



Pontificia Universidad  
**JAVERIANA**  
Cali

# Mejora de la atención al usuario en el bot de atención web de la Pontificia Universidad Javeriana: Implementación de IA y flujos de trabajo inteligentes en Salesforce

Rafael Hermida Toledo  
Juan Miguel Rojas Mejía

Proyecto de grado entregado para obtener el título de  
**Ingeniero de Sistemas y Computación**

Dirigida por  
PhD. Luisa Fernanda Rincón Pérez

Pontificia Universidad Javeriana Cali  
Facultad de Ingeniería y Ciencias  
Ingeniería de Sistemas y Computación  
Santiago de Cali  
21 de julio de 2025

---

Santiago de Cali, Julio 2025

Señores.

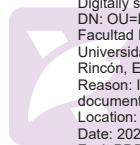
**Pontificia Universidad Javeriana Cali.**

PhD. Gerardo Mauricio Sarria, Director de Carrera  
Ingeniería de Sistemas y Computación  
Cali.

Cordial saludo,

Por medio de la presente me permito informarle que los estudiantes de Ingeniería de Sistemas y Computación, Rafael Hermida Toledo (cod: 8964994) y Juan Miguel Rojas Mejía (cod: 8963761) trabajaron bajo mi dirección en el proyecto de grado titulado “Mejora de la atención al usuario en el bot de atención web de la Pontificia Universidad Javeriana: Implementación de IA y flujos de trabajo inteligentes en Salesforce” el cual se encuentra finalizado y listo para sustentación.

Atentamente,



Digitally signed by Luisa Rincón  
DN: OU=Dpto Elect y Ciencias Comput -  
Facultad Ing y Ciencias, O=Pontificia  
Universidad Javeriana Cali, CN=Luisa  
Rincón, E=lfrincon@javerianacali.edu.co  
Reason: I agree to specified parts of this  
document  
Location:  
Date: 2025.11.11 16:26:18-05'00'  
Foxit PDF Reader Version: 2024.4.0

---

PhD. Luisa Fernanda Rincón Pérez

---

Santiago de Cali, Julio 2025

Señores.

**Pontificia Universidad Javeriana Cali.**

PhD. Gerardo Mauricio Sarria, Director de Carrera  
Ingeniería de Sistemas y Computación  
Cali.

Cordial saludo,

Nos permitimos presentar a su consideración el trabajo de grado titulado “Mejora de la atención al usuario en el bot de atención web de la Pontificia Universidad Javeriana: Implementación de IA y flujos de trabajo inteligentes en Salesforce” con el fin de cumplir con los requisitos exigidos por la Universidad para llevar a cabo el proyecto de grado y posteriormente optar al título de Ingeniero de Sistemas y Computación.

Al firmar aquí, damos fe que entendemos y conocemos las directrices para la presentación de trabajos de grado de la Facultad de Ingeniería aprobadas el 26 de Noviembre de 2009, donde se establecen los plazos y normas para el desarrollo del anteproyecto y del trabajo de grado.

Atentamente,



---

Rafael Hermida Toledo  
Código: 8964994



---

Juan Miguel Rojas Mejía  
Código: 8963761

---

## Abstract

In an increasingly interconnected world driven by technology, educational institutions face the challenge of responding promptly and efficiently to the information and support needs of their users. Traditional service models, based on face-to-face channels or static responses, are insufficient to handle the growing volume of inquiries and the expectation of immediate, personalized interactions. This creates the need for conversational systems that not only address frequently asked questions but also understand user context, process complex information, and automate administrative processes.

However, institutional virtual assistants often present significant limitations: they rely on predefined flows, lack advanced semantic understanding, and are not always integrated with internal systems to provide continuity to ongoing processes. This scenario is especially evident in universities, where academic, administrative, and service-related information changes constantly, and users expect up-to-date and reliable responses in real time.

In this context, the integration of Retrieval-Augmented Generation (RAG) architectures and intelligent automation workflows emerges as a viable solution to transform traditional bots into more powerful assistants. By combining semantic retrieval from vector databases with generative language models, it is possible to provide accurate, contextualized responses, while integration with CRM systems such as Salesforce allows the automation of key tasks such as lead management and PQRS case creation.

Thus, the main objective of this thesis is to design and implement an enhanced conversational solution for Pontificia Universidad Javeriana Cali. This proposal seeks to optimize the user experience on the web channel, delivering up-to-date information, natural language interaction, and the ability to automate administrative processes without manual intervention.

**Keywords:** Retrieval-Augmented Generation, Conversational Systems, Intelligent Automation, Salesforce, n8n, Qdrant, Generative Artificial Intelligence, User Support, CRM Integration, Semantic Retrieval.

---

## Resumen

En un mundo cada vez más interconectado y mediado por la tecnología, las instituciones educativas se enfrentan al reto de responder de manera oportuna y eficiente a las necesidades de información y acompañamiento de sus usuarios. Los modelos tradicionales de atención, basados en canales presenciales o respuestas estáticas, resultan insuficientes ante el volumen creciente de consultas y la expectativa de interacciones inmediatas y personalizadas. Esto plantea la necesidad de sistemas conversacionales que no solo resuelvan preguntas frecuentes, sino que también comprendan el contexto del usuario, procesen información compleja y automaticen procesos administrativos.

Sin embargo, los asistentes virtuales institucionales suelen presentar limitaciones importantes: dependen de flujos predefinidos, carecen de comprensión semántica avanzada y no siempre están integrados con los sistemas internos para dar continuidad a los procesos que inician. Este escenario es especialmente evidente en universidades, donde la información académica, administrativa y de servicios cambia constantemente, y los usuarios esperan respuestas actualizadas y confiables en tiempo real.

En este contexto, la integración de arquitecturas de Generación Aumentada por Recuperación (RAG) y flujos de automatización inteligentes se presenta como una solución viable para transformar los bots tradicionales en asistentes más potentes. Combinando recuperación semántica desde bases vectoriales con modelos generativos de lenguaje, es posible ofrecer respuestas precisas y contextualizadas, mientras que la conexión con sistemas CRM como Salesforce permite automatizar tareas clave como la gestión de leads y PQRS.

Así, el objetivo principal de este trabajo de grado es diseñar e implementar una solución conversacional mejorada para la Pontificia Universidad Javeriana Cali. Esta propuesta busca optimizar la experiencia del usuario en el canal web, brindando información actualizada, interacción en lenguaje natural y la posibilidad de automatizar procesos administrativos sin intervención manual.

**Palabras Clave:** Generación Aumentada por Recuperación, Sistemas Conversacionales, Automatización Inteligente, Salesforce, n8n, Qdrant, Inteligencia Artificial Generativa, Atención al Usuario, Integración CRM, Recuperación Semántica.

