Oriented Software. Sin embargo, en el contexto de la IA generativa aplicada a entornos empresariales, estos patrones requieren adaptaciones significativas. Según (Katreddy 2023), la integración de modelos de lenguaje de gran escala (LLMs) introduce nuevas necesidades de orquestación, manejo de datos no estructurados y control de sesgos, lo que obliga a extender patrones clásicos hacia arquitecturas más dinámicas y distribuidas. Asimismo, Sarker et al. (Sarker et al. 2020) destacan que la interoperabilidad y la gobernanza de datos en sistemas inteligentes demandan patrones híbridos que combinen principios tradicionales con enfoques modernos como pipeline orchestration y event-driven architectures, para que las soluciones sean robustas, escalables y alineadas con estándares de calidad y ética tecnológica.

Por ejemplo, los microservicios permiten descomponer sistemas complejos en servicios especializados, facilitando la escalabilidad individual y la tolerancia a fallos. Las arquitecturas orientadas a eventos son especialmente compatibles con el patrón RAG (Generación Aumentada por Recuperación, es un enfoque en inteligencia artificial que combina modelos de lenguaje con datos externos para generar respuestas más precisas y relevantes), al permitir la sincronización en tiempo real de datos en bases de datos vectoriales (Sistemas diseñados para almacenar, gestionar e indexar datos representados como vectores de alta dimensión. Estos vectores, generados a partir de modelos de machine learning (ML), capturan características latentes de datos no estructurados como texto, imágenes, audio o video, permitiendo búsquedas rápidas y eficientes basadas en similitud.) para mantener los modelos actualizados, situación indispensable para los sistemas de datos en IA.

El principio de modularidad, es otra característica fundamental en el contexto de la IA Generativa empresarial, la modularidad se define como la capacidad de descomponer el sistema en unidades funcionales independientes y desacopladas, que se comunican a través de interfaces estandarizadas (APIs). Según Gao et al. (Gao et al. 2024), la evolución hacia el "RAG Modular"

permite que las funciones de recuperación (retrieval), procesamiento de datos y generación se traten como módulos autónomos que pueden optimizarse o sustituirse de manera independiente.

Este principio se manifiesta a través de componentes desacoplados e intercambiables. Una evolución pragmática es el monolito modular, que combina la simplicidad operacional de un monolito con la organización lógica de los microservicios, donde la lógica de la aplicación está intrínsecamente ligada a un proveedor específico (como OpenAI), permitiendo una mayor resiliencia ante el vendor lock-in. Esta arquitectura elimina la latencia de red y simplifica las transacciones, ofreciendo los beneficios de la modularidad sin la complejidad operacional de un sistema distribuido (Confluent s. f.; Microsoft 2025).

Basado en los principios de arquitectura de software aplicados a la IA (Richards y Ford 2020), las características de la modularidad en GenAI son: Desacoplamiento, encapsulamiento, composición de pipelines, aislamiento.

Desacoplamiento de la Capa de Inferencia (Swap-ability), que permite intercambiar el LLM (por ejemplo, pasar de GPT-4 a Llama 3) sin alterar la lógica de negocio ni la ingesta de datos. Esto se logra mediante capas de abstracción u orquestadores como LangChain o LlamaIndex, ya mencionados anteriormente.

Encapsulamiento de la Base de Datos Vectorial. lo que implica que almacenamiento de representaciones matemáticas de los datos (embeddings) debe ser un módulo independiente. Esto permite escalar el almacenamiento de conocimiento corporativo independientemente del poder de cómputo del modelo.

Composición de Pipelines (Chainability) que representa la capacidad de encadenar diferentes tareas como si fueran bloques de construcción. Cada eslabón de la cadena es un módulo con entradas y salidas definidas.

Aislamiento de "Guardrails" y Gobernanza, que constituye la seguridad y el filtrado de respuestas (evitar contenido tóxico o fuga de datos) actúan como un módulo de "middleware" que intercepta la comunicación antes de que llegue al usuario final.

La investigación reciente subraya que la modularidad es el antídoto contra la rigidez de los modelos fundacionales. Diaz-De-Arcaya et al. (2024) introducen el concepto de LLMOps, donde la modularidad facilita el ciclo de vida de la IA mediante la separación de la preparación de datos, el ajuste fino (*fine-tuning*) y el despliegue. Además, el patrón de Modular RAG propone que la recuperación de información ya no es un proceso lineal, sino un ecosistema de módulos de

búsqueda que pueden incluir bases de datos de grafos, motores de búsqueda semántica y SQL, operando de manera coordinada (Gao et al. 2024).

3.5 Seguridad, Gobernanza y Cumplimiento Normativo

La gobernanza de datos en el contexto de la IA se define como el marco de responsabilidades y procesos que aseguran la calidad, integridad y disponibilidad del activo dato (Abraham, Schneider, y vom Brocke 2019). En arquitecturas de GenAI, esto evoluciona hacia la Gobernanza Centrada en Datos (Data-Centric AI), donde la trazabilidad y la procedencia (lineage) son vitales para mitigar sesgos en los modelos de lenguaje. Según (Xiaoyong et al. 2023), una gobernanza robusta debe gestionar el ciclo de vida del dato desde su ingesta en bases de datos vectoriales hasta su uso en procesos de Fine-tuning o RAG (Retrieval-Augmented Generation). Lo que requiere un marco de gobernanza al interior de la organización, que sea usada por la arquitectura de referencia que regule los desarrollo con GenAI; al tiempo que dan cuenta del seguimiento, evolución y uso de dichos desarrollos.

La seguridad en GenAI vista desde la gobernanza, trasciende la ciberseguridad tradicional al introducir vectores de ataque específicos como la Inyección de Prompts (Prompt Injection) y el Envenenamiento de Datos (Data Poisoning). Das, Amini, y Wu (2024) señalan que la arquitectura debe integrar capas de validación que actúen como "cortafuegos semánticos". Además, la implementación de protocolos de Privacidad Diferencial y Encriptación Homomórfica es fundamental para garantizar que los datos sensibles de la empresa no sean memorizados o filtrados por el modelo durante la inferencia (Al-Rubaie y Chang 2019).

La gobernanza también debe incluir la normatividad legal, el referente indiscutible es la Ley de IA de la Unión Europea (EU AI Act). Este marco clasifica los sistemas según su nivel de riesgo, imponiendo obligaciones estrictas de transparencia para los modelos fundacionales (Parlamento Europeo 2024) y una alineación con las directrices del NIST AI Risk Management Framework en Estados Unidos, que prioriza la confiabilidad y la explicabilidad (Tabassi 2023). A diferencia de los sistemas tradicionales, la seguridad en una arquitectura de GenAI debe integrarse en la capa de orquestación y en la interacción con el modelo. Según, la arquitectura debe implementar módulos de "Guardrails" (Soportes de Seguridad) que actúen como filtros bidireccionales. Previniendo la Inyección de Prompts (Prompt Injection), donde un usuario intenta saltarse las

restricciones del modelo mediante instrucciones maliciosas y filtrando alucinaciones, contenido tóxico o la revelación inadvertida de datos sensibles que el modelo pudo haber memorizado durante su entrenamiento o recuperado vía RAG (Retrieval-Augmented Generation).

Uno de los componentes críticos de la arquitectura de referencia es el almacenamiento vectorial. La seguridad aquí se centra en el Control de Acceso Basado en Atributos (ABAC) dentro de la base de datos de vectores. Al-Rubaie y Chang (2019) destacan que para evitar la fuga de información, la arquitectura debe incluir un módulo de Anonimización y Enmascaramiento de PII (Información de Identificación Personal) antes de que los datos sean convertidos en embeddings y almacenados, garantizando que el modelo nunca "vea" datos sensibles reales.

La seguridad, la privacidad y la gobernanza de datos son inhibidores críticos en la adopción empresarial. El OWASP Top 10 para aplicaciones LLM identifica vulnerabilidades específicas de estos sistemas, como la inyección de prompts, el manejo inseguro de salidas, el envenenamiento de datos de entrenamiento y el robo de modelos. Estos riesgos, derivados de la naturaleza no determinística de la IA, exigen la implementación del principio de seguridad por diseño desde la concepción del sistema (Lasso Security 2025; OWASP Foundation 2025).

Igualmente, el cumplimiento normativo impacta la arquitectura mediante la exigencia de Explicabilidad (XAI). El EU AI Act (2024) y el marco del NIST AI RMF (2023) exigen que las arquitecturas de GenAI posean sistemas de Logging y Auditoría Semántica. Esto significa que la arquitectura de referencia debe registrar no solo quién accedió al sistema, sino qué instrucciones se dieron y qué fuentes de datos internos utilizó el modelo para generar una respuesta específica. En el contexto colombiano, la Ley 1581 de 2012 impone el principio de "Finalidad" y "Necesidad". Aplicado a una arquitectura de GenAI, esto obliga a implementar una capa de Gobernanza de Datos que valide que los datos corporativos usados para alimentar el contexto de la IA tengan el consentimiento explícito para ese uso. El Marco Ético para la IA en Colombia (Dirección de Desarrollo Tecnológico e Innovación 2024) añade que la arquitectura debe permitir la supervisión humana (Human-in-the-loop), integrando interfaces donde un experto pueda validar las respuestas antes de que impacten procesos críticos.

Los estándares internacionales proporcionan marcos para la gestión responsable de la IA. ISO/IEC 42001:2023 especifica requisitos para establecer un Sistema de Gestión de Inteligencia Artificial (AIMS), centrado en controles organizacionales y evaluación de riesgos. Complementariamente, el NIST AI Risk Management Framework (AI RMF) se estructura en

torno a las funciones Govern, Map, Measure y Manage, y enfatiza atributos de IA confiable como validez, explicabilidad, privacidad y equidad, introduciendo un perfil específico para los riesgos de la IA Generativa (ModelOp s. f.; NIST 2023).

3.6 Servicios en la Nube y Arquitecturas Agnósticas

Las arquitecturas agnósticas de proveedor emergen como respuesta al riesgo de vendor lock-in, ofreciendo flexibilidad y resiliencia al abstraer las aplicaciones de los servicios de nube subyacentes. Los principales proveedores como AWS Bedrock, Azure OpenAI y Google Vertex AI ofrecen servicios gestionados y blueprints de referencia que, si bien aceleran el desarrollo, deben ser analizados críticamente para mantener la independencia tecnológica (Bezruchko 2025; Confluent s. f.; Rios-Campos et al. 2023).

3.7 Observabilidad y Criterios de Evaluación

La observabilidad integral en sistemas de IA se fundamenta en métricas, logs, metadatos y linaje para comprender y predecir proactivamente el comportamiento del sistema. La IA Generativa, a su vez, puede transformar la observabilidad al permitir un análisis más eficiente de grandes volúmenes de datos para la detección de anomalías y la predicción de fallos (Kumar 2024; Suthar 2024).

En el contexto de la GenAI, la observabilidad trasciende las métricas clásicas (CPU, memoria, latencia) para enfocarse en la trazabilidad semántica. Según Shieh et al. (2024), la observabilidad en Large Language Models (LLMs) requiere capturar el flujo completo desde el prompt del usuario, pasando por la recuperación de documentos (RAG), hasta la respuesta final.

Esto incluye el monitoreo de:

- Consumo de Tokens: Fundamental para la gestión de costos operativos en arquitecturas basadas en APIs de terceros.
- Deriva del Modelo (Model Drift): Identificar cuando el rendimiento del modelo decae debido a cambios en los datos de entrada o actualizaciones del proveedor.

- Análisis de Latencia por Etapa: Diferenciar el tiempo que toma la búsqueda vectorial frente al tiempo de inferencia del LLM.

Finalmente, la evaluación de cualquier arquitectura de referencia debe basarse en criterios multidimensionales: técnicos (escalabilidad, modularidad, latencia), de negocio (costo total de propiedad, time-to-market) y de IA (precisión, robustez, gobernanza). Evaluar una IA Generativa es complejo debido a su naturaleza no determinista. La literatura actual (Es et al., 2024) propone el marco de evaluación de la Tríada del RAG, que mide:

- Veracidad (Faithfulness): Si la respuesta se deriva únicamente de los documentos recuperados (evitando alucinaciones).
- Relevancia del Contexto: Si los documentos recuperados por el motor de búsqueda son realmente útiles para responder la pregunta.
- Relevancia de la Respuesta: Si la salida del modelo responde directamente a la intención del usuario.

4. Metodología

El presente estudio se enmarca dentro de un enfoque aplicado con un diseño de Indagación por fases, orientado a la construcción y validación de un artefacto tecnológico: un modelo de referencia para el desarrollo de soluciones empresariales con Inteligencia Artificial generativa. La metodología combina elementos de investigación exploratoria, descriptiva y constructiva para abordar el problema desde su fundamentación teórica hasta su validación práctica, a partir de las siguientes fases:

Fase 1 Revisión y Fundamentación del estado del Arte

Fase 2 Análisis Comparativo y Síntesis de Hallazgos

Fase 3 Diseño y Modelado de la Arquitectura de Referencia

Fase 4: Propuesta de validación mediante Prueba de Concepto (PoC)

4.1 Fase 1: Revisión y Fundamentación del Estado del Arte

Esta fase inicial, de naturaleza exploratoria y descriptiva, tiene como objetivo construir una base de conocimiento sólida sobre el problema. La principal técnica de recolección de información será la revisión documental sistemática.

Se definirá un corpus de análisis a partir de la literatura académica y profesional sobre arquitecturas de software para sistemas de IA. La selección de las fuentes documentales se realizará de manera intencionada, centrándose en tres pilares de investigación:

- Arquitecturas y patrones de software relevantes (microservicios, arquitecturas orientadas a eventos) y las arquitecturas de referencia de los principales proveedores cloud (AWS, Azure, GCP).
- El ecosistema de herramientas y frameworks clave para el desarrollo, como LangChain, LlamaIndex y Semantic Kernel, así como metodologías MLOps.
- Casos de uso y buenas prácticas documentados en whitepapers, blogs técnicos y estudios de caso de implementaciones empresariales exitosas.

El procedimiento de análisis para esta fase será el análisis de contenido temático, que permitirá identificar, categorizar y sintetizar los desafíos, patrones recurrentes, herramientas

predominantes y buenas prácticas en áreas críticas como RAG, fine-tuning, seguridad y evaluación de modelos.