# Índice general

<b>1. Descripción del Problema</b>	<b>13</b>
1.1. Planteamiento del Problema . . . . .	13
1.1.1. Formulación . . . . .	14
1.1.2. Sistematización . . . . .	14
1.2. Objetivos . . . . .	15
1.2.1. Objetivo General . . . . .	15
1.2.2. Objetivos Específicos . . . . .	15
1.3. Justificación . . . . .	16
1.4. Delimitaciones y Alcances . . . . .	17
1.4.1. Alcances . . . . .	17
1.4.2. Limitaciones . . . . .	17
1.5. Metodología . . . . .	18
1.5.1. Tipo de Estudio . . . . .	18
1.5.2. Estrategia Metodológica . . . . .	18
<b>2. Marco Teórico y Trabajos Relacionados</b>	<b>20</b>
2.1. Marco Teórico . . . . .	20
2.2. Trabajos Relacionados . . . . .	24
2.3. Diferenciadores del Proyecto . . . . .	26
2.3.1. Principales Aportes de la Herramienta . . . . .	26

## ÍNDICE GENERAL

---

2.3.2. Limitaciones y Áreas de Mejora . . . . .	26
<b>3. Exploración del bot actual</b>	<b>28</b>
3.1. La implementación del bot . . . . .	29
3.2. Ventajas y desventajas del modelo . . . . .	30
<b>4. Casos de uso</b>	<b>34</b>
4.1. Reuniones con Alexander Valencia . . . . .	35
4.2. Reuniones con Carlos Gutiérrez . . . . .	36
4.3. Reuniones con Carlos Domínguez . . . . .	37
4.4. Requisitos del proyecto . . . . .	38
<b>5. Implementación</b>	<b>40</b>
5.1. Exploración de aproximaciones . . . . .	41
5.1.1. Conexión directa con la API de Gemini a través de Apex . . .	41
5.1.2. Uso de Salesforce Agentforce . . . . .	44
5.1.3. Solución definitiva mediante n8n . . . . .	47
5.2. Arquitectura de la solución final . . . . .	51
5.2.1. Análisis de intención . . . . .	52
5.2.2. Procesamiento según tipo de intención . . . . .	53
5.2.3. Componente RAG: recuperación y generación . . . . .	54
5.2.4. Memoria conversacional y enriquecimiento . . . . .	55
5.2.5. Ventajas de la arquitectura implementada . . . . .	55
5.3. Herramientas utilizadas . . . . .	56
5.4. Configuración del bot y funcionalidades implementadas . . . . .	57
<b>6. Validación del Sistema y Resultados</b>	<b>59</b>
6.1. Plan de Pruebas . . . . .	59

## ÍNDICE GENERAL

---

6.2.	Alcance de las Pruebas . . . . .	59
6.3.	Validación de Requisitos . . . . .	60
6.3.1.	Requisitos Funcionales . . . . .	60
6.3.2.	Requisitos No Funcionales . . . . .	61
6.4.	Escenarios de Prueba . . . . .	61
6.5.	Métricas de Evaluación . . . . .	62
6.6.	Validación de Integración . . . . .	63
6.7.	Resultados y Análisis . . . . .	63
6.8.	Resultados y Análisis . . . . .	63
6.8.1.	Resultados de las pruebas . . . . .	63
6.8.2.	Usuario del equipo desarrollador . . . . .	64
6.8.3.	Usuario perteneciente a la comunidad estudiantil . . . . .	67
6.8.4.	Usuario del equipo técnico universitario . . . . .	70
6.8.5.	Comparación de tiempos bajo concurrencia . . . . .	72
6.8.6.	Análisis de los resultados de las pruebas . . . . .	72
<b>7.</b>	<b>Conclusiones</b>	<b>74</b>
<b>8.</b>	<b>Trabajo Futuro</b>	<b>76</b>
8.1.	Trabajos Futuros Propuestos . . . . .	76
8.1.1.	Implementación en el Ambiente Productivo Institucional . . . . .	76
8.1.2.	Ampliación de Cobertura a Todas las Áreas de Acción del Bot . . . . .	77
8.1.3.	Entrenamiento Continuo y Aprendizaje Adaptativo . . . . .	77
8.1.4.	Implementación de Caché para Optimizar Costos . . . . .	77
8.1.5.	Análisis Predictivo y Métricas de Impacto . . . . .	78
8.1.6.	Integración con Fuentes de Conocimiento Dinámicas . . . . .	78
8.1.7.	Bots para Llamadas de Voz Basados en Transcripción . . . . .	78

## ÍNDICE GENERAL

---

8.1.8. Mejoras en la Experiencia Conversacional . . . . .	79
Apéndice A. Acuerdo de Confidencialidad . . . . .	80

# Índice de cuadros

4.1. Personas . . . . .	34
4.2. Requisitos funcionales del proyecto . . . . .	39
4.3. Requisitos no funcionales del proyecto . . . . .	39
6.1. Validación de Requisitos Funcionales del sistema RAG . . . . .	60
6.2. Validación de Requisitos No Funcionales del sistema RAG . . . . .	61
6.3. Criterios de evaluación de calidad de respuesta . . . . .	62
6.4. Comparación de tiempos de respuesta bajo diferentes niveles de concurrencia . . . . .	72
7.1. Participantes en la reunión de socialización y validación final del sistema	75

# Índice de figuras

2.1. Esquema general de una arquitectura RAG, combinando recuperación y generación. . . . .	23
3.1. Página de inicio de la Pontificia Universidad Javeriana Cali . . . . .	28
3.2. Ejemplo de un flujo implementado con Salesforce Flow . . . . .	29
3.3. Transferencia a asesor luego de dos intentos fallidos . . . . .	30
3.4. Redireccionamiento de consulta a WhatsApp . . . . .	31
3.5. Función de palabras atajo . . . . .	32
3.6. El bot entrega contenido desactualizado (calendario administrativo 2024) . . . . .	32
4.1. Flujo de tratamiento de las preguntas frecuentes . . . . .	35
4.2. Caso de uso de la generación automática de leads . . . . .	36
4.3. Flujo para escalar el caso a un agente humano . . . . .	37
4.4. Caso de uso de la generación automática de PQRS . . . . .	38
5.1. Respuesta acertiva del bot de Agentforce . . . . .	45
5.2. Inconsistencia en los documentos cargados a Agentforce . . . . .	46
5.3. Clasificación automática de intención, emoción y motivo del mensaje mediante el modelo Gemini. . . . .	48
5.4. Configuración del nodo nativo de Salesforce en n8n para la creación automática de leads a partir de datos capturados en la conversación .	49

## ÍNDICE DE FIGURAS

---

5.5. Configuración del nodo nativo de Salesforce en n8n para la creación automática de casos a partir de datos capturados en la conversación . . . . .	49
5.6. Vista de la interfaz gráfica para carga de archivos. . . . .	50
5.7. Diagrama de flujo general con enfoque RAG e integración CRM . . . . .	52
5.8. Solicitud de datos adicionales para generar el lead en Salesforce . . . . .	53
5.9. Solicitud de datos adicionales para generar el caso en Salesforce . . . . .	54

# Introducción

En la actualidad, las plataformas de atención web se han convertido en un recurso fundamental para la comunicación en tiempo real, tanto a nivel personal como institucional. Un bot de atención web, integrado en los sitios oficiales, facilita la interacción rápida y accesible, permitiendo el intercambio de mensajes, archivos y otro tipo de información de manera sencilla. En este contexto, muchas universidades han adoptado estos asistentes digitales para mejorar la comunicación con sus estudiantes, docentes y personal administrativo, aprovechando su facilidad de uso y su amplio alcance. Sin embargo, el potencial de estos canales aún no se ha explotado completamente, ya que la mayoría de los bots implementados carecen de capacidades avanzadas de análisis y personalización que optimicen realmente la experiencia de los usuarios.

La Pontificia Universidad Javeriana ha implementado un chatbot de atención web como vía principal para responder consultas y brindar soporte a sus usuarios de manera automatizada. A pesar de sus beneficios, el bot actual presenta limitaciones, ya que no cuenta con una capacidad profunda de análisis de interacciones ni con flujos de trabajo inteligentes que adapten las respuestas de manera dinámica a las necesidades y emociones de los usuarios. Esto representa una oportunidad significativa para explorar e implementar herramientas avanzadas de inteligencia artificial y procesamiento de lenguaje natural (NLP) que permitan caracterizar sentimientos e intenciones, optimizando así la calidad de la interacción y la satisfacción del usuario.

Este proyecto tiene como objetivo principal analizar y mejorar el funcionamiento del chatbot de atención web de la universidad mediante el uso de tecnologías de inteligencia artificial integradas en Salesforce, como Einstein Language, que permite el análisis de texto mediante NLP. A través de un enfoque exploratorio, se busca comprender las necesidades reales de los usuarios y desarrollar un sistema que adapte sus respuestas en función del análisis de emociones y tipos de consulta. Al final del proyecto, se espera obtener un sistema optimizado que no solo responda de manera eficaz, sino que también brinde una experiencia más personalizada y eficiente, logrando una mayor conexión y satisfacción en la comunidad universitaria.

# Capítulo 1

## Descripción del Problema

### 1.1. Planteamiento del Problema

Los bots de atención web se han convertido en una de las herramientas de comunicación más populares a nivel mundial. Su facilidad de uso, accesibilidad y la posibilidad de intercambiar mensajes, archivos e incluso contenido en tiempo real lo hacen un recurso poderoso tanto para usuarios como para organizaciones. En este sentido, un canal de atención web se refiere a un espacio de contacto oficial que una institución, como una universidad, habilita para interactuar con su comunidad. Su principal propósito es ofrecer una vía rápida y directa para resolver inquietudes, brindar soporte y mejorar la experiencia de comunicación entre sus usuarios.

En el contexto de la Pontificia Universidad Javeriana Cali, el uso de este bot de atención web se ha convertido en uno de los medios clave para atender consultas y mantener contacto con estudiantes, docentes y demás interesados. Su accesibilidad lo convierte en un recurso valioso para que la comunidad universitaria obtenga respuestas rápidas y efectivas, sin necesidad de recurrir a métodos más formales o lentos.

Con el fin de optimizar la comunicación mediante su plataforma web, la universidad ha implementado un bot de atención automatizada integrado a través de Salesforce, su plataforma de gestión de relaciones con el cliente (CRM). Este bot permite la interacción automática con los usuarios, respondiendo preguntas frecuentes, gestionando solicitudes básicas y generando respuestas rápidas, permitiendo una gestión más eficiente de los datos y ofreciendo a la universidad la posibilidad de mantener un registro detallado de todas las interacciones con los usuarios.

A pesar de tener esta infraestructura en funcionamiento, este bot actual se limita a ser solo una herramienta que contesta preguntas, lo que deja un aspecto que sigue sin ser explotado: la gran cantidad de datos generados a través de las interacciones. Si bien las consultas se almacenan y se registran, no se realiza un análisis profundo

que permita identificar patrones de consulta, temáticas recurrentes o incluso las emociones e intenciones detrás de las preguntas. Si se implementaran herramientas para analizar esta información, se podrían configurar flujos de trabajo inteligentes que automatizarían las respuestas basándose en el análisis de sentimientos e intenciones, redirigiendo automáticamente consultas complejas a un agente humano o guiando al usuario a la información adecuada. Esto mejoraría significativamente la manera en que la universidad responde, creando una experiencia mucho más personalizada y eficiente para los usuarios.

Aunque Salesforce ofrece herramientas de inteligencia artificial para optimizar estos procesos, su implementación va más allá de solo activar funciones predeterminadas. Requiere una configuración más avanzada y un análisis profundo de las necesidades específicas de la universidad.

### 1.1.1. Formulación

La pregunta ahora es: **¿Cómo aplicar las herramientas de inteligencia artificial de Salesforce para caracterizar las interacciones de los usuarios y configurar flujos de trabajo inteligentes en el bot de atención web de la Pontificia Universidad Javeriana?**

### 1.1.2. Sistematización

Para resolver esta pregunta se deben tener en cuenta algunas subpreguntas:

- ¿Cuál es la arquitectura y configuración técnica del bot de atención web de la universidad, incluyendo su programación, métodos de almacenamiento de registros y gestión del historial de conversaciones?
- ¿Qué criterios de selección se aplican para definir las interacciones del bot que se utilizarán en el entrenamiento del modelo de IA?
- ¿Cómo se pueden aprovechar las herramientas de IA de Salesforce en el desarrollo del proyecto para entrenar y mejorar continuamente el modelo de análisis de interacciones?
- ¿De qué manera se puede sacar provecho de los resultados del análisis de interacciones en la configuración de flujos de trabajo inteligentes?
- ¿Cómo se evaluará la precisión del modelo de clasificación y la efectividad de las respuestas automatizadas generadas por los flujos de trabajo inteligentes?

## 1.2. Objetivos

### 1.2.1. Objetivo General

Implementar flujos de trabajo inteligentes para el bot de atención automatizada de la Pontificia Universidad Javeriana utilizando las herramientas de inteligencia artificial de Salesforce, basados en el análisis de las interacciones de los usuarios en el bot de atención web.

### 1.2.2. Objetivos Específicos

- Analizar la arquitectura y configuración técnica del bot de atención web de la universidad, comprendiendo su programación, métodos de almacenamiento de registros y gestión del historial de conversaciones.
- Establecer los criterios de selección para las interacciones del bot que se utilizarán en el entrenamiento del modelo de inteligencia artificial.
- Entrenar el modelo de análisis de interacciones aprovechando las herramientas de inteligencia artificial de Salesforce.
- Configurar flujos de trabajo inteligentes que automatizan las respuestas del bot basadas en los resultados del análisis de interacciones.
- Evaluar el desempeño del bot en la generación de respuestas automatizadas y optimizar su comportamiento con base en los resultados de dicha evaluación.

### 1.3. Justificación

En la actualidad, las organizaciones, incluidas las universidades, dependen cada vez más de herramientas de atención web para interactuar con sus usuarios. Un bot de atención web, al ser una de las soluciones más versátiles, se ha convertido en un recurso fundamental para brindar soporte a estudiantes, docentes y demás miembros de la comunidad universitaria. Sin embargo, aunque este medio de contacto permite una comunicación rápida y eficiente, existe una gran cantidad de datos valiosos que no se están aprovechando al máximo. Cada intercambio de mensajes contiene información relevante sobre las consultas más frecuentes, las intenciones de los usuarios y sus percepciones, pero hasta el momento no se ha explorado cómo utilizar estos datos para mejorar la experiencia de usuario.