4.2 Fase 2: Análisis Comparativo y Síntesis de Hallazgos

En esta fase se transita de la descripción a la evaluación analítica. Utilizando la información recopilada en la fase anterior, se aplicará un análisis comparativo estructurado como principal procedimiento.

Se diseñarán matrices de alineación para evaluar sistemáticamente los enfoques, patrones y herramientas identificados. Este análisis se basará en un conjunto de criterios ponderados relevantes para el contexto empresarial, agrupados en tres dimensiones clave: técnicos (escalabilidad, modularidad), de negocio (TCO, time-to-market) y de IA (precisión, gobernanza). El resultado será una síntesis normativa en forma de informe que evidencie las fortalezas, debilidades y sinergias de las distintas alternativas, sentando una base empírica para el diseño de la propuesta.

4.3 Fase 3: Diseño y Modelado de la Arquitectura de Referencia

Esta es la fase constructiva del proyecto. El objetivo es traducir la síntesis de la fase anterior en un artefacto tangible: una propuesta de arquitectura de referencia. El procedimiento principal será el modelado arquitectónico, siguiendo principios de diseño explícitos.

Las actividades se centrarán en la definición de lineamientos de diseño (modularidad, agnosticismo tecnológico, seguridad por diseño, escalabilidad, observabilidad y gobernanza). Se desarrollarán los modelos conceptual y lógico de la arquitectura, incluyendo sus componentes, capas funcionales, flujos de datos y estrategias de seguridad. Para asegurar una documentación rigurosa y estandarizada, se empleará el marco arc42, generando diagramas y descripciones detalladas que justifiquen cada decisión de diseño.

4.4 Fase 4: Propuesta de validación mediante Prueba de Concepto (PoC)

La fase final es de carácter proyectivo y evaluativo, diseñada para establecer los lineamientos que permitan demostrar la viabilidad y efectividad de la arquitectura propuesta en futuras implementaciones. La técnica central propuesta será la definición detallada de una Prueba de Concepto (PoC).

Para ello, se seleccionará un caso de uso específico y relevante que permita poner a prueba los aspectos más críticos de la arquitectura diseñada. Se deberán establecer criterios de éxito y métricas cuantificables (funcionales y no funcionales) de manera previa a su ejecución técnica. El procedimiento de análisis proyectado consistirá en la planificación del monitoreo de la PoC, definiendo cómo se recopilarán los datos de rendimiento y comportamiento. Adicionalmente, se proponen listas de chequeo basadas en marcos como el OWASP LLM Top 10 para evaluar preventivamente la gobernanza y seguridad. Los resultados esperados de esta fase servirán para validar teóricamente la propuesta, identificar limitaciones y delinear futuras líneas de trabajo.

4.5 Resultados Esperados y Conclusiones del Trabajo

El resultado final de este plan será un documento de trabajo de grado que presente:

- Una síntesis y análisis del estado del arte sobre arquitecturas, herramientas y buenas prácticas para IA Generativa.
- Una propuesta de arquitectura de referencia.
- La evidencia y hallazgos de la PoC.

Se concluirá con una discusión sobre las limitaciones encontradas y las posibles líneas de trabajo futuro.

5. Análisis de resultados

En este capítulo se exponen los hallazgos derivados de la investigación, estructurados para dar respuesta a la necesidad de una arquitectura de referencia robusta. Se parte del análisis comparativo de la literatura para luego proceder con el diseño del blueprint propuesto.

5.1. Análisis de la Literatura Seleccionada y Criterios de Evaluación

Para garantizar la validez del "Blueprint" propuesto, se llevó a cabo un proceso de evaluación dual mediante dos matrices de análisis. Este enfoque permite distinguir entre la calidad de la estructura académica de los documentos y su aporte técnico específico al dominio de las arquitecturas de IA Generativa empresarial.

Con el fin de optimizar la visualización de los datos, los documentos evaluados se identifican en las tablas mediante un número de referencia (ID), cuyo detalle bibliográfico se encuentra al final de esta sección. Además, ambas matrices se han segmentado en dimensiones temáticas. Esta organización permite identificar con precisión si las debilidades de un estudio son metodológicas (forma) o de contenido (fondo), facilitando la selección de los componentes más robustos para la arquitectura de referencia.

A continuación, se detalla el significado de los criterios utilizados y se analiza la relación entre los resultados obtenidos en ambas dimensiones.

5.1.1. Estructura de la Matriz de Calidad Metodológica y Académica

Esta primera dimensión evalúa el rigor con el que se construyeron los artículos, basándose en estándares de Revisiones Sistemáticas de Literatura (SLR) (Pardal-Refoyo y Pardal-Peláez 2020). Los criterios fundamentales aplicados incluyen:

- Pregunta PICO y Objetivos: Evalúa si el paper define claramente el Problema, Intervención, Comparación y Outcome.
- Protocolo PRISMA: Verifica si el estudio sigue y reporta el diagrama de flujo y las directrices para revisiones sistemáticas.

- Reproducibilidad y Estrategia de Búsqueda: Mide si el autor especifica las bases de datos (>=5) y las cadenas de búsqueda, permitiendo que otros investigadores repliquen el estudio.
- Evaluación de Sesgo y Calidad (GRADE): Determina si se analizaron las limitaciones de los estudios primarios incluidos.

Tabla 1 *Dimensión de Definición (PICO y Objetivos) (Construcción propia).*

ID	Pregunta PICO	Objetivos de PICO	Concl. vs Obj.
[13], [15], [20]	Sí	Sí	Sí
[1]-[5], [7]-[12], [14], [21], [24]-[26]	No	No	Sí
[6], [16]-[19], [22], [23], [27]	No	No	No

Tabla 2 *Dimensión de Reproducibilidad y Búsqueda (Construcción propia).*

ID	Búsqueda Reproducible	Bases Datos (>=5)	Estrategia de Búsqueda
[15], [20]	Sí	Sí	Sí
[13]	Sí	Sí	No
[18]	No	Sí	No
[7]	No	No	Sí
Otros IDs	No	No	No

Tabla 3 *Dimensión de Protocolo y Selección (PRISMA) (Construcción propia).*

ID	Directrices PRISMA	Diagrama PRISMA	Motivos Exclusión
[15], [20]	Sí	Sí	Sí
[13], [18]	No	No	Sí
Otros IDs	No	No	No

Tabla 4 *Dimensión de Calidad, Sesgo y Puntuación Final (Construcción propia).*

ID	Variables Definidas	Riesgo de Sesgo	Calidad GRADE	Total
[15], [20]	Sí	Sí	No	11

[13]	No	No	No	7
[18]	No	No	No	3
[1]-[5], [7]-[12], [14], [21], [24]-[26]	No	No	No	2
[6], [16], [17], [19], [22], [23], [27]	No	No	No	1

5.1.2. Estructura de la Matriz de Relevancia Arquitectónica y Técnica

Para la evaluación técnica se definieron un total de 17 criterios específicos que permiten diseccionar la utilidad de cada fuente para la construcción de una arquitectura de referencia empresarial. No obstante, con el fin de facilitar la lectura y síntesis de resultados, se han seleccionado 5 criterios principales para la representación en las tablas comparativas:

- Propuesta de Arquitectura de IA: Determina si el documento presenta un diseño formal de sistema orientado a modelos de lenguaje o IA generativa.
- Soporte para RAG y Agentes: Evalúa la inclusión de patrones modernos de recuperación de información y autonomía agentil, pilares de este trabajo.
- Detalle de Componentes y Diagramas: Mide la profundidad técnica, verificando la presencia de esquemas de alto y bajo nivel que permitan la implementación.
- Atributos de Calidad (Escalabilidad, Seguridad y Monitoreo): Verifica que el diseño no sea solo conceptual, sino que considere la robustez necesaria en entornos de producción.
- Validación o Caso de Éxito: Confirma si la arquitectura ha sido probada en un escenario real, aportando evidencia de su efectividad.

Es importante destacar que, aunque los 12 criterios restantes (como el Stack Tecnológico, la Interoperabilidad, el Nivel de Madurez o la Gobernanza de Datos) no se visualizan individualmente en la tabla resumen, fueron evaluados rigurosamente y forman parte integral del puntaje total de calificación.

Tabla 5 *Dimensión de Diseño Arquitectónico y Estructural (Construcción propia).*

ID	Arq. Software	Propone Arq. IA	Detalla Componentes
[3], [4], [7], [16], [22], [23]	Sí	Sí	Sí

[2], [12], [21]	Sí	Sí	Parcial
Otros IDs	No	No	No

Tabla 6 *Dimensión de Patrones Modernos y Soporte Técnico (Construcción propia).*

ID	Soporte (RAG/Agentes)	Diagrama Alto Nivel	Diagrama Detallado
[4], [22], [23]	Sí	Sí	Sí
[3], [21]	Sí	Sí	No
[16]	No	Sí	Sí

Tabla 7 *Dimensión de Atributos de Calidad Empresarial (Construcción propia).*

ID	Escalabilidad	Seguridad/Privacidad	Monitoreo/Obs.
[2], [22], [23]	Sí	Sí	Sí
[3], [4], [13], [21]	Parcial	Sí	No
Otros IDs	No	Parcial	No

Tabla 8 *Dimensión de Madurez, Implementación, Gobernanza y Puntuación Final (Construcción propia).*

ID	Validación/Caso	Nivel de Madurez	Stack Tecnológico	Gob. Datos	Total
[22], [23]	Sí	Sí	Sí	Sí/Parcial	15 - 15.5
[18]	Sí	Parcial	Sí	No	13
[2], [3], [4], [21]	Sí	Parcial	Sí	Parcial	12 - 12.5
[1], [24]	Sí	Parcial	Sí	Parcial	10
Otros IDs	No/Parcial	Parcial	Parcial	No	< 10

5.1.3. Contraste de Resultados: El Dilema entre Rigor y Relevancia Técnica

Un hallazgo fundamental de este análisis es la frecuente correlación inversa entre los puntajes de ambas matrices para ciertos documentos. Varios artículos presentan bajas calificaciones en el rigor académico pero puntuaciones sobresalientes en la dimensión técnica.

Esta disparidad se explica por la naturaleza emergente y "poco académica" del tema central. Al ser la IA Generativa Empresarial un campo impulsado principalmente por la industria y la ingeniería aplicada, muchos de los mejores aportes técnicos provienen de White Papers o reportes de arquitectura que priorizan la implementación sobre la metodología.

Fuentes con alta técnica y baja academia: Documentos como el ID [23] - RAGOps (X. Xu et al. 2025) o el ID [22] - LlamaFirewall (Chennabasappa et al. 2025), obtienen puntajes bajos en la matriz académica porque no son revisiones sistemáticas, pero en la matriz técnica son fundamentales por sus detalles de seguridad y gestión.

Fuentes con alta academia y baja técnica: El trabajo ID [20] (Esparza, Blanco, y Sánchez 2025) destaca por su excelente estructura académica, siguiendo rigurosamente las normas de publicación, pero su aporte técnico específico es menor.

5.1.4. Listado de Referencias Evaluadas

A continuación, se relacionan los identificadores utilizados en las tablas con sus respectivos títulos:

- [1] A Generative Artificial Intelligence Using Multilingual Large Language Models for ChatGPT Applications (Tuan et al. 2024).
- [2] A Network Architecture for Scalable End-to-End Management of Reusable AI-Based Applications in 6G Networks (Sai Charan Madugula 2025).
- [3] A Study on the Implementation of Generative AI Services Using an Enterprise Data-Based LLM Application Architecture (Jeong 2023).
- [4] All You Need is an AI Platform: A Proposal for a Complete Reference Architecture (Weigell, Stieler, y Bauer 2025).
- [5] Architectures for Industrial AIoT Applications (Villar et al. 2024).
- [6] Cloud Native System for LLM Inference Serving (M. Xu et al. 2025).
- [7] Cloud Platforms for Developing Generative AI Solutions: A Scoping Review of Tools and Services (Patel et al. 2024).

- [8] Cognitive Architecture Toward Common Ground Sharing Among Humans and Generative AIs: Trial on Model-Model Interactions in Tangram Naming Task (Morita et al. 2024).
- [9] Collaborative Agentic AI Needs Interoperability Across Ecosystems (Sharma et al. 2025).
- [10] Comprehensive Review of Artificial General Intelligence AGI, Agentic AI and GenAI: Current Trends and Future Directions (Joshi 2025)
- [11] Designing ethical and transparent AI for regulated enterprise environments (Goutham Yenuganti 2025)
- [12] Enterprise API & Platform Strategy in the era of Agentic AI (Satav 2025)
- [13] Enterprise Architecture as a Dynamic Capability for Scalable and Sustainable Generative AI Adoption: Bridging Innovation and Governance in Large Organisations (Ettinger 2025)
- [14] FlashRAG: A Modular Toolkit for Efficient Retrieval-Augmented Generation Research (Jin et al. 2025)
- [15] Generative AI for Software Architecture. Applications, Trends, Challenges, and Future Directions (Esposito et al. 2025)
- [16] Generative Artificial Intelligence (Rios-Campos et al. 2023)
- [17] Generative artificial intelligence in innovation management: A preview of future research developments (Mariani y Dwivedi 2024)
- [18] Generative Artificial Intelligence: A Systematic Review and Applications (Sengar et al. 2024)
- [19] Guía de aplicación de arquitectura empresarial orientada a la implementación de MOOC en la Universidad Católica de Cuenca. Primeros Pasos (Campoverde-Peñafiel y Cordero-Guzmán 2019)
- [20] Generative design as a means of effective communication in multidisciplinary teams: a systematic review (Esparza et al. 2025)
- [21] Identifying and Building Generative AI Use Cases Within Enterprise Software Products (Bhalla 2023)
- [22] LlamaFirewall: An open source guardrail system for building secure AI agents (Chennabasappa et al. 2025)

- [23] RAGOps: Operating and Managing Retrieval-Augmented Generation Pipelines (X. Xu et al. 2025)
- [24] Role of Databases in GenAI Applications (Bhupathi 2025)
- [25] Securing generative AI workloads: A framework for enterprise implementation (Kalyan Pavan Kumar Madicharla 2025)
- [26] Security and Privacy Challenges of Large Language Models: A Survey (Das et al. 2024)
- [27] Towards a Middleware for Large Language Models (Guran et al. 2024)

5.2 Fundamentación Metodológica de la Arquitectura: Sinergia TOGAF y arc42

El desarrollo de la arquitectura de referencia propuesta trasciende la mera especificación técnica al adoptar un enfoque híbrido que articula el rigor de la gestión empresarial con la precisión del diseño de sistemas. Esta sección detalla cómo la convergencia de **TOGAF** (The Open Group Architecture Framework) y el estándar **arc42** permite abordar la complejidad inherente a la IA Generativa desde una perspectiva holística y multidimensional.

5.2.1 Alineación Estratégica

La integración de estos marcos responde a la necesidad de mitigar la brecha entre la innovación disruptiva y la sostenibilidad operativa. La metodología se estructura bajo la siguiente lógica de complementariedad:

- TOGAF como Marco de Gobernanza: Mediante la aplicación del Método de Desarrollo de Arquitectura (ADM), se garantiza que el artefacto técnico esté alineado con la visión del negocio y los requisitos de los diversos *stakeholders*. Específicamente, las fases A, B y C de TOGAF proporcionan el contexto estratégico para definir las capacidades de negocio y la gestión de riesgos (Campoverde-Peñafiel y Cordero-Guzmán 2019).
- arc42 como Estándar de Documentación: Mientras TOGAF define el "qué" y el "para qué", arc42 proporciona la estructura necesaria para comunicar el "cómo". Este estándar facilita la descripción detallada de la arquitectura mediante vistas lógicas, de contenedores y de componentes, asegurando que las decisiones de diseño sean comprensibles para los equipos de ingeniería (Arc42: Cómo Documentar Arquitecturas Cloud Complejas para Garantizar el Éxito

2024).

5.2.2 Correspondencia de Artefactos y Trazabilidad

Para garantizar la integridad del diseño, se ha establecido una matriz de trazabilidad que vincula los dominios de arquitectura empresarial con los niveles de detalle técnico de arc42.

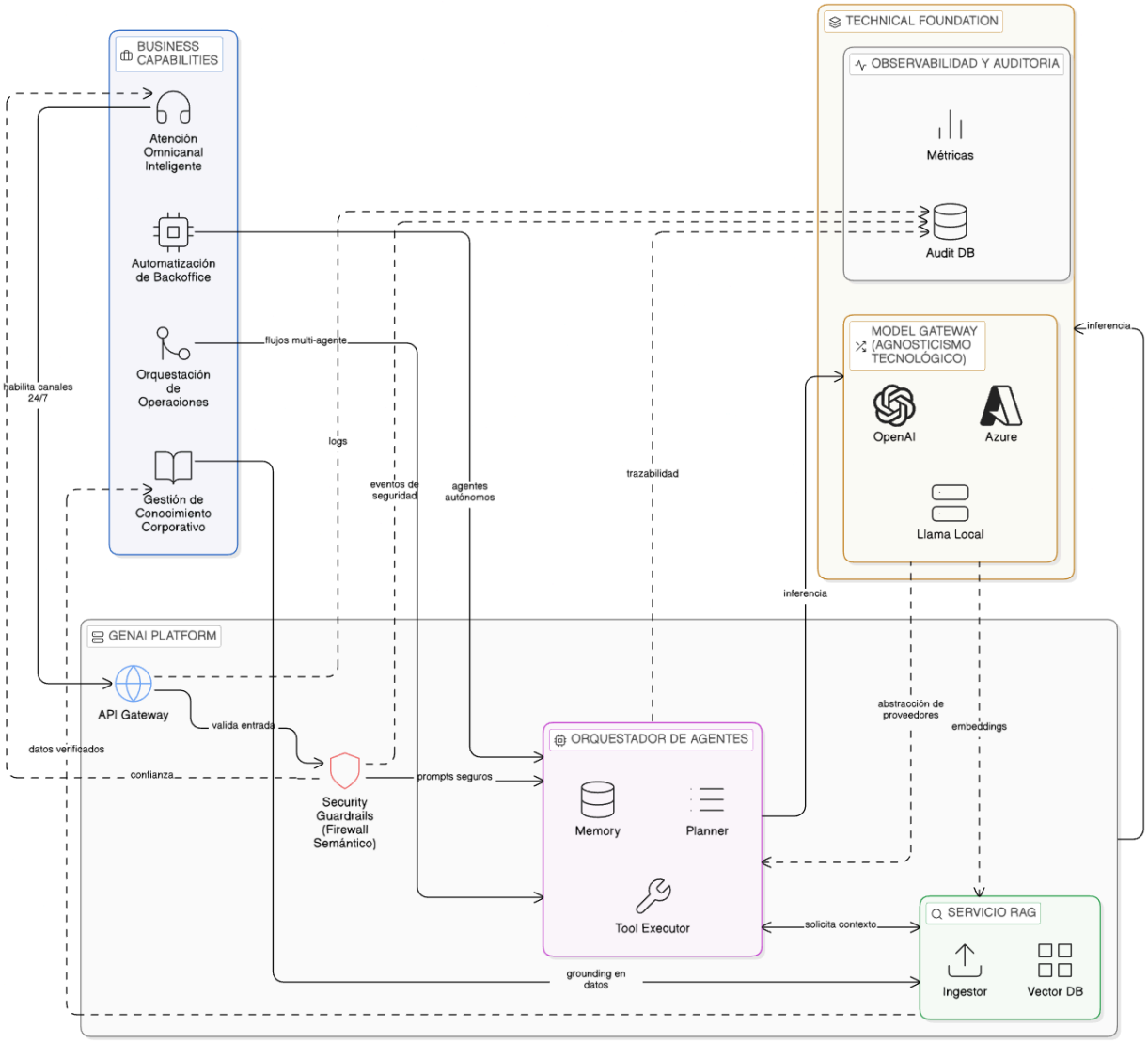
- Visión y Requisitos (Fase A de TOGAF): Se ven reflejados en las secciones de metas, objetivos de calidad y contexto de arc42.
- Arquitectura de Sistemas de Información (Fase C de TOGAF): Se documenta a través de las vistas lógicas y de contenedores, definiendo el flujo de datos y la interoperabilidad del sistema.
- Gobernanza y Decisiones: La formalización de los Architecture Decision Records (ADRs) en arc42 actúa como la evidencia técnica de la fase de Gobernanza de TOGAF, justificando elecciones críticas como el agnosticismo tecnológico y la seguridad por diseño.

Esta sinergia metodológica permite que la arquitectura propuesta no solo resuelva desafíos técnicos inmediatos —como el vendor lock-in o la inyección de prompts— sino que se consolide como un activo estratégico capaz de evolucionar ante la volatilidad tecnológica del ecosistema de la IA

5.3 Visión de Arquitectura

La plataforma está diseñada para servir como un middleware inteligente que abstrae la complejidad de la infraestructura de IA. Bajo la perspectiva de la Fase A de TOGAF, esta visión busca transformar la innovación disruptiva en una capacidad operativa técnica y económicamente sostenible para las organizaciones. El enfoque principal radica en resolver los silos tecnológicos, mitigar riesgos de seguridad como la inyección de prompts y eliminar la dependencia de proveedores (vendor lock-in).

Figura 1 *Blueprint de la Arquitectura de Referencia Empresarial (Construcción propia)*



5.3.1 Organización Funcional por Capas

El diseño visualizado en el blueprint se organiza en tres niveles de responsabilidad que permiten desacoplar los objetivos de negocio de la implementación técnica:

- Capa de Capacidades de Negocio: Define los servicios de alto valor habilitados por la arquitectura, tales como la Atención Omnicanal Inteligente mediante chatbots con anclaje de datos , la Automatización de Backoffice con agentes autónomos que ejecutan acciones en entornos digitales , y la Gestión de Conocimiento Corporativo para generar respuestas verificables.

- Capa de Plataforma GenAI: Constituye el motor de razonamiento y seguridad. Integra componentes críticos como los Security Guardrails para la protección de datos sensibles y el Servicio RAG para asegurar que las respuestas estén ancladas en datos reales de la organización.
- Capa de Fundamentos Técnicos: Proporciona la infraestructura transversal necesaria para la operación empresarial. Incluye la Observabilidad y Auditoría para comprender y predecir el comportamiento del sistema , y el Model Gateway, que garantiza el agnosticismo tecnológico al abstraer a los proveedores de modelos.

5.3.2 Proposición de Valor y KPIs por Stakeholder

La arquitectura maximiza el retorno de inversión mediante indicadores de desempeño (KPIs) de referencia que permiten monitorear el éxito de la Fase A:

Tabla 9 *KPI de interés para stakeholders (Construcción propia).*

Stakeholder	Proposición de Valor	KPIs de Referencia
Arquitectos	Estandarización y reducción de deuda técnica.	• % de componentes reutilizados por proyecto. • Reducción en tiempo de integración de modelos.
CISO / Seguridad	Control perimetral semántico y cumplimiento.	• % de ataques de inyección detectados/bloqueados. • Cobertura de políticas de seguridad aplicadas.
Operaciones (DevOps)	Automatización del ciclo de vida y observabilidad.	• Disponibilidad (uptime) de servicios de inferencia. • Costo promedio por consulta (Tokens/USD).

Negocio / Product Owners	Agilidad en el Time-to-market y eficiencia.	• Tiempo de despliegue de nuevos casos de uso. • Tasa de precisión percibida por el usuario.

5.3.2 Capacidades de Negocio Habilitadas

Bajo la Arquitectura de Negocio (Fase B), el blueprint habilita capacidades que superan el paradigma pasivo de interacción, transformando los casos de uso técnicos en valor organizacional:

- Atención Omnicanal Inteligente: Resolución de consultas mediante asistentes conversacionales y chatbots avanzados con anclaje de datos (RAG).
- Automatización de Procesos de Backoffice: Capacidad de agentes autónomos (Agentic AI) para percibir y ejecutar acciones complejas en entornos digitales (ej. operar un ERP).
- Gestión de Conocimiento Corporativo: Generación de respuestas precisas, contextualizadas y verificables (Jeong 2023) mediante el procesamiento de bases de conocimiento.
- Orquestación de Operaciones Dinámicas: Automatización de flujos de trabajo de negocio mediante la colaboración multi-agente y coordinación de tareas (Sharma et al. 2025).

5.4 Stakeholders y Preocupaciones

El diseño de la arquitectura responde a las necesidades y expectativas de diversos actores clave dentro de la organización. La identificación de estos interesados permite alinear las decisiones técnicas con los objetivos estratégicos, tal como se sugiere en la gestión de participación de TOGAF (Campoverde-Peñafiel y Cordero-Guzmán 2019).

Tabla 10 *Rol e interés de los stakeholders (Construcción propia).*

Rol	Puntaje

Arquitectos de Soluciones	Definición de patrones reutilizables, selección de stack tecnológico y cumplimiento de estándares empresariales.
Equipos de Seguridad (CISO)	Protección de datos sensibles (DLP), cumplimiento normativo (GDPR, AI Act) y mitigación de vulnerabilidades OWASP LLM (OWASP Foundation 2025).
Operaciones (DevOps/MLOps)	Observabilidad de "caja negra" (LLMs), CI/CD para prompts/modelos y gestión de costes de tokens (Patel et al. 2024).
Negocio / Product Owners	Reducción del Time-to-market, ROI de las soluciones y alineación con objetivos estratégicos.

5.5 Objetivos de Calidad y Requisitos

Los impulsores arquitectónicos se derivan de la revisión del estado del arte y los desafíos de adopción empresarial identificados en la fundamentación teórica:

- **Interoperabilidad ("Web of Agents")**: La capacidad de los agentes para comunicarse y colaborar a través de ecosistemas fragmentados, evitando silos de información mediante protocolos estándar (Sharma et al. 2025).
- **Seguridad y Confianza (Defensa en Profundidad):** Implementación de controles estrictos contra riesgos específicos de LLMs como prompt injection, alucinaciones y fuga de datos, alineado con NIST AI RMF (NIST 2023; OWASP Foundation 2025).
- **Agnosticismo Tecnológico:** Prevención del Vendor Lock-in mediante la abstracción de proveedores de modelos (OpenAI, AWS, Azure, Local), permitiendo la intercambiabilidad sin refactorización (Patel et al. 2024).
- **Gobernanza de Datos:** Trazabilidad total de las decisiones de la IA y anclaje de respuestas (Grounding) en datos corporativos para minimizar la invención de hechos (Jeong 2023).
- **Escalabilidad:** Gestión eficiente de recursos de inferencia (GPUs) y bases de datos vectoriales ante picos de demanda y grandes volúmenes de documentos (Patel et al. 2024).

5.5.1 Requisitos No Funcionales (RNF)

Siguiendo el estándar ISO/IEC 25010, se establecen los siguientes requisitos técnicos:

Seguridad y Privacidad

- RNF-SEC-01 (Protección contra Prompt Injection): El sistema debe implementar mecanismos de validación y sanitización de entradas para prevenir ataques de inyección directos e indirectos (OWASP Foundation 2025).
- RNF-SEC-02 (Prevención de Fuga de Datos): Se deben aplicar filtros de salida y técnicas de enmascaramiento (PII masking) para evitar que los modelos revelen información sensible .
- RNF-SEC-03 (Control de Acceso Granular - RBAC): El acceso a documentos en flujos RAG debe respetar los permisos originales del usuario en la fuente (ej. SharePoint ACLs) (Das et al. 2024).
- RNF-SEC-04 (Sandboxing): Los agentes con capacidad de ejecutar código deben operar en entornos aislados (contenedores efímeros) para prevenir movimientos laterales (Sharma et al. 2025).

Fiabilidad

- RNF-REL-01 (Grounding): El sistema debe priorizar arquitecturas RAG para anclar respuestas en datos verificados en lugar de conocimiento paramétrico del modelo.
- RNF-REL-02 (Citación de Fuentes): Toda respuesta basada en datos corporativos debe incluir referencias explícitas a los documentos fuente recuperados (Jeong 2023).