Aquí es donde entra en juego la inteligencia artificial, al implementar un modelo de IA capaz de analizar estos datos, se podrían identificar patrones que permitan generar respuestas más precisas y personalizadas. Salesforce, a través de herramientas como Einstein Language, facilita la clasificación de textos por sentimientos e intenciones. Aprovechar estas capacidades no solo optimizaría el servicio de atención a los usuarios, sino que también permitiría a la universidad automatizar respuestas a consultas frecuentes y redirigir las más complejas al personal adecuado.

Desde una perspectiva operativa, la implementación de IA no solo beneficiará a los usuarios que recibirán respuestas más rápidas y acertadas, sino que también reducirá la carga de trabajo del equipo de atención al usuario, permitiendo que se enfoquen en casos que requieren mayor intervención humana. Además, la viabilidad del proyecto es alta, ya que la universidad cuenta con la infraestructura necesaria, como la integración del CRM Salesforce, lo que permite una implementación fluida de herramientas como Einstein Language, sin la necesidad de grandes inversiones iniciales o rediseños complejos del sistema. Al aprovechar las herramientas ya existentes y escalar gradualmente las nuevas funcionalidades, se asegura un proceso de adaptación eficiente, reduciendo costos operativos y garantizando un uso más efectivo de los recursos tecnológicos.

Este proyecto permitirá a la universidad mejorar la interacción con su comunidad, optimizando sus procesos internos y asegurando una gestión más ágil y eficiente de las solicitudes, todo dentro de un marco de implementación técnica viable y financieramente sostenible.

## 1.4. Delimitaciones y Alcances

### 1.4.1. Alcances

El proyecto se centró exclusivamente en la optimización del bot de atención web de la Pontificia Universidad Javeriana mediante la integración de herramientas de inteligencia artificial disponibles en Salesforce. Dentro de este alcance se incluyó:

- El análisis de interacciones históricas del bot para extraer intenciones, emociones y temas frecuentes.
- La configuración de flujos de trabajo inteligentes que automatizaron respuestas y redireccionamientos en función del análisis previo.
- La validación del sistema implementado en un entorno controlado (*sandbox*) de Salesforce, evaluando métricas de precisión y efectividad del modelo.
- Este proyecto se limitó a trabajar con interacciones textuales ya almacenadas en el CRM, sin intervenir otros canales de atención ni otros sistemas de la universidad.

### 1.4.2. Limitaciones

- **Alcance institucional limitado:** El proyecto no contempló la integración del bot con canales alternos como WhatsApp, correo electrónico o redes sociales. Su enfoque se restringió exclusivamente al canal web ya implementado.
- **Limitaciones técnicas de Salesforce:** Algunas configuraciones avanzadas estuvieron restringidas por las licencias actuales o por la arquitectura del CRM institucional.
- **Despliegue restringido:** La implementación completa en producción dependió de la aprobación institucional. Este trabajo se limitó a una propuesta técnica y un entorno de pruebas validado, sin garantizar su puesta en marcha operativa inmediata.

## 1.5. Metodología

### 1.5.1. Tipo de Estudio

El presente proyecto adopta un enfoque de tipo exploratorio, ya que se centra en el análisis y mejora de un fenómeno aún poco estudiado en el ámbito académico: el uso de bots de atención automatizada en contextos universitarios. Este tipo de investigación resulta útil para comprender áreas emergentes, identificar variables relevantes y establecer una base conceptual para estudios futuros [1]. En particular, se busca mejorar la experiencia del usuario al interactuar con el bot web institucional, una solución que, si bien ya está en funcionamiento, presenta oportunidades claras de mejora en aspectos como la personalización, precisión de respuesta y adaptabilidad conversacional.

La elección de un enfoque exploratorio también permite flexibilidad metodológica, lo cual es fundamental para el desarrollo de soluciones tecnológicas iterativas y centradas en el usuario. Se busca así construir un conocimiento progresivo a partir de la interacción directa con los usuarios y de la experimentación práctica con las herramientas. La recopilación de información se llevará a cabo mediante reuniones técnicas con administrativos y miembros del equipo del Centro de Servicios Informáticos (CSI) de la universidad, con el fin de identificar patrones de uso, puntos críticos y expectativas en torno a la atención automatizada.

### 1.5.2. Estrategia Metodológica

La estrategia metodológica empleada combinó un enfoque de investigación aplicada con un desarrollo iterativo de prototipos. Se trabajó de manera cíclica entre análisis, diseño, implementación y validación, asegurando que cada fase respondiera a las necesidades institucionales identificadas.

En términos generales, el proceso metodológico se desarrolló en las siguientes etapas:

1. **Planeación y análisis:** se definieron los objetivos específicos del proyecto, se revisaron las limitaciones del bot existente y se identificaron los principales casos de uso a optimizar.
2. **Diseño y prototipado:** se planteó un diseño que integrara herramientas de inteligencia artificial y automatización para superar las limitaciones del modelo actual.
3. **Implementación progresiva:** se desarrollaron y configuraron los componentes necesarios para procesar mensajes, recuperar información relevante y automatizar tareas administrativas.

4. **Validación controlada:** se realizaron pruebas en un entorno seguro (*sandbox*) para medir la precisión en la clasificación de intenciones, los tiempos de respuesta del sistema y la calidad de las respuestas generadas. Adicionalmente, se evaluó la correcta ejecución de los flujos automatizados, así como la capacidad del bot para mantener coherencia conversacional en interacciones prolongadas.
5. **Socialización y retroalimentación:** se presentó el resultado final a los directivos y actores institucionales, quienes evaluaron su funcionamiento y expresaron su interés en su futura implementación productiva.

# Capítulo 2

## Marco Teórico y Trabajos Relacionados

### 2.1. Marco Teórico

El uso de la inteligencia artificial (IA) en la gestión de interacciones en tiempo real ha revolucionado la forma en que las organizaciones se comunican con sus usuarios. En el contexto de un bot de atención web, las tecnologías de procesamiento de lenguaje natural (NLP) y el análisis de sentimientos desempeñan un papel esencial para lograr una comunicación efectiva y automatizada. El NLP, respaldado por algoritmos de aprendizaje automático, facilita la comprensión del lenguaje humano, permitiendo a las máquinas interpretar y generar respuestas adecuadas basadas en la intención y el sentimiento del usuario. Estas herramientas son especialmente útiles en canales de atención automatizada en la web, donde la precisión y la personalización son fundamentales para garantizar una buena experiencia del usuario.

- **Procesamiento de Lenguaje Natural (NLP)**

El procesamiento de lenguaje natural es un campo de la inteligencia artificial que permite a las máquinas comprender, interpretar y generar lenguaje humano. Este campo ha ganado relevancia con el aumento de los datos textuales generados por las interacciones en plataformas de mensajería y redes sociales. Las aplicaciones de NLP abarcan desde la clasificación de texto hasta el análisis de sentimientos e intenciones, facilitando la automatización de respuestas en sistemas de atención al cliente basados en chatbots [2].

En el contexto de las interacciones del bot de atención web, el NLP permite que los sistemas automáticos como los chatbots identifiquen la intención detrás de las consultas, clasificando si el usuario busca información general, asistencia técnica o cualquier otro tipo de ayuda. Al incorporar técnicas de aprendizaje profundo y redes neuronales, el procesamiento de lenguaje puede mejorar

continuamente con el análisis de grandes volúmenes de datos conversacionales [3].