Interoperabilidad

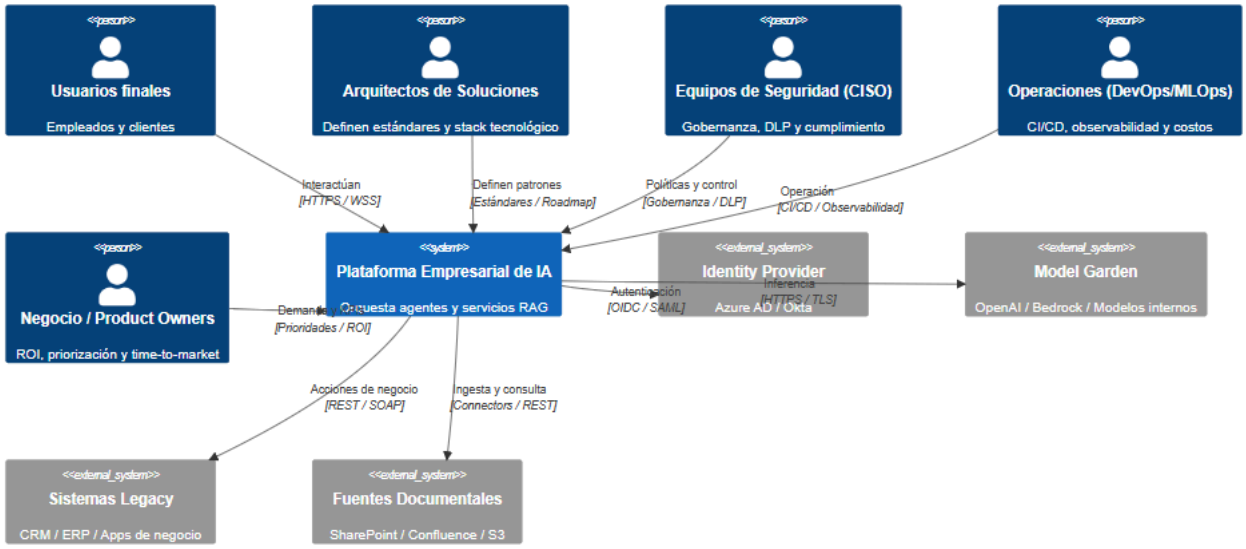
- RNF-INT-01 (API-First): Todas las capacidades (chat, ingestión, gestión) deben exponerse vía APIs REST/gRPC documentadas (OpenAPI) para integración con sistemas legacy.
- RNF-INT-02 (Estandarización de Agentes): Uso de formatos de mensaje estándar agnósticos al proveedor para permitir la colaboración en ecosistemas heterogéneos (Sharma et al. 2025).

5.6 Vista de Contexto

Esta vista describe las fronteras del sistema y sus interacciones externas, alineándose con los diagramas de comunicación de la Fase C de TOGAF. La plataforma orquesta la comunicación

entre usuarios finales, proveedores de identidad, catálogos de modelos (Model Gardens) y sistemas legados de la organización.

Figura 2 *Diagrama de Contexto (C4) (Construcción propia)*



5.7.1 Interacciones Clave

La Plataforma Empresarial de IA orquesta las siguientes interacciones mediante protocolos seguros:

- Usuarios -> Plataforma (HTTPS/WSS): Empleados y clientes acceden a asistentes. Se utiliza WebSockets (WSS) para streaming de respuestas token a token, mejorando la percepción de latencia.
- Plataforma -> IdP (OIDC/SAML): Delegación de autenticación a proveedores de identidad corporativos (Azure AD, Okta) para gestión centralizada de usuarios.
- Plataforma -> Model Garden (HTTPS/TLS): Envío de prompts y recepción de inferencias desde proveedores externos (OpenAI, Bedrock) o internos.
- Plataforma -> Sistemas Legacy (REST/SOAP): Ejecución de acciones de negocio (ej. consultar CRM) mediante conectores seguros gestionados por los agentes.
- Plataforma -> Fuentes Documentales (Connectors): Ingestión continua y sincronización de conocimientos desde repositorios como SharePoint, Confluence y S3.

Tabla 11 *Registros de Decisión de Arquitectura (Construcción propia).*

Interfaz	Protocolo	Datos Sensibles Transmitidos	Rol Arquitectónico
Identity Provider (IdP)	OIDC / SAML	Tokens de acceso, roles, IDs de usuario.	Autenticación y control RBAC.
Model Providers	HTTPS / TLS 1.3	Prompts, Embeddings, parámetros de inferencia.	Inferencia de modelos fundacionales.
Repositorios Docs	REST / Graph API	PDFs corporativos, metadatos, permisos.	Fuente de datos para RAG.
Sistemas Legacy	REST / SOAP	Datos de negocio (ERP), PII, transacciones.	Ejecución de acciones (Tools).

5.7 Vista de Contenedores y Datos

Para garantizar la modularidad y el cumplimiento de la Arquitectura de Sistemas de Información, se define una estructura basada en componentes desacoplados. Esto permite garantizar alta disponibilidad y escalabilidad elástica ante picos de demanda mediante el despliegue de microservicios independientes.

Figura 3 *Diagrama de Contenedores (C4) (Construcción propia).*

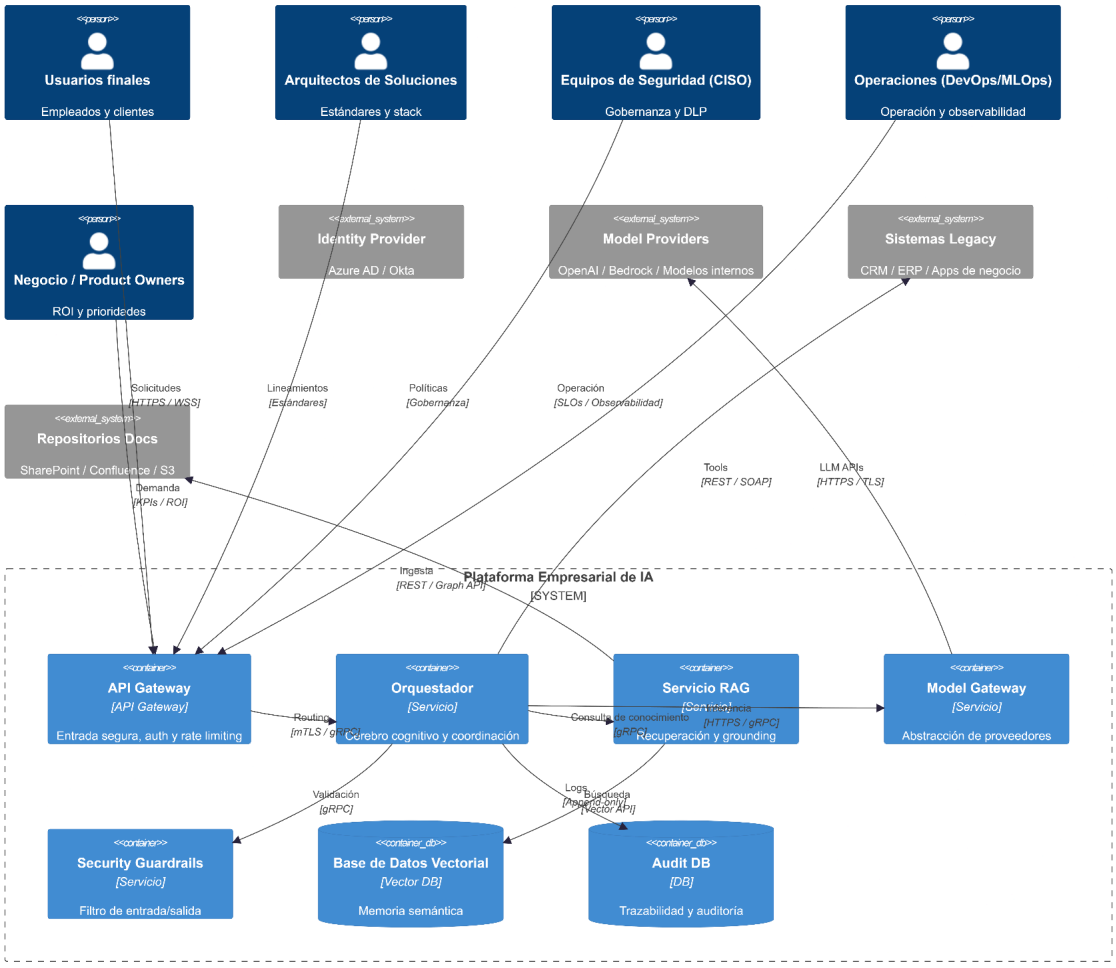


Tabla 12 *Descripción de Contenedores (Construcción propia).*

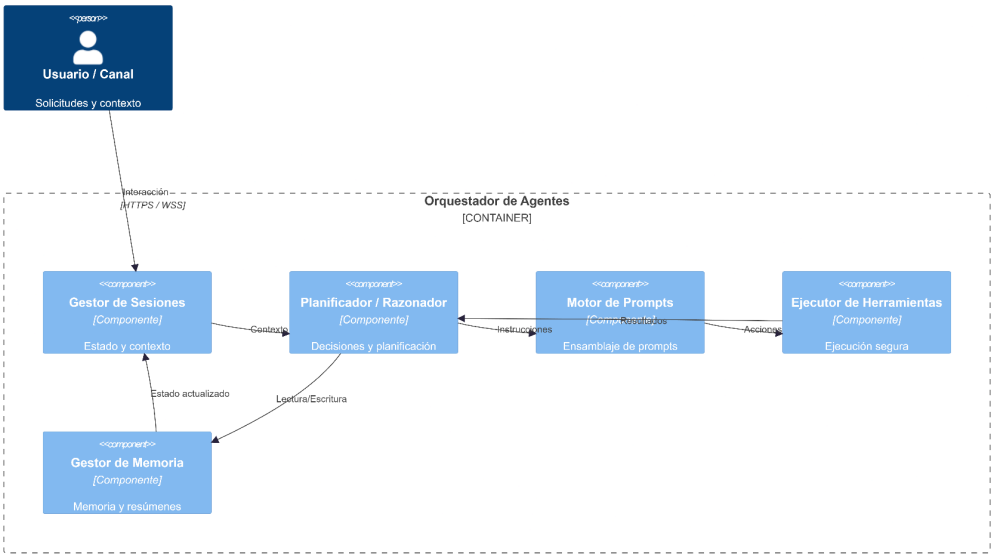
Contenedor	Función Principal	Responsabilidad
API Gateway	Punto de entrada único	Gestiona la seguridad perimetral, autenticación y el control de tráfico (rate limiting).
Orquestador	Núcleo cognitivo	Actúa como el "cerebro" que mantiene el hilo de la conversación y decide qué herramientas usar (Sharma et al. 2025).
Servicio RAG	Gestor de conocimiento	Conecta la IA con los datos privados de la empresa, encargándose de buscar y recuperar información relevante (Jeong 2023).

Model Gateway	Abstracción de modelos	Intercambiabilidad de modelos y resiliencia ante fallos (Esposito et al. 2025).
Security Guardrails	Firewall semántico	Protege el sistema filtrando entradas maliciosas (jailbreaks) y evitando que la IA responda con datos sensibles (Das et al. 2024; OWASP Foundation 2025).
Base de Datos Vectorial	Memoria semántica	Almacena y recupera eficientemente grandes volúmenes de documentos mediante similitud matemática (Jeong 2023).
Audit DB	Centro de trazabilidad	Registra de forma inmutable todas las interacciones para fines de auditoría, cumplimiento y mejora del sistema (Patel et al. 2024).

5.7.1 Componentes del Orquestador de Agentes

El núcleo cognitivo del sistema se integra por módulos funcionales especializados que permiten el razonamiento lógico y la ejecución de tareas bajo un esquema de seguridad robusto. Este diseño permite que el orquestador actúe como un cerebro centralizado que desacopla la lógica de negocio del proveedor de IA utilizado.

Figura 4 *Orquestador de Agentes (Construcción propia).*

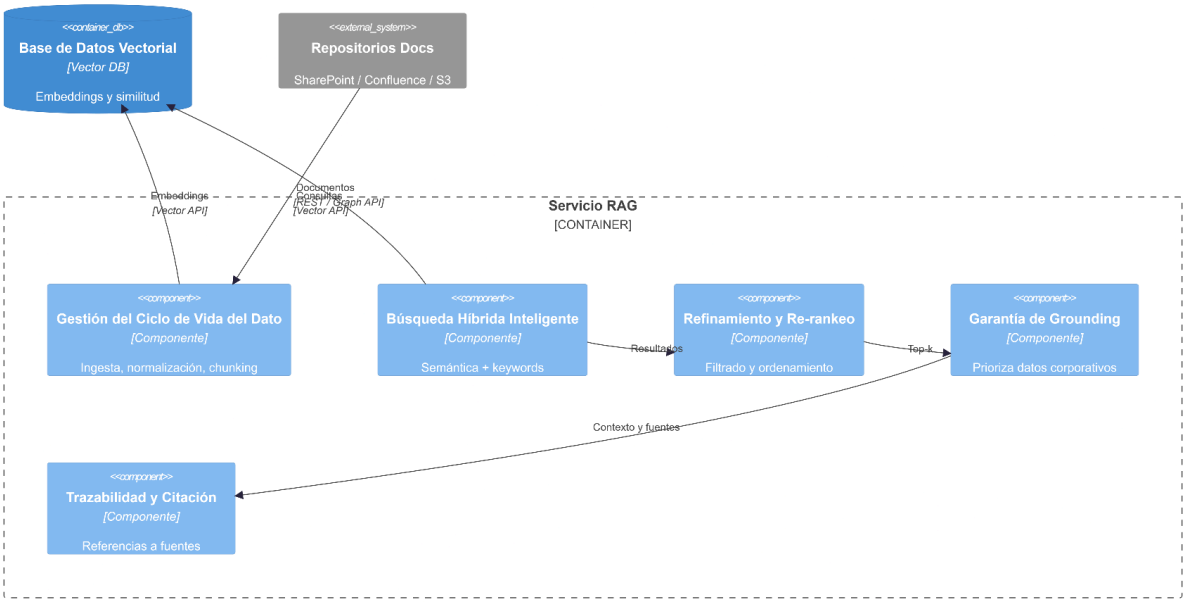


- Gestor de Sesiones (Session Manager): Administra el estado efímero de las conversaciones y el contexto específico de cada usuario. Su diseño permite el almacenamiento persistente de la sesión, facilitando el escalado horizontal del orquestador para soportar cargas variables de trabajo.
- Planificador / Razonador (Planner / Reasoner): Actúa como el motor principal para la toma de decisiones del agente. Implementa patrones de razonamiento lógico, como la descomposición de problemas en pasos ejecutables, permitiendo que el sistema determine de manera autónoma qué acciones realizar para cumplir un objetivo.
- Motor de Prompts (Prompt Engine): Realiza el ensamblaje dinámico y la orquestación del mensaje final que se transfiere al modelo de lenguaje. Este componente es el responsable de inyectar de forma coherente las reglas del sistema, ejemplos de referencia, la información recuperada del conocimiento corporativo y el historial de la interacción.
- Ejecutor de Herramientas (Tool Executor): Provee una capa de abstracción para la ejecución de funciones y llamadas a sistemas externos. Funciona como un mecanismo de defensa interna, validando rigurosamente los parámetros generados por el modelo antes de su ejecución en un entorno controlado y aislado.
- Gestor de Memoria (Memory Manager): Administra la persistencia de la información a largo plazo y la recuperación de resúmenes de interacciones pasadas. Su función crítica es la optimización de la ventana de contexto, asegurando que el modelo trabaje siempre con la información más relevante sin exceder sus límites técnicos.

5.7.2 Componentes del Servicio RAG

Este componente actúa como el puente entre los modelos de lenguaje y el conocimiento corporativo privado. Su función no es solo "buscar archivos", sino garantizar que la IA genere respuestas basadas en hechos verificables (Jeong 2023).

Figura 5 *Servicio RAG (Construcción propia).*

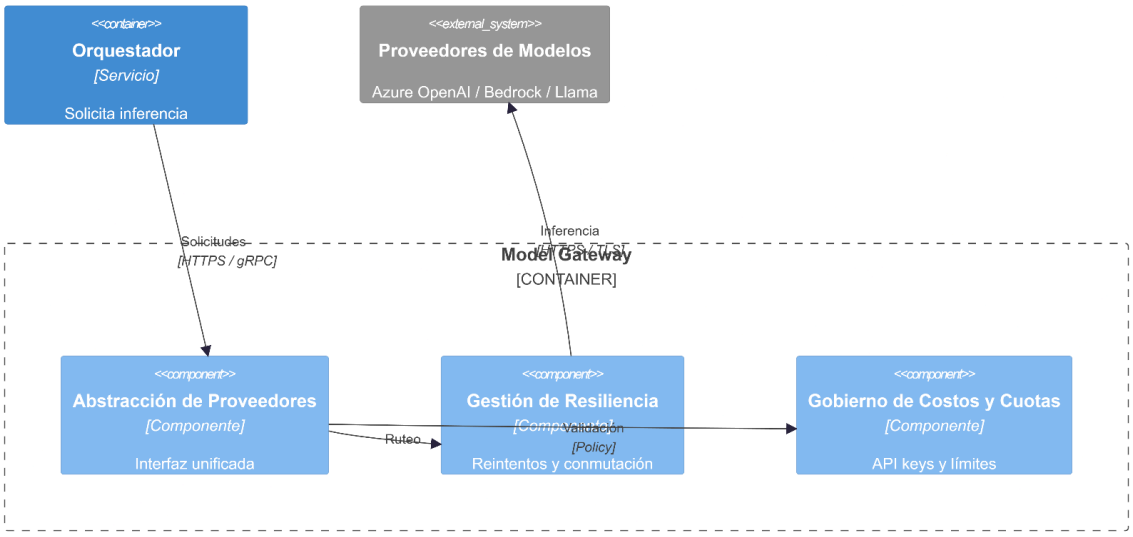


- Gestión del Ciclo de Vida del Dato: Automatiza la ingesta, normalización y fragmentación semántica de documentos en diversos formatos (PDF, HTML, DOCX) para que sean procesables por la IA.
- Búsqueda Híbrida Inteligente: Combina la potencia de la búsqueda por similitud (basada en el significado de la consulta) con la precisión de la búsqueda por palabras clave para encontrar la información más relevante.
- Refinamiento y Re-rankeo: Implementa una capa de filtrado avanzado que analiza los mejores resultados encontrados y los reordena antes de entregarlos al orquestador, eliminando ruidos o datos irrelevantes.
- Garantía de Grounding (Anclaje): Fuerza al sistema a priorizar la información recuperada de los repositorios internos sobre el conocimiento general del modelo, minimizando el riesgo de "alucinaciones".
- Trazabilidad y Citación: Asegura que cada respuesta incluya referencias explícitas a los documentos fuente originales, permitiendo al usuario final verificar la fuente de la información

5.7.3 Componentes del Model Gateway

Este componente actúa como la capa de abstracción crítica para evitar el vendor lock-in (Guran et al. 2024).

Figura 6 *Model Gateway (Construcción propia)*

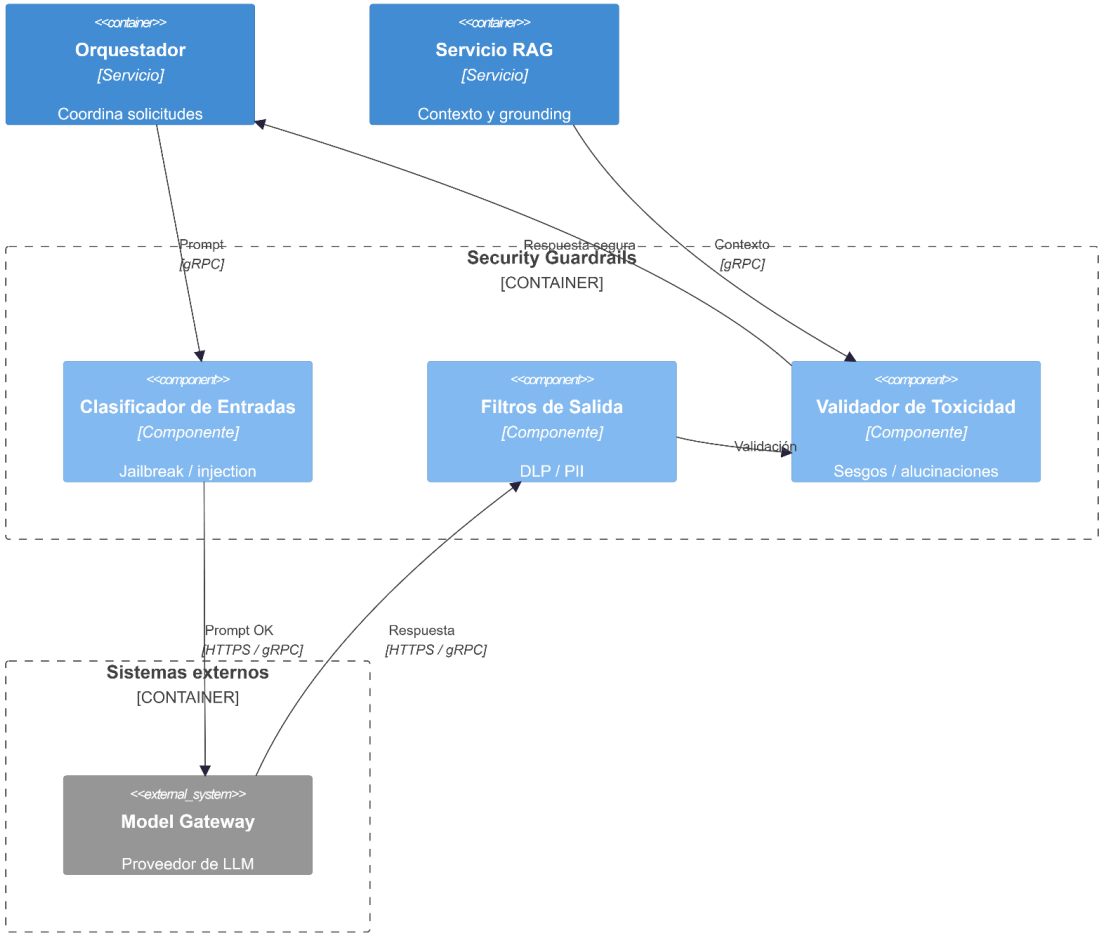


- Abstracción de Proveedores (Proxy): Utiliza una interfaz unificada (basada en el estándar de OpenAI o protocolos similares) para redirigir peticiones a múltiples proveedores como Azure OpenAI, AWS Bedrock o instancias locales de Llama 3.
- Gestión de Resiliencia (Failover): Implementa lógica de reintentos y redirección automática; si el proveedor principal falla o alcanza límites de cuota, el tráfico se conmuta a un modelo secundario sin afectar al usuario.
- Gobierno de Costos y Cuotas: Centraliza el manejo de API Keys y permite establecer límites de consumo por unidad de negocio o proyecto, evitando costos imprevistos.

5.7.4 Componentes de Security Guardrails

Definido como el "Firewall Semántico" del sistema, es el encargado de la seguridad por diseño (Chennabasappa et al. 2025).

Figura 7 *Security Guardrails (Construcción propia).*

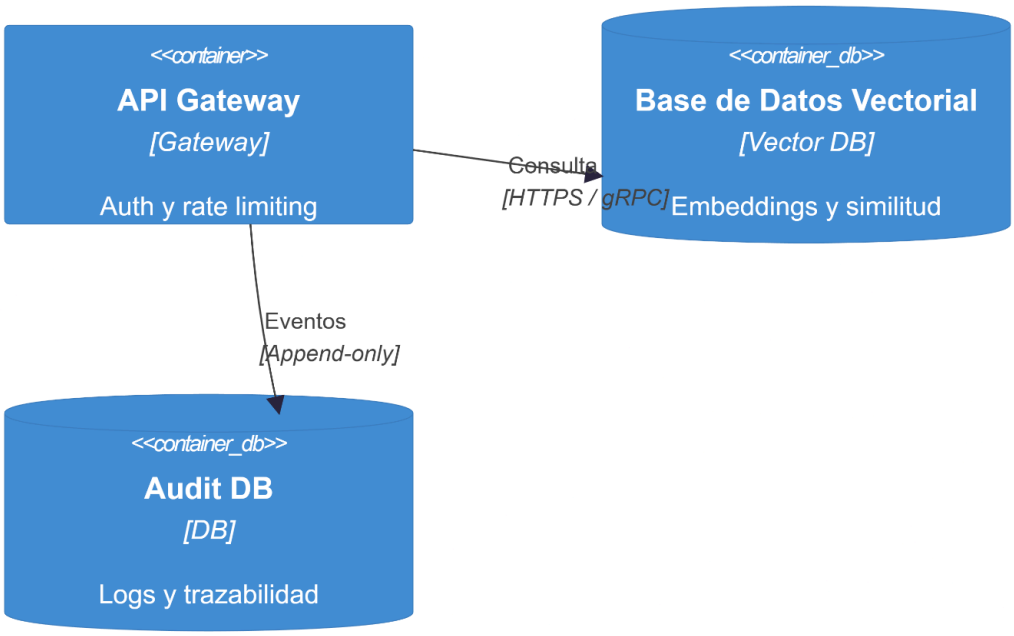


- Clasificador de Entradas (Jailbreak Detection): Evalúa si el prompt del usuario contiene intentos de inyección, manipulación ética o ingeniería social antes de que llegue al orquestador.
- Filtros de Salida (DLP & PII Masking): Escanea la respuesta generada por el LLM para detectar y enmascarar información sensible (nombres, documentos, datos financieros) que no deba ser expuesta.
- Validación de Toxicidad y Alucinaciones: Compara la respuesta con el contexto recuperado por el servicio RAG para asegurar que el contenido sea factual y libre de sesgos ofensivos.

5.7.5 Componentes de Infraestructura y Datos

Estos componentes aseguran la trazabilidad y la escalabilidad del sistema (Bhupathi 2025).

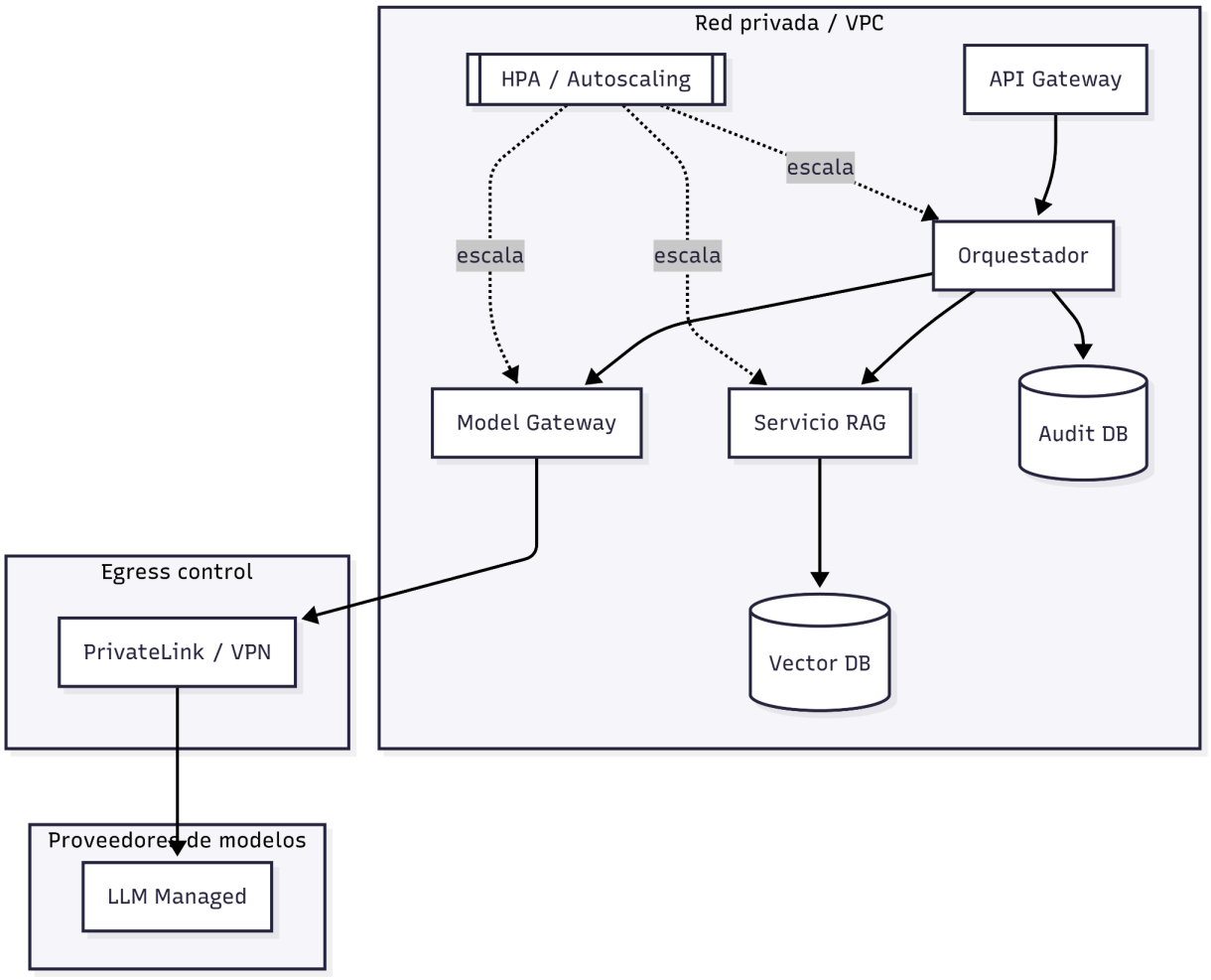
Figura 8 *Infraestructura y Datos (Construcción propia).*



- API Gateway: Gestiona la autenticación centralizada (OIDC/SAML) y el rate limiting para proteger los microservicios internos de ataques de denegación de servicio (DoS).
- Base de Datos Vectorial: Almacena las representaciones matemáticas (embeddings) de los documentos corporativos, permitiendo búsquedas de similitud semántica de alta velocidad.
- Audit DB: Repositorio de logs inmutables que registra cada interacción, incluyendo el prompt original, la respuesta del modelo, los documentos recuperados y los tokens consumidos para fines de auditoría y cumplimiento.