- **Large Language Models (LLM)**

Un Large Language Model (LLM) es un tipo de modelo de inteligencia artificial entrenado con enormes cantidades de texto mediante aprendizaje no supervisado o auto-supervisado, diseñado para comprender y generar lenguaje natural con alta fluidez y coherencia. Estos modelos están basados en arquitecturas neuronales profundas, las cuales permiten captar relaciones complejas entre palabras y construir respuestas contextuales [4, 5].

El uso de miles de millones de parámetros y grandes volúmenes de datos permite que los LLMs lleven a cabo tareas variadas, como redacción, traducción, resumen o chat conversacional incluso sin un entrenamiento específico en dichos dominios [4]. Gracias a su capacidad para inferir a partir de contexto, los LLMs se han convertido en herramientas fundamentales dentro de los sistemas conversacionales avanzados.

En la solución implementada para este proyecto, el LLM actúa como el núcleo generativo: recibe tanto la consulta del usuario como el contexto recuperado, integrándolos para producir respuestas coherentes, precisas y alineadas con el dominio institucional.

- **Análisis de Sentimientos e Intenciones**

El análisis de sentimientos se refiere a la capacidad de un sistema para identificar las emociones expresadas en un texto, determinando si una interacción es positiva, negativa o neutral. El análisis de intenciones, por su parte, busca entender el objetivo principal de una consulta. Estos dos tipos de análisis son esenciales en la gestión automatizada de conversaciones, ya que permiten no solo responder a las preguntas, sino también ofrecer respuestas más adecuadas a las emociones de los usuarios [2, 3].

Las tecnologías modernas permiten que este análisis se realice en tiempo real, ajustando las respuestas automáticas en función de la emoción o la intención detectada. En el caso de las universidades, esto podría traducirse en un manejo más efectivo de las solicitudes de los estudiantes, priorizando aquellas interacciones que requieren intervención humana o proporcionando respuestas automáticas que se adapten a las necesidades inmediatas del usuario [3, 6].

- **Salesforce y los Einstein Bots.**

Salesforce es una plataforma de *Customer Relationship Management* (CRM) basada en la nube que permite a las organizaciones gestionar de forma centralizada procesos de ventas, marketing, servicio al cliente y análisis de datos. A través de su Service Cloud, las empresas pueden automatizar flujos de atención, optimizar la gestión de casos y mejorar la experiencia del usuario mediante herramientas como Omni-Channel, Knowledge Base y automatizaciones de procesos [6].

Dentro de estas herramientas, los Einstein Bots son asistentes virtuales integrados en Salesforce que utilizan procesamiento de lenguaje natural (NLP) para interactuar con los usuarios, guiarlos en consultas frecuentes y, en caso necesario, escalar la conversación a un agente humano. Estos bots permiten la automatización de tareas repetitivas, la recopilación de datos relevantes y la creación de casos o registros directamente en el CRM, mejorando la eficiencia operativa y reduciendo los tiempos de respuesta [6].

- **Salesforce AgentForce**

AgentForce es una herramienta de Salesforce diseñada para crear agentes autónomos potenciados con inteligencia artificial, que pueden interactuar con usuarios en tiempo real de manera natural y personalizada. Esta plataforma permite construir agentes conversacionales que combinan capacidades de lenguaje natural, automatización de procesos y conectividad nativa con todo el ecosistema Salesforce. Su diseño modular está orientado tanto al soporte de empleados como a la atención de clientes, facilitando tareas como la gestión de casos, el acceso a datos contextuales o la generación de respuestas dinámicas mediante IA generativa. Gracias a su integración profunda con Salesforce Flow y Einstein, AgentForce representa una solución completa para el desarrollo de agentes conversacionales inteligentes y escalables [7].

- **Chatbots en la Atención Automatizada**

Los chatbots han evolucionado significativamente gracias a los avances en inteligencia artificial y NLP. En el contexto de la atención al cliente, los chatbots permiten una atención más rápida y eficiente, reduciendo la carga de trabajo del personal humano [8]. El uso de chatbots web para la atención automatizada es un ejemplo de cómo la tecnología puede gestionar consultas frecuentes y redirigir las más complejas a los agentes humanos.

El diseño de chatbots también ha avanzado hacia la personalización, ajustando las respuestas no solo en función de la consulta del usuario, sino también basándose en el análisis de su personalidad y comportamiento en interacciones anteriores. Esto permite una interacción más efectiva y mejora la experiencia general del usuario [6].

- **Configuración de Flujos de Trabajo Inteligentes**

La configuración de flujos de trabajo inteligentes es un aspecto clave en la automatización de respuestas en sistemas de atención automatizada. Los flujos de trabajo inteligentes permiten que el sistema determine automáticamente el mejor curso de acción según las interacciones previas y los análisis de intenciones y sentimientos. Por ejemplo, cuando se detecta que una consulta es particularmente compleja o que el usuario está expresando una emoción negativa, el sistema puede redirigir la consulta a un agente humano para que la gestione [9].

- **Arquitectura RAG**

La arquitectura RAG (Retrieval-Augmented Generation) es un enfoque que combina la generación de texto de modelos de lenguaje con un mecanismo de recuperación de información desde fuentes externas. A diferencia de los modelos puramente generativos, que responden únicamente con base en su conocimiento entrenado, un sistema RAG primero localiza fragmentos relevantes en una base documental y utiliza ese contenido como contexto para producir respuestas más precisas y actualizadas [10].

Este tipo de arquitectura es especialmente útil en entornos institucionales donde se requiere que las respuestas sean coherentes con reglamentos, calendarios u otros documentos oficiales. Además, permite mantener la información actualizada sin necesidad de reentrenar el modelo de lenguaje, simplemente añadiendo o modificando los documentos accesibles para la búsqueda [11, 10].

La figura 2.1 (Tomada de [12]) muestra de forma simplificada el flujo de trabajo en una arquitectura RAG, donde la consulta del usuario se complementa con información recuperada de una base documental antes de ser procesada por un modelo de lenguaje para generar una respuesta fundamentada.

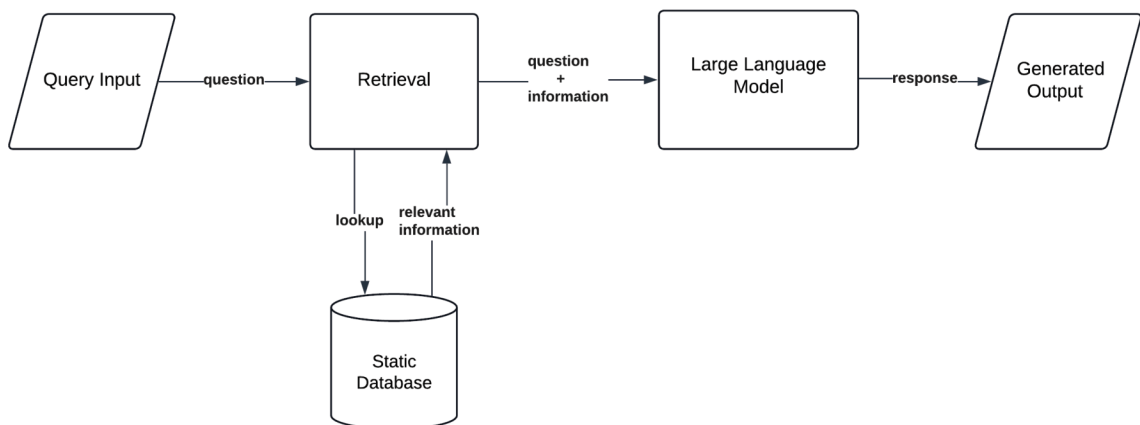


Figura 2.1: Esquema general de una arquitectura RAG, combinando recuperación y generación.