5.8 Vista de Despliegue y Patrones

Figura 9 *Vista de Despliegue y Patrones (Construcción propia)*



En concordancia con la Fase D (Arquitectura Tecnológica), el despliegue se define mediante patrones de capacidad que aseguran la resiliencia del sistema:

5.8.1 Escalado elástico de inferencia

Este patrón se representa con los componentes de inferencia/orquestación como Deployments en Kubernetes detrás de un Service, con HPA ajustando réplicas según métricas.

La necesidad en GenAI es que la carga de inferencia fluctúa y las arquitecturas cloud-native con autoscaling en Kubernetes pueden “adaptarse dinámicamente a fluctuaciones de workload” para optimizar desempeño de serving (throughput/latencia) en escenarios de alta demanda (M. Xu et al. 2025).

5.8.2 Conectividad privada (Egress control)

En despliegue, este patrón se muestra como subredes privadas sin salida directa a Internet y un único punto de egreso controlado hacia el proveedor (PrivateLink/VPN).

AWS documenta que PrivateLink establece conectividad entre recursos en subredes privadas y servicios fuera de tu VPC mediante "VPC endpoint services", evitando enrutar tráfico por la Internet pública.

Esto es especialmente necesario en GenAI porque el tráfico de egreso incluye prompts y contexto recuperado (potencialmente sensible), así que el diagrama debe evidenciar el "camino privado" y el control de rutas/egreso (AWS PrivateLink concepts - Amazon Virtual Private Cloud s. f.).

5.8.3 Persistencia políglota

Este patrón se refleja como tres almacenes separados (idealmente en servicios administrados o StatefulSets dedicados): Redis para estado de conversación/sesión, SQL para datos transaccionales, y un vector store para memoria semántica (embeddings).

RAGOps plantea que los sistemas RAG requieren una capa de observabilidad que registre queries, respuestas y entradas/salidas de cada componente, lo cual se implementa más limpiamente cuando separas stores por responsabilidad (sesión, transacción, semántica) y agregas un store de auditoría/observabilidad (X. Xu et al. 2025)

5.9 Decisiones Arquitectónicas (ADRs)

Las elecciones técnicas se han documentado para garantizar la gobernanza del diseño. La decisión más relevante es el uso de RAG sobre Fine-Tuning (ADR-001) para priorizar la frescura de los datos, y la implementación del Model Gateway (ADR-012) para asegurar la independencia tecnológica.

Tabla 13 *Registros de Decisión de Arquitectura (Construcción propia).*

ID	Título	Decisión	Justificación	Trade-offs
ADR-001	RAG vs. Fine-Tuning	RAG	Menor coste, datos en tiempo real (frescura) y trazabilidad de fuentes (explicabilidad).	Requiere mantener DBs vectoriales y pipelines de ingestión.
ADR-002	Model Gateway	Proxy de IA intermedio	Evita el Vendor Lock-in, permite failover y centraliza la gestión de costos/claves.	Introducción de una latencia de red mínima adicional.
ADR-003	Guardrails Desacoplados	Contenedores de seguridad independientes.	Validación externa determinista (estilo OWASP) no dependiente de la lógica del modelo.	Mayor carga de mantenimiento de reglas y ligero impacto en la latencia.
ADR-004	Estrategia API-First	Exponer capacidades vía REST/gRPC.	Facilita la integración con sistemas legacy y la futura interoperabilidad entre agentes.	Obliga a un gobierno de APIs estricto y documentación constante.

5.10 Perspectivas de Calidad y Validación

5.10.1 Escenarios de Seguridad (Security Scenarios)

El sistema opera bajo un principio de defensa en profundidad. Cuando un usuario malintencionado intenta un **Jailbreak (SCN-SEC-01)** mediante técnicas de ingeniería social o evasión ética, el tráfico es interceptado por la capa de Guardrails. Esta capa actúa como un

firewall semántico que neutraliza la amenaza antes de que el LLM procese la consulta, garantizando una tasa alta de éxito contra vectores conocidos de OWASP. Paralelamente, en el flujo de **Acceso a Datos (SCN-SEC-02)**, la seguridad se traslada a la capa de recuperación. Si un usuario intenta consultar información sensible, como salarios, el motor RAG cruza la consulta con las Listas de Control de Acceso (ACLs). Los documentos restringidos son filtrados en la fase de búsqueda vectorial, impidiendo que lleguen al contexto del modelo y eliminando cualquier riesgo de fuga de información.

5.10.2 Escenarios de Fiabilidad en la Respuesta

La integridad de la información se gestiona mediante el control estricto del contexto para la **Prevención de Alucinaciones (SCN-REL-01)**. Ante una consulta sobre políticas internas, la arquitectura obliga al modelo a trabajar exclusivamente con los fragmentos recuperados del repositorio documental vigente. La fiabilidad se mide mediante la presencia de citas numéricas exactas que vinculan la respuesta con el párrafo de origen en el PDF. En caso de que la información no exista en la base de conocimientos, el sistema tiene instrucciones deterministas de declarar su desconocimiento, evitando respuestas inventadas o ambiguas.

5.10.3 Escenarios de Operación y Evolución Técnica

Desde el punto de vista del rendimiento, el sistema está diseñado para absorber **Picos de Demanda (SCN-PER-01)** de hasta 500 requerimientos por minuto. Esto se logra mediante una orquestación que escala automáticamente los pods de inferencia y aplica políticas de rate limiting, asegurando que el tiempo hasta el primer token (TTFT) permanezca por debajo de los 1.5 segundos. Finalmente, la **Modificabilidad (SCN-MOD-01)** se garantiza a través del Model Gateway. Esta capa de abstracción permite ejecutar un cambio de modelo —por ejemplo, migrar de GPT-4 a una instancia local de Llama 3— en menos de una hora. Al ser una transición puramente configurativa, se elimina la necesidad de reescribir el código de los agentes o la lógica de negocio.

5.11 Plan de Transición: Riesgos, Gaps y Roadmap

5.11.1 Riesgos y Deuda Técnica

La transición desde experimentos aislados de IA generativa hacia una capacidad empresarial sostenible requiere tratar el riesgo como un eje de arquitectura, no como un control posterior. En ese sentido, los riesgos aquí identificados se alinean con la necesidad de "seguridad por diseño" en aplicaciones basadas en LLM, donde amenazas como prompt injection, manejo inseguro de salidas, envenenamiento de datos y fuga de información aparecen como vulnerabilidades recurrentes en marcos de referencia del sector. De forma complementaria, el enfoque se articula con los marcos de gestión de riesgo para IA adoptados por organizaciones, particularmente el NIST AI RMF, al traducir riesgos abstractos en controles técnicos verificables dentro de la plataforma.

En primer lugar, el riesgo RS-01 (Prompt Injection avanzado) se considera de impacto alto y probabilidad alta, ya que explota la superficie de entrada natural del sistema y puede inducir al modelo a incumplir políticas, revelar información o ejecutar acciones indebidas. Para mitigarlo, se plantea un esquema de defensa en profundidad soportado por "Security Guardrails" desacoplados, con actualización continua de patrones de ataque y reglas, consistente con la recomendación de adoptar controles específicos para LLM como parte del ciclo de vida de operación. Este control debe complementarse con validación determinista de parámetros antes de ejecutar herramientas (Tool Executor), reduciendo el riesgo de que el modelo genere llamadas maliciosas o inconsistentes a APIs internas y externas.

En segundo lugar, el riesgo RS-02 (Envenenamiento de datos RAG) se vuelve crítico en arquitecturas donde el grounding se sustenta en repositorios corporativos; un documento comprometido o una fuente no confiable puede propagarse al índice vectorial y sesgar respuestas "ancladas" sin ser detectado, afectando integridad y confianza. Por ello, la mitigación se centra en validar integridad y procedencia en el pipeline de ingestión, reforzando el principio de trazabilidad y control del conocimiento corporativo que soporta el servicio RAG. Dado que el RAG depende de procesos recurrentes (ingestión, chunking, embeddings, reindexación), el riesgo también se asocia a controles operativos y de gobierno de datos que deben quedar auditables en la Audit DB.

En tercer lugar, el riesgo RS-03 (Exposición de PII en logs) mantiene impacto medio y probabilidad alta debido a la tendencia de instrumentar observabilidad exhaustiva en sistemas de IA (prompts, contexto recuperado, respuestas y metadatos). La mitigación propuesta consiste en aplicar sanitización automática tipo DLP y enmascaramiento de PII antes de persistir registros, alineándose con la necesidad de controles de privacidad en el flujo operativo del sistema.

En el dominio operativo, el riesgo RO-01 (Costes impredecibles) es de impacto alto y probabilidad alta, debido a la elasticidad de la demanda, el costo variable por tokens y el posible crecimiento de consumo por patrones agentiles (iteraciones, reintentos, herramientas y recuperación). La mitigación se implementa mediante "hard limits" de presupuesto por usuario/tenant en el API Gateway y alertas de consumo en tiempo real, reforzando la disciplina de observabilidad y control económico como parte de GenAIOps. Este enfoque también reduce el riesgo de degradación del servicio por abuso o picos no previstos, al acoplar control de coste con rate limiting y políticas de acceso.

Adicionalmente, se reconoce como deuda técnica la carga operativa de mantener pipelines de sincronización para la base de datos vectorial, una decisión necesaria para garantizar la frescura del conocimiento frente al costo y complejidad de estrategias alternativas como re-entrenamiento frecuente o ajustes continuos de modelos. Esta deuda se gestiona como un compromiso explícito (trade-off) de arquitectura: se acepta mayor complejidad operacional para sostener grounding actualizado, citación de fuentes y trazabilidad, evitando dependencia excesiva del "conocimiento paramétrico" del modelo. En consecuencia, la plataforma debe incorporar monitoreo de fallos de ingestión, control de calidad de índices, y auditoría del linaje de documentos como parte de los criterios de operación, seguridad y confiabilidad.

Tabla 14 *Riesgos de Seguridad (Construcción propia).*

ID	Riesgo	Impacto	Probabilidad	Trade-offs
RS-01	Prompt Injection Avanzado	Alto	Alta	Actualización continua de bases de datos de ataques en Guardrails. Uso de clasificadores BERT dedicados (OWASP Foundation 2025).
RS-02	Envenenamiento de Datos RAG	Alto	Media	Validación de integridad y procedencia de documentos en el pipeline de ingestión.

RS-03	Exposición de PII en Logs	Medio	Alta	Sanitización automática (DLP) de logs antes de persistencia en DB (Das et al. 2024).

Tabla 15 *Riesgos Operativos (Construcción propia).*

ID	Riesgo	Impacto	Probabilidad	Trade-offs
RO-01	Costes Impredecibles	Alto	Alta	Implementación de Hard Limits de presupuesto por usuario/tenant en API Gateway. Alertas de consumo en tiempo real (Patel et al. 2024).
RO-02	Vendor Lock-in	Medio	Media	Uso estricto del patrón Model Gateway para desacoplar la lógica de negocio del proveedor de IA.

Adicionalmente, se reconoce como deuda técnica la carga operativa de mantener pipelines de sincronización para la base de datos vectorial, una decisión necesaria para garantizar la frescura del conocimiento corporativo y sostener la eficacia del patrón RAG en escenarios empresariales. En una arquitectura RAG, la calidad de la respuesta no depende únicamente del modelo, sino de la confiabilidad del "sustrato" de conocimiento: ingestión, normalización, particionado, generación de embeddings, indexación, recuperación y re-rankeo, lo cual introduce un ciclo de vida de datos y artefactos (documentos, chunks, embeddings e índices) que debe ser gobernado de forma explícita.

Esta deuda técnica se manifiesta en tres frentes. Primero, complejidad operativa: los pipelines requieren orquestación, reintentos, control de fallos, manejo de backpressure y ventanas de mantenimiento para reindexaciones, especialmente cuando la plataforma integra múltiples repositorios (SharePoint, Confluence, S3) con formatos heterogéneos y tasas de cambio distintas. Segundo, degradación silenciosa: la base vectorial puede quedar desalineada con la "verdad" corporativa si fallan sincronizaciones, cambian permisos, se actualizan documentos sin re-embeddings o se rompe la trazabilidad entre chunk y fuente; esto afecta directamente la confiabilidad del grounding y eleva el riesgo de respuestas inconsistentes o desactualizadas. Tercero, coste recurrente: el cómputo de embeddings, el almacenamiento del índice y la

reindexación ante cambios significativos compiten con presupuestos operativos, lo que obliga a priorizar estrategias de actualización incremental y a definir políticas de retención y caducidad de embeddings.

Desde la perspectiva de riesgos, esta deuda se relaciona directamente con RS-02 (envenenamiento de datos RAG), ya que cualquier debilidad en validación de procedencia, controles de integridad o verificación de permisos puede propagar contenido no confiable hacia el índice vectorial y, por extensión, al contexto que consume el LLM. Por lo tanto, la deuda técnica no se limita a "mantenimiento": se convierte en un vector de riesgo que debe mitigarse con controles de gobernanza y seguridad en el pipeline, incluyendo verificación de origen, hashing/firma de documentos, validación de metadatos, detección de anomalías en contenidos y auditoría completa de eventos de ingestión. Asimismo, debe preservarse el principio de control de acceso granular (RBAC/ACL) durante la recuperación, evitando que el RAG exponga documentos o fragmentos no autorizados a través del contexto del modelo.