### ■ Embeddings semánticos

Un embedding es una representación digital de texto (palabras, frases o documentos) como un vector denso en un espacio de baja dimensión. A diferencia de métodos tradicionales que tratan cada palabra como una entidad aislada, los embeddings permiten capturar el significado semántico basándose en el contexto en el que aparecen [13, 14].

Conceptualmente, esto se basa en la hipótesis distribucional: “las palabras que aparecen en contextos similares tienen significados similares” [13]. El embedding mapea cada palabra a un punto en un espacio numérico donde la cercanía matemática refleja similitud semántica (por ejemplo, los vectores de “perro” y “canino” estarán cerca).

Los embeddings modernos son contextualizados, es decir, dos ocurrencias de la misma palabra pueden tener embeddings distintos según su contexto, resolviendo ambigüedades lingüísticas [13, 14]. Estos vectores permiten comparar significado mediante medidas matemáticas (como la similitud del coseno), facilitando procesos como la recuperación de información por similitud.

### **2.2. Trabajos Relacionados**

En esta sección se presentan diferentes trabajos de investigación, los cuales se basan en temáticas similares a las tratadas en este proyecto, o usan tecnologías relacionadas a este.

#### **Análisis de Chats de WhatsApp mediante Aprendizaje Supervisado**

Un estudio reciente abordó el análisis de conversaciones de WhatsApp mediante técnicas de aprendizaje supervisado para extraer patrones y clasificar el sentimiento de los usuarios. Utilizando algoritmos como Naïve Bayes y herramientas de análisis de sentimientos, este estudio demostró cómo las organizaciones pueden mejorar la atención al cliente mediante la automatización de la clasificación de interacciones. Estos modelos de análisis supervisado permiten identificar tendencias y ofrecer respuestas automatizadas más precisas basadas en el sentimiento detectado en los mensajes de los usuarios, lo cual se relaciona directamente con el propósito de este proyecto al optimizar la gestión de consultas mediante la IA [3].

#### **NLP para el Análisis de Sentimientos en Textos Japoneses**

Investigaciones recientes en el campo del procesamiento de lenguaje natural (NLP) han aplicado técnicas de aprendizaje profundo para analizar textos en japonés y clasificar las emociones expresadas. Este trabajo muestra cómo el uso de redes neuronales y técnicas avanzadas de análisis semántico permite mejorar la precisión del análisis de sentimientos [2].

#### **Automatización de Respuestas mediante Flujos de Trabajo Inteligentes**

La automatización de respuestas en sistemas de atención al cliente ha sido abordada en estudios que utilizan flujos de trabajo inteligentes para optimizar la resolución de consultas. Un estudio analizó cómo estas tecnologías permiten reducir los tiempos de respuesta y mejorar la satisfacción del usuario al automatizar tareas rutinarias y redirigir consultas más complejas a agentes humanos [6].

### **Análisis de Personalidad en Chats Grupales de WhatsApp utilizando IA**

Un estudio reciente analizó la personalidad de los participantes en chats grupales de WhatsApp mediante técnicas de inteligencia artificial y aprendizaje automático. Esta investigación muestra cómo los patrones de interacción y el análisis de personalidad pueden utilizarse para personalizar aún más las respuestas en sistemas automatizados de atención al cliente [15].

## 2.3. Diferenciadores del Proyecto

### 2.3.1. Principales Aportes de la Herramienta

- **Propuesta de integración con inteligencia artificial generativa:** A diferencia de otros enfoques centrados en reglas o NLP tradicional, este proyecto plantea el uso de modelos de IA generativa para mejorar la comprensión de las solicitudes de los usuarios, ampliar la cobertura temática y ofrecer respuestas más naturales y personalizadas.
- **Diseño de un prototipo funcional basado en casos reales:** Aunque el modelo aún no ha sido desplegado en un entorno 100 % operativo, se construye a partir del análisis de necesidades reales dentro de la universidad, usando como base el bot institucional actual.
- **Enfoque hacia la adaptabilidad conversacional:** El modelo está orientado a superar las limitaciones de interacción rígida, permitiendo flujos más dinámicos y abiertos. El uso de IA generativa permite interpretar diversas formas de expresión del usuario sin requerir coincidencia exacta con frases preconfiguradas.

### 2.3.2. Limitaciones y Áreas de Mejora

Teniendo en cuenta los artículos [6], [15], [3] y [8] expuestos anteriormente, es posible identificar algunas limitaciones relevantes en la naturaleza académica del proyecto de grado propuesto:

- **Dependencia tecnológica de Salesforce:** Dado que el proyecto se apoya exclusivamente en la infraestructura y herramientas del ecosistema Salesforce, su replicabilidad en instituciones con recursos distintos o sin acceso a esta plataforma puede verse limitada.
- **Falta de validación en un entorno productivo:** Aunque se diseña a partir de un caso real, el modelo aún no ha sido probado con usuarios finales en un entorno operativo, lo cual impide evaluar el impacto final del bot en la experiencia de atención o en la eficiencia del servicio.
- **Desconexión multicanal en la fase actual:** El prototipo actual no está habilitado para funcionar de forma simultánea o coherente en múltiples plataformas, lo que limita su alcance comunicativo.
- **Riesgos asociados al uso de IA generativa:** La incorporación de modelos generativos planteó desafíos en cuanto al control de respuestas, la coherencia con los lineamientos institucionales y la detección de errores o ambigüedades,

especialmente en contextos sensibles como la información académica o administrativa.

Para mitigar este riesgo, se ajustaron las instrucciones del bot, definiendo prompts restrictivos que limitaron su comportamiento a la información institucional disponible. Además, se añadió una política de escalamiento a un asesor humano para consultas fuera de contexto.

# Capítulo 3

## Exploración del bot actual

La Pontificia Universidad Javeriana de Cali, a través de su página web principal ([javerianacali.edu.co](http://javerianacali.edu.co)), ofrece a sus usuarios y visitantes la posibilidad de interactuar con un chatbot llamado ‘Ana’. En la figura 3.1 se observa la pestaña del bot en la esquina inferior derecha del portal. Este asistente virtual está diseñado para mejorar la experiencia de los usuarios, proporcionando respuestas inmediatas a preguntas frecuentes relacionadas con admisiones, programas académicos, requisitos, becas, procesos institucionales y otros temas de interés. Su implementación busca optimizar los canales de atención al público, reduciendo la carga de trabajo del personal administrativo, garantizando disponibilidad permanente las 24 horas del día y facilitando el acceso a la información.

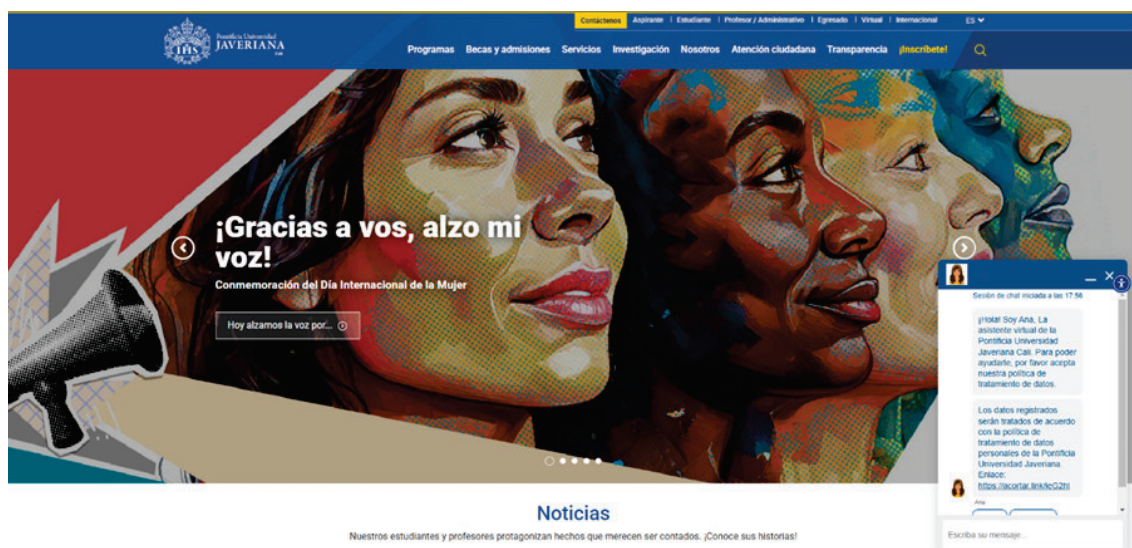


Figura 3.1: Página de inicio de la Pontificia Universidad Javeriana Cali

‘Ana’ funciona como un primer punto de contacto para quienes visitan el sitio web institucional, orientando a los usuarios de forma rápida sin necesidad de recurrir

a llamadas telefónicas o correos electrónicos. Además de ofrecer respuestas automatizadas, el bot permite redirigir algunas consultas hacia otros medios institucionales cuando es necesario, contribuyendo a una atención más eficiente y ordenada. En particular, ha sido una herramienta útil para futuros estudiantes interesados en conocer la oferta académica, requisitos de ingreso y fechas clave, así como para estudiantes actuales que buscan resolver inquietudes administrativas o académicas. Su existencia representa un paso hacia la modernización de los canales de atención universitaria y responde a la creciente necesidad de soluciones digitales inmediatas y accesibles.

### 3.1. La implementación del bot

El chatbot ha sido desarrollado utilizando las herramientas de Salesforce, el sistema de gestión de relaciones con clientes (CRM) de la universidad. En particular, se ha implementado a través de los Einstein Bots, una solución que permite la creación de agentes virtuales capaces de interactuar con los usuarios de manera estructurada. Esta herramienta funciona mediante flujos de conversación predefinidos creados en Salesforce Flow, los cuales establecen rutas específicas para la interacción con los usuarios, guiándolos mediante árboles de decisión y nodos conversacionales hacia la información que necesitan.

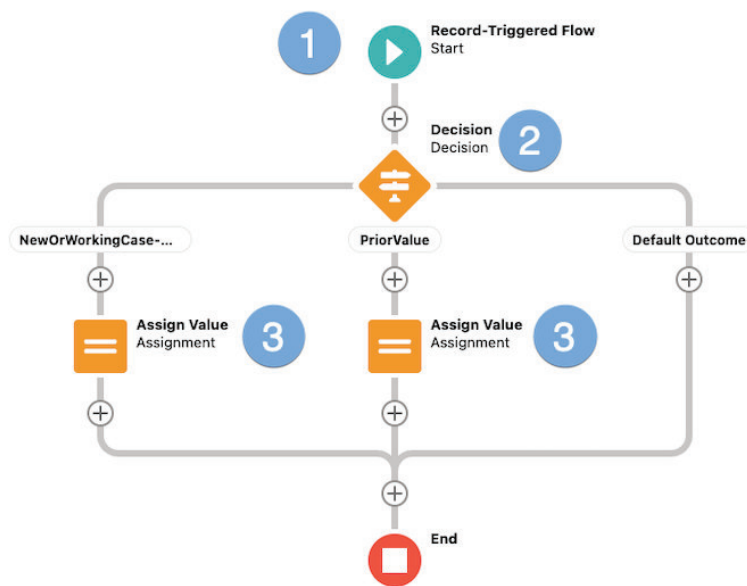


Figura 3.2: Ejemplo de un flujo implementado con Salesforce Flow

Salesforce Flow es una herramienta de automatización sin necesidad de código (no-code), que permite a los administradores y desarrolladores crear procesos complejos dentro del ecosistema Salesforce mediante una interfaz gráfica. Esta funcionalidad facilita la automatización de tareas, la recopilación de datos del usuario,

la aplicación de lógica condicional y la conexión con múltiples servicios sin necesidad de programar, optimizando la gestión de procesos y personalizando la experiencia del usuario [16].

Cada flujo conversacional está compuesto por elementos como mensajes, preguntas, reglas condicionales, asignaciones, validaciones de entrada y llamadas a otros procesos externos, lo que permite manejar interacciones complejas de forma estructurada [17]. Gracias a esta arquitectura modular, es posible construir conversaciones que soliciten datos al usuario, validen formatos (por ejemplo, correos electrónicos), desplieguen menús de opciones o redirijan al usuario según sus respuestas.

Además, el sistema incorpora capacidades de reconocimiento de lenguaje natural (NLU) que permiten identificar la intención del usuario con base en palabras clave o frases comunes. Una vez clasificada la intención, el bot despliega respuestas automatizadas, opciones de navegación o realiza validaciones. En caso de que el bot no pueda interpretar la solicitud después de dos intentos consecutivos, el sistema está configurado para escalar la conversación a un agente humano disponible (Figura 3.3), garantizando así una atención continua y adecuada [17].



Figura 3.3: Transferencia a asesor luego de dos intentos fallidos

### 3.2. Ventajas y desventajas del modelo

Una de las principales ventajas del chatbot es su capacidad para ofrecer atención automatizada las 24 horas del día, sin necesidad de intervención humana. Esta disponibilidad constante permite reducir la carga operativa del personal administrativo y atender de forma simultánea múltiples solicitudes, optimizando significativamente los recursos institucionales [18]. Asimismo, el bot ofrece una experiencia de usuario

ágil, con tiempos de respuesta inmediatos y navegación guiada a través de menús conversacionales.

El sistema también garantiza respuestas consistentes y alineadas con los mensajes institucionales, ya que estas son definidas previamente en los flujos. De esta forma, la atención que brinda se limita a respuestas predefinidas sobre temas como admisiones, becas, matrículas, programas académicos y otros procesos institucionales. En la mayoría de los casos, el bot complementa sus respuestas con enlaces a recursos oficiales, formularios o incluso a chats con asesores de la universidad (como puede verse en la figura 3.4), facilitando el acceso a la información.