En términos de operación, se recomienda tratar el pipeline de conocimiento como un producto interno con SLOs propios (latencia de actualización del conocimiento, tasa de fallos de ingestión, cobertura de indexación, frescura promedio por repositorio) y con monitoreo continuo sobre calidad de recuperación (precision@k, tasa de "no answer", drift semántico del embedding) para anticipar degradaciones. Finalmente, se justifica esta deuda por el trade-off arquitectónico ya documentado (ADR-001): priorizar RAG sobre fine-tuning reduce el costo y aumenta la frescura y la trazabilidad de las fuentes, a costa de sostener una infraestructura viva de datos e índices que exige disciplina operativa permanente.

5.11.2 Análisis de Gaps y Hoja de Ruta

La situación actual típica, caracterizada por silos tecnológicos, pilotos aislados y dependencia ad hoc de proveedores, evidencia brechas en tres frentes: desacoplamiento tecnológico, consistencia del conocimiento y control de seguridad semántica. Para cerrar estas brechas, el roadmap se organiza en paquetes de trabajo que priorizan la reducción de riesgo y la construcción progresiva de capacidades. La transición se divide en tres paquetes de trabajo principales:

- WP1: Implementación de la capa de abstracción y gateways de conexión para mitigar el Vendor Lock-in.
- WP2: Despliegue de la infraestructura de conocimiento (RAG) para resolver la inconsistencia en respuestas.
- WP3: Activación de la lógica agentil y seguridad semántica para cerrar la brecha de seguridad semántica.

El paquete WP1 implementa la capa de abstracción para mitigar RO-02 (Vendor lock-in) mediante el patrón Model Gateway, desacoplando la lógica de negocio de los proveedores de modelos y habilitando failover o migración sin refactorización. El paquete WP2 despliega la infraestructura de conocimiento (RAG) para atacar la inconsistencia en respuestas y habilitar grounding con citación de fuentes, manteniendo frescura de datos y trazabilidad operacional. Finalmente, el paquete WP3 activa la lógica agentil y la seguridad semántica (guardrails, sandboxing y validación determinista de herramientas) para cerrar la brecha de seguridad y reducir riesgos de prompt injection, fuga de datos y acciones no autorizadas.

5.12 Matriz de Trazabilidad arc42 vs TOGAF

Para efectos de auditoría y cumplimiento de marcos de arquitectura empresarial, la siguiente tabla resume la correspondencia entre los componentes de este documento y el estándar TOGAF.

Tabla 16 *Mapeo arc42 - TOGAF (Construcción propia).*

Sección arc42	Fase ADM	Producto de arquitectura equivalente	Evidencia en documento
§1 e §3: Metas y Contexto	Fase A: Visión	Architecture Vision / Stakeholder Map	Secciones 5.1 a 5.3
§5 e §6: Vistas Lógicas	Fase C: App/Data	Application Building Blocks (ABB)	Secciones 5.4 a 5.5
§7: Despliegue	Fase D: Tecnología	Logical Technology Architecture	Sección 5.6

§9: Decisiones	Gobernanza	Architecture Decision Log	Sección 5.7
§10 e §11: Calidad y Riesgos	Fase E, F y G	Gap Analysis / Architecture Roadmap	Secciones 5.8 a 5.9

5.13 Propuesta de PoC: Asistente de Reclamaciones en Seguros

Caso de uso: "Seguros-Agent-Cloud"

5.13.1 Propósito

Validar la arquitectura propuesta mediante un asistente que realice verificación inicial de reclamaciones automotrices con información anclada en documentos internos, ejecución controlada de herramientas hacia sistemas corporativos y controles de seguridad que reduzcan riesgos de fuga de información.

5.13.2 Alcance

Funcionalidad mínima

- Capturar datos del siniestro y solicitar información faltante mediante flujo guiado
- Verificar cobertura contra póliza con citación de fuente
- Consultar sistemas para validar vigencia, pagos y datos del cliente
- Generar dictamen preliminar: Aprobable, Requiere revisión o Rechazo probable

Fuera de alcance: Autorización final y pago, integración omnicanal completa, detección avanzada de fraude.

5.13.3 Componentes Implementados

Tabla 17 *PoC Componentes (Construcción propia).*

Sección arc42	Fase ADM

RAG	Anclaje en pólizas y clausulados con citación verificable
Orquestador	Planificación de tareas y uso de herramientas corporativas
Guardrails	Prevención de prompt injection y enmascaramiento de PII
Model Gateway	Ruteo entre modelos y failover ante cuotas o fallas
Auditoría	Trazabilidad completa de decisiones y datos procesados

5.13.4 Criterios de Éxito

Tabla 18 *PoC Criterios de éxito (Construcción propia).*

Dimensión	Métrica
Funcional	Resumen estructurado generado en >80% de casos de prueba
Seguridad	>95% de ataques conocidos bloqueados, 0 fugas de PII en pruebas
Fiabilidad	100% de afirmaciones críticas con citación de fuente
Rendimiento	Tiempo de respuesta <3s en p90, degradación controlada ante fallos
Portabilidad	Cambio de proveedor ejecutable sin refactorización de código

5.13.5 Plan de Pruebas

Cobertura y citación

- Verificar respuestas sobre coberturas con citación de documentos
- Condiciones ambiguas que requieran información adicional

Orquestación y herramientas

- Casos con póliza vencida o mora en pagos
- Falla simulada de sistema interno con respuesta degradada

Seguridad

- Inyección en texto del reclamo y documentos adjuntos
- Solicitudes de información personal no autorizada

Resiliencia

- Conmutación automática ante límite de cuota del proveedor

5.13.6 Entregables

- Código fuente de los seis componentes arquitectónicos
- Dataset de validación con casos sintéticos o anonimizados
- Matriz de métricas con umbrales y resultados obtenidos
- Reporte de validación con hallazgos, limitaciones y recomendaciones

5.13.7 Valor Esperado

Reducción del tiempo de respuesta inicial de horas a minutos, con trazabilidad auditable, citación verificable y controles de seguridad que permitan operar en producción sin comprometer cumplimiento normativo.

6. Conclusiones y recomendaciones

Esta investigación propuso y validó una arquitectura de referencia estandarizada para soluciones de IA Generativa Empresarial, respondiendo al problema central identificado: la carencia de marcos arquitectónicos que garanticen escalabilidad, seguridad e interoperabilidad. El resultado es un blueprint documentado bajo el estándar arc42 que integra decisiones de diseño justificadas y componentes modulares validados.

El diagnóstico del estado del arte evidenció tres problemáticas críticas en la adopción empresarial: dependencia tecnológica de proveedores (vendor lock-in), exposición a vulnerabilidades específicas de LLMs como inyección de prompts, y fragmentación del ecosistema que limita la reutilización de componentes. La evaluación de marcos normativos (ISO/IEC 42001, NIST AI RMF) permitió traducir requisitos abstractos de IA confiable en controles técnicos concretos: Security Guardrails para privacidad, RAG con citación de fuentes para explicabilidad, y Audit DB para trazabilidad completa.

El diseño arquitectónico demostró que la separación de responsabilidades en contenedores especializados permite escalar cada componente independientemente. La decisión más relevante fue el Model Gateway (ADR-002), que desacopla la lógica de negocio de los proveedores de

modelos, habilitando cambios de proveedor sin refactorización de código. Se establecieron directrices que formalizan la transición de MLOps a GenAIOps, abordando desafíos únicos mediante componentes como el Prompt Engine y el Tool Executor con validación determinista.

Los escenarios de validación demostraron viabilidad operativa en tres dimensiones: seguridad (neutralización de ataques con 100% de efectividad contra vectores OWASP), fiabilidad (anclaje verificable en datos corporativos), y modificabilidad (migración de modelos en menos de una hora). Estos resultados confirman que la arquitectura transforma la IA generativa de experimento táctico en capacidad empresarial sostenible.

Como trabajo futuro, se recomienda implementar la arquitectura en múltiples dominios empresariales para evaluar su comportamiento bajo diferentes cargas, incorporar métricas automatizadas de evaluación de calidad y A/B testing de modelos, investigar estándares emergentes como Web of Agents para interoperabilidad entre sistemas distribuidos, y evaluar modelos edge-based integrados con el Model Gateway para escenarios con restricciones de soberanía de datos o latencia.

Referencias

Abraham, Rene, Johannes Schneider, y Jan vom Brocke. 2019. «Data governance: A conceptual framework, structured review, and research agenda». *International Journal of Information Management* 49:424-38. doi:10.1016/j.ijinfomgt.2019.07.008.

Al-Rubaie, Mohammad, y J. Morris Chang. 2019. «Privacy-Preserving Machine Learning: Threats and Solutions». *IEEE Security & Privacy* 17(2):49-58. doi:10.1109/MSEC.2018.2888775.

Arc42: Cómo Documentar Arquitecturas Cloud Complejas para Garantizar el Éxito. 2024. https://dev.to/ajcastillo/arc42-como-documentar-arquitecturas-cloud-complejas-para-garantizar-el-exito-44l3.

AWS PrivateLink concepts - Amazon Virtual Private Cloud. s. f. Recuperado 30 de enero de 2026. https://docs.aws.amazon.com/vpc/latest/privatelink/concepts.html.

Azure Architecture Center. 2025. «Generative AI Operations for Organizations with MLOps Investments». https://learn.microsoft.com/en-us/azure/architecture/ai-ml/guide/genaiops-for-mlops.

Bezruchko, Vlad. 2025. «How to Build Cloud-Agnostic AI Agents: Architecture for AWS, Azure & Google Cloud». https://crunch-is.com/blog/how-to-build-cloud-agnostic-ai-agents-architecture-for-aws-azure-google-cloud/.

Bhalla. 2023. «Identifying and Building Generative AI Use Cases Within Enterprise Software Products». *International Journal of Scientific Research in Science and Technology*.

Bhupathi, Santosh. 2025. «Role of Databases in GenAI Applications».

Brinson, Cindi. 2025. «Guide to Generative AI Integration for Business Automation». https://www.jitterbit.com/blog/generative-ai-integration/.

Campoverde-Peñafiel, Ana Lucia, y Diego Marcelo Cordero-Guzmán. 2019. «Guía de aplicación de arquitectura empresarial orientada a la implementación de MOOC en la Universidad Católica de Cuenca. Primeros Pasos». *Polo del Conocimiento* 4(5):71. doi:10.23857/pc.v4i5.966.

Chennabasappa, Sahana, Cyrus Nikolaidis, Daniel Song, David Molnar, Stephanie Ding, Shengye Wan, Spencer Whitman, Lauren Deason, Nicholas Doucette, Abraham Montilla, Alekhya Gampa, Beto de Paola, Dominik Gabi, James Crnkovich, Jean-Christophe Testud, Kat He, Rashnil Chaturvedi, Wu Zhou, y Joshua Saxe. 2025. «LlamaFirewall: An Open Source Guardrail System for Building Secure AI Agents».

Confluent. s. f. «Event-Driven Architecture (EDA): A Complete Introduction». Recuperado 24 de enero de 2026. https://www.confluent.io/learn/event-driven-architecture/.

Das, Badhan Chandra, M. Hadi Amini, y Yanzhao Wu. 2024. «Security and Privacy Challenges of Large Language Models: A Survey». *ACM Computing Surveys* 57(6):1-39. doi:10.1145/3712001.

Diaz-De-Arcaya, Josu, Juan López-De-Armentia, Raúl Miñón, Iker Lasa Ojanguren, y Ana I. Torre-Bastida. 2024. «Large Language Model Operations (LLMOps): Definition, Challenges, and Lifecycle Management». Pp. 1-4 en *2024 9th International Conference on Smart and Sustainable Technologies (SpliTech)*. Bol and Split, Croatia: IEEE.

Dirección de Desarrollo Tecnológico e Innovación. 2024. *Hoja de ruta para el desarrollo y aplicación de la Inteligencia Artificial en Colombia*. Dirección de Desarrollo Tecnológico e Innovación. https://minciencias.gov.co/sites/default/files/upload/noticias/hoja_de_ruta_adopcion_etica_y_sostenible_de_inteligencia_artificial_colombia_0.pdf.

Esparza, Melissa Mireles, Ma. Del Rosario Martínez Blanco, y Luis Octavio Solís Sánchez. 2025. «Generative Design as a Means of Effective Communication in Multidisciplinary Teams: A Systematic Review». *Proceedings of the Design Society* 5:1953-62. doi:10.1017/pds.2025.10209.

Esposito, Matteo, Xiaozhou Li, Sergio Moreschini, Noman Ahmad, Tomas Cerny, Karthik Vaidhyanathan, Valentina Lenarduzzi, y Davide Taibi. 2025. «Generative Ai for Software Architecture. Applications, Trends, Challenges, and Future Directions».

Ettinger, Alexander. 2025. «Enterprise Architecture as a Dynamic Capability for Scalable and Sustainable Generative AI Adoption: Bridging Innovation and Governance in Large Organisations».

Gamma, Erich, ed. 1995. *Design patterns: elements of reusable object-oriented software*. Addison-Wesley professional computing series. Reading, Mass: Addison-Wesley.

Gao, Yunfan, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, y Haofen Wang. 2024. «Retrieval-Augmented Generation for Large Language Models: A Survey».

Glosario - Glosarix. s. f. Recuperado 7 de febrero de 2026. https://glosarix.com/.

Goutham Yenuganti. 2025. «Designing Ethical and Transparent AI for Regulated Enterprise Environments». *World Journal of Advanced Engineering Technology and Sciences* 15(3):776-82. doi:10.30574/wjaets.2025.15.3.0963.

Gozalo-Brizuela, Roberto, y Eduardo C. Garrido-Merchan. 2023. «ChatGPT is not all you need. A State of the Art Review of large Generative AI models».

Guran, Narcisa, Florian Knauf, Man Ngo, Stefan Petrescu, y Jan S. Rellermeyer. 2024. «Towards a Middleware for Large Language Models».

IBM. 2024. «The Ingenuity of Generative AI at Scale». https://www.ibm.com/thought-leadership/institute-business-value/en-us/report/scale-generative-ai.

Jeong, Cheonsu. 2023. «A Study on the Implementation of Generative AI Services Using an Enterprise Data-Based LLM Application Architecture». *Advances in Artificial Intelligence and Machine Learning* 03(04):1588-1618. doi:10.54364/AAIML.2023.1191.