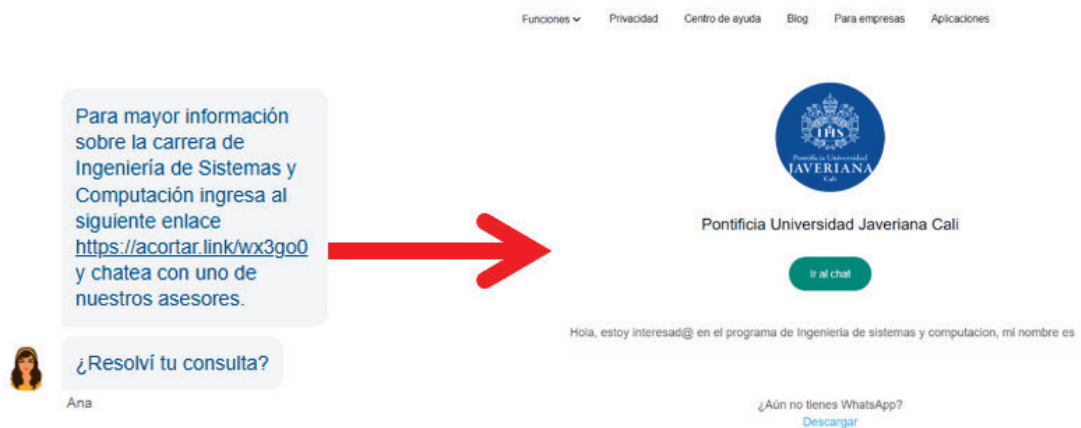


Figura 3.4: Redireccionamiento de consulta a WhatsApp

Sin embargo, esta misma estructura basada en reglas presenta algunas limitaciones. El bot no está diseñado para mantener conversaciones libres ni interpretar lenguaje ambiguo o natural de forma avanzada. La interacción depende en gran medida de que el usuario seleccione opciones del menú o escriba palabras clave exactas para activar los flujos apropiados. Como se muestra en la figura 3.5, aunque se utilizan atajos por palabras clave como “Modalidad Virtual”, estos requieren una coincidencia literal y no toleran variaciones en el texto. Por ejemplo, escribir “Estoy interesado en la modalidad virtual” no activa el flujo, mientras que “Modalidad Virtual” sí lo hace [17].

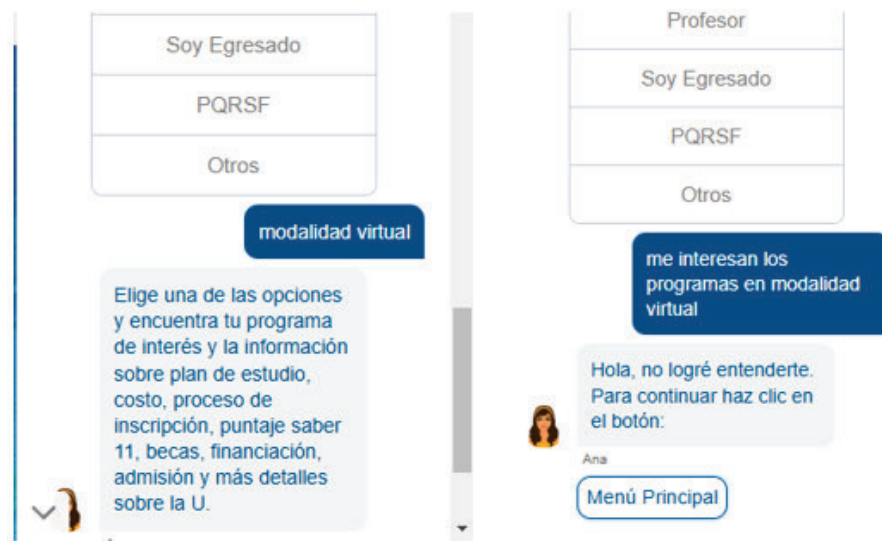


Figura 3.5: Función de palabras atajo

Más allá de estas restricciones conversacionales, se han identificado también problemas relacionados con la actualización de la información que ofrece el bot. Por ejemplo, como se evidencia en la figura 3.6, al consultar por el calendario administrativo vigente, el sistema proporciona el correspondiente al año 2024, lo que demuestra que su base de conocimiento no se actualiza de forma oportuna. Este desfase puede generar confusión entre los usuarios y disminuir la confianza en el canal automatizado, resaltando la necesidad de implementar mecanismos que permitan actualizar dinámicamente los contenidos institucionales.



Figura 3.6: El bot entrega contenido desactualizado (calendario administrativo 2024)

En conjunto, estas limitaciones evidencian la necesidad de incorporar herramientas más avanzadas de inteligencia artificial que permitan interpretar la intención y el contexto del usuario con mayor precisión, mejorar la personalización de las respuestas y garantizar que la información ofrecida sea actualizada y confiable [18].

# Capítulo 4

## Casos de uso

En el capítulo anterior se expuso que el bot implementado en la página web de la universidad cumple un papel relevante como canal de interacción con los visitantes. No obstante, se identificaron diversas limitaciones que revelan un amplio margen de mejora en su funcionamiento. A partir de esta observación, se definió el propósito de este proyecto de grado: proponer alternativas orientadas a optimizar el desempeño del bot.

Para alcanzar dicho propósito, fue fundamental identificar cuáles de las oportunidades de mejora tienen mayor relevancia e impacto potencial tanto en las operaciones institucionales como en los principales intereses de los usuarios. Con este objetivo en mente, una de las etapas clave del proyecto fue la gestión y coordinación de reuniones con los actores involucrados en el desarrollo y gestión del sistema. Estas interacciones permitieron recoger información valiosa sobre las necesidades actuales, las expectativas funcionales y las interacciones prioritarias que debería cubrir el bot. A continuación, se presentan las personas que participaron en esta etapa y sus respectivos aportes:

<b>Nombre</b>	<b>Cargo</b>
Alexander Valencia	Jefe CRM
Carlos Domínguez	Ingeniero proyectos de Desarrollo TI
Carlos Gutiérrez	Coordinador de análisis de datos TI

Cuadro 4.1: Personas

A continuación, se detallan los principales aportes que cada persona realizó al proyecto:

## 4.1. Reuniones con Alexander Valencia

Durante las sesiones de trabajo con Alexander Valencia, Jefe del área de CRM de la universidad, se identificaron una serie de requerimientos orientados a fortalecer las capacidades del bot institucional, especialmente en lo relacionado con la comprensión de lenguaje natural, el acceso a fuentes de información más completas y la cobertura de temas complejos que actualmente no pueden ser gestionados por los flujos automatizados existentes.

Uno de los principales puntos discutidos fue la necesidad de implementar un **componente de inteligencia artificial generativa** que permita al bot gestionar consultas que exceden el alcance de las preguntas frecuentes predefinidas en el bot ya existente. La propuesta consiste en incorporar un modelo de IA capaz de interpretar preguntas no estructuradas, redactadas en lenguaje natural, y generar respuestas adaptadas al contexto de la universidad. Este tipo de solución permitiría ampliar la cobertura temática del bot sin depender exclusivamente de coincidencias exactas con frases preconfiguradas. El componente generativo se activaría únicamente en los casos en que los flujos tradicionales no puedan ofrecer una respuesta satisfactoria, funcionando así como un recurso complementario y controlado dentro