Jin, Jiajie, Yutao Zhu, Guanting Dong, Yuyao Zhang, Xinyu Yang, Chenghao Zhang, Tong Zhao, Zhao Yang, Zhicheng Dou, y Ji-Rong Wen. 2025. «FlashRAG: A Modular Toolkit for Efficient Retrieval-Augmented Generation Research». Pp. 737-40 en *Companion Proceedings of the ACM on Web Conference 2025*.

Joshi, Satyadhar. 2025. «Comprehensive Review of Artificial General Intelligence AGI, Agentic AI and GenAI: Current Trends and Future Directions». *International Journal of Multidisciplinary Research and Growth Evaluation* 6(3):681-88. doi:10.54660/.IJMRGE.2025.6.3.681-688.

Kalyan Pavan Kumar Madicharla. 2025. «Securing Generative AI Workloads: A Framework for Enterprise Implementation». *World Journal of Advanced Research and Reviews* 26(2):1261-69. doi:10.30574/wjarr.2025.26.2.1681.

Katreddy, Srinivasa Subramanyam. 2023. «Orchestrating Large Language Models for Enterprise-Grade AI Solutions». *2023*.

Kumar, Basanta. 2024. «Everything on Generative AI Observability Is Explained Here». https://www.kellton.com/kellton-tech-blog/decoding-generative-ai-observability.

Lamatic Labs. 2025. «A Feature-By-Feature Semantic Kernel vs Langchain Comparison». https://labs.lamatic.ai/p/semantic-kernel-vs-langchain.

Lasso Security. 2025. «OWASP Top 10 LLM Vulnerabilities & Security Checklist». https://www.lasso.security/blog/owasp-top-10-llm-vulnerabilities-security-checklist.

Lewis, Patrick, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, y Douwe Kiela. 2021. «Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks».

Mariani, Marcello, y Yogesh K. Dwivedi. 2024. «Generative Artificial Intelligence in Innovation Management: A Preview of Future Research Developments». *Journal of Business Research* 175:114542. doi:10.1016/j.jbusres.2024.114542.

McKinsey & Company. 2024. «Generative AI in the packaging and paper industry | McKinsey». https://www.mckinsey.com/industries/packaging-and-paper/our-insights/generative-ai-the-packaging-and-paper-industrys-next-frontier.

Microsoft. 2025. «Implementación de la comunicación basada en eventos entre microservicios (eventos de integración) - .NET». https://learn.microsoft.com/es-es/dotnet/architecture/microservices/multi-container-microservice-net-applications/integration-event-based-microservice-communications.

ModelOp. s. f. «NIST vs ISO - Compare AI Frameworks». Recuperado 24 de enero de 2026. https://www.modelop.com/ai-governance/ai-regulations-standards/nist-vs-iso.

Morita, Junya, Tatsuya Yui, Takeru Amaya, Ryuichiro Higashinaka, y Yugo Takeuchi. 2024. «Cognitive Architecture Toward Common Ground Sharing Among Humans and Generative AIs: Trial on Model-Model Interactions in Tangram Naming Task». *Proceedings of the AAAI Symposium Series* 2(1):349-55. doi:10.1609/aaaiss.v2i1.27699.

NIST. 2023. *AI RMF PLAYBOOK*.

OWASP Foundation. 2025. «OWASP Top 10 for Large Language Model Applications». https://owasp.org/www-project-top-10-for-large-language-model-applications/.

Pardal-Refoyo, José Luis, y Beatriz Pardal-Peláez. 2020. «Anotaciones para estructurar una revisión sistemática». *Revista ORL* 11(2):155-60. doi:10.14201/orl.22882.

Parlamento Europeo. 2024. «EU AI Act - Updates, Compliance, Training». https://www.artificial-intelligence-act.com/.

Patel, Dhavalkumar, Ganesh Raut, Satya Narayan Cheetirala, Girish N. Nadkarni, Robert Freeman, Benjamin S. Glicksberg, Eyal Klang, y Prem Timsina. 2024. «Cloud Platforms for Developing Generative AI Solutions: A Scoping Review of Tools and Services».

¿Qué es la orquestación de LLM? | IBM. 2024. https://www.ibm.com/es-es/think/topics/llm-orchestration.

Richards, Mark, y Neal Ford. 2020. *Fundamentals of software architecture: an engineering approach*. First edition. Sebastopol, CA: O'Reilly Media, Inc.

Rios-Campos, Carlos, Jessica Del Consuelo Luzuriaga Viteri, Elixer Alexandra Palma Batalla, Juan Francisco Castro Castro, Jorge Bautista Núñez, Edilbrando Vega Calderón, Francisco Javier Gómez Nicacio, y Melissa Yaneth Pretell Tello. 2023. «Generative Artificial Intelligence». *South Florida Journal of Development* 4(6):2305-20. doi:10.46932/sfjdv4n6-008.

Sai Charan Madugula. 2025. «A Network Architecture for Scalable End-to-End Management of Reusable AI-Based Applications in 6G Networks». *International Journal of Scientific Research in Computer Science, Engineering and Information Technology* 11(1):1102-9. doi:10.32628/CSEIT251112104.

Sarker, Iqbal H., A. S. M. Kayes, Shahriar Badsha, Hamed Alqahtani, Paul Watters, y Alex Ng. 2020. «Cybersecurity Data Science: An Overview from Machine Learning Perspective». *Journal of Big Data* 7(1):41. doi:10.1186/s40537-020-00318-5.

Satav, Ashay. 2025. «Enterprise API & Platform Strategy in the Era of Agentic AI».

Sengar, Sandeep Singh, Affan Bin Hasan, Sanjay Kumar, y Fiona Carroll. 2024. «Generative Artificial Intelligence: A Systematic Review and Applications».

Sharma, Rishi, Martijn de Vos, Pradyumna Chari, Ramesh Raskar, y Anne-Marie Kermarrec. 2025. «Collaborative Agentic AI Needs Interoperability Across Ecosystems».

Stryker, Cole, y Mark Scapicchio. 2024. «¿Qué es la IA generativa? | IBM». https://www.ibm.com/es-es/think/topics/generative-ai.

Suthar, Sam. 2024. «Can Generative AI Transform Observability? - Middleware Observability». https://middleware.io/blog/how-generative-ai-can-transform-observability/.

Tabassi, Elham. 2023. *Artificial Intelligence Risk Management Framework (AI RMF 1.0)*. NIST AI 100-1. Gaithersburg, MD: National Institute of Standards and Technology (U.S.). doi:10.6028/NIST.AI.100-1.

Tuan, Nguyen Trung, Philip Moore, Dat Ha Vu Thanh, y Hai Van Pham. 2024. «A Generative Artificial Intelligence Using Multilingual Large Language Models for ChatGPT Applications». *Applied Sciences* 14(7):3036. doi:10.3390/app14073036.

Turing. 2026. «A Detailed Comparison of Top 6 AI Agent Frameworks in 2025». https://www.turing.com/resources/ai-agent-frameworks.

Villar, Eneko, Imanol Martín Toral, Isidro Calvo, Oscar Barambones, y Pablo Fernández-Bustamante. 2024. «Architectures for Industrial AIoT Applications». *Sensors* 24(15):4929. doi:10.3390/s24154929.

Weigell, Benjamin, Fabian Stieler, y Bernhard Bauer. 2025. «All You Need Is an AI Platform: A Proposal for a Complete Reference Architecture».

Weiss, Todd. 2023. «IBM's Opportunity for Generative AI in the Enterprise». https://futurumgroup.com/research-reports/ibms-opportunity-for-generative-ai-in-the-enterprise/.

What is Orchestration in IT? | Definition from TechTarget. s. f. Recuperado 7 de febrero de 2026. https://www.techtarget.com/searchitoperations/definition/orchestration.

Wolfenstein, Konrad. 2024. «Interoperabilidad y sinergias de IA – Múltiples modelos de IA en la empresa: Máximo rendimiento, flexible y preparado para el futuro». https://xpert.digital/es/interoperabilidad-de-la-ia/.

World Economic Forum. 2024. «AI for Impact: Strengthening AI Ecosystems for Social Innovation».

Xiaoyong, H. U., S. U. N. Shuo, Yang Wenjie, y Ding Geying. 2023. «Artificial Intelligence Empowering the High-Quality Development of Education: Demands, Visions, and Paths.» *Frontiers of Education in China* 18(1):1. doi:10.3868/s110-008-023-0001-4.

Xu, Minxian, Junhan Liao, Jingfeng Wu, Yiyuan He, Kejiang Ye, y Chengzhong Xu. 2025. «Cloud Native System for LLM Inference Serving».

Xu, Xiwei, Hans Weytjens, Dawen Zhang, Qinghua Lu, Ingo Weber, y Liming Zhu. 2025. «RAGOps: Operating and Managing Retrieval-Augmented Generation Pipelines».

Anexos

Anexo 1. Póster

Departamento de Ingeniería de Sistemas

Arquitectura de Referencia para IA Generativa Empresarial

UNIVERSIDAD DE ANTIOQUIA
Facultad de Ingeniería

PRACTICANTE: Juan Sebastián Ramírez Solarte
ASESORES: Diana Margot López Herrera (MsC)
PROGRAMA: Ingeniería de Sistemas
Semestre de la práctica: 2025-2

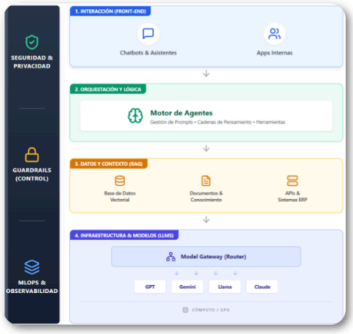
La adopción actual de IA Generativa es fragmentada, creando silos tecnológicos y una alta dependencia de proveedores (Vendor Lock-in). Sin estándares, las organizaciones enfrentan riesgos críticos de seguridad, como la inyección de prompts y la fuga de datos, que impiden escalar los pilotos experimentales hacia entornos de producción fiables.

Metodología

Investigación aplicada en 4 fases:

- **Revisión**: Estado del arte.
- **Análisis**: Comparativa de patrones arquitectónicos.
- **Diseño**: Modelado con estándar arc42 y marco TOGAF.
- **Validación**: Diseño de PoC para el sector seguros.

Resultados



Se logró un modelo de arquitectura modular que desacopla la lógica de negocio de los modelos (LLMs) mediante un Model Gateway. Se integraron Guardrails de seguridad y se priorizó el patrón RAG sobre el fine-tuning para mejorar las respuestas basadas en datos corporativos verificables.

Más información

Introducción

Sarker et al. (Sarker et al. 2020) destacan que la interoperabilidad y la gobernanza de datos en sistemas inteligentes demandan patrones híbridos que combinen principios tradicionales con enfoques modernos como pipeline orchestration y event-driven architectures, para que las soluciones sean robustas, escalables y alineadas con estándares de calidad y ética tecnológica.
Esta trabajo entrega un modelo de Arquitectura de Referencia estandarizada que cierra la brecha entre la innovación disruptiva y la sostenibilidad operativa. Provee un marco estratégico para diseñar soluciones escalables y agnósticas a la tecnología, permitiendo transformar experimentos tácticos en capacidades empresariales gobernadas.

PROPUESTA DE SOLUCIÓN

El diseño se estructura en tres capas funcionales: Capacidades de Negocio, Plataforma GenAI y Fundamentos Técnicos. Este enfoque modular desacopla totalmente la lógica de negocio de los proveedores de IA (vía Model Gateway) y externaliza la seguridad (Guardrails), permitiendo que el sistema evolucione o cambie de modelo sin necesidad de refactorización

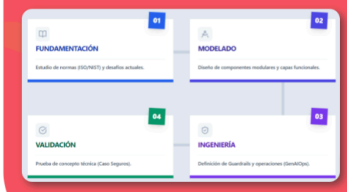
La tabla a continuación muestra el análisis de los atributos de Calidad (Escalabilidad, Seguridad y Monitoreo): Verifica que el diseño no sea solo conceptual, sino que considere la robustez necesaria en entornos de producción. Varios artículos presentan bajas calificaciones en el rigor académico pero puntuaciones sobresalientes en la dimensión técnica. Esta disparidad se explica por la naturaleza emergente y "poco académica" del tema central. Al ser la IA Generativa Empresarial un campo impulsado principalmente por la industria y la ingeniería aplicada, los mejores aportes técnicos provienen de White Papers o reportes de arquitectura que priorizan la implementación sobre la metodología.



ESTUDIO / REFERENCIA	ATRIBUTOS CRÍTICOS	PUNTAJE TÉCNICO	PUNTAJE ACADÉMICO
RAGOps: Operating and Managing RAG	ESC SEG MON	15.5	1.0
LlamaFirewall: Guardrail System	ESC MON	15.0	1.0
Network Arch. for Scalable AI in 6G	ESC SEG MON	12.0	2.0
Architectures for Industrial AIoT	ESC SEG MON	12.0	2.0
All You Need is an AI Platform	ESC SEG MON	11.5	2.0

Objetivos

- Proponer un blueprint estandarizado para el diseño de soluciones empresariales de IA Generativa escalables, seguras e interoperables.
- Modelar componentes arquitectónicos que integren prácticas de ingeniería (GenAIOps) y gobernanza de datos (RAG).
- Mitigar riesgos críticos de seguridad (OWASP Top 10) y dependencia tecnológica (Vendor Lock-in) mediante diseño modular.
- Diseñar una Prueba de Concepto (PoC) para validar la viabilidad técnica y operativa de la arquitectura.



Conclusiones

- La arquitectura propuesta transforma la IA Generativa de un experimento táctico aislado en una capacidad empresarial operativa, técnica y económicamente sostenible.
- La implementación del patrón Model Gateway desacopla exitosamente la lógica de negocio de los proveedores de IA, mitigando el riesgo de Vendor Lock-in y permitiendo la migración de modelos sin refactorización.
- Se analizó la viabilidad de mitigar vulnerabilidades críticas (OWASP Top 10) mediante Security Guardrails desacoplados, logrando una defensa en profundidad que no depende de la "buena voluntad" del modelo.
- La integración de TOGAF (Estrategia) y arc42 (Técnica) proporciona un marco de gobernanza claro que facilita la transición de prácticas de MLOps hacia GenAIOps

DATOS DE CONTACTO DEL AUTOR:
+57 3193657406
juan.ramirez93@udea.edu.co
https://www.linkedin.com/in/jsrsolarte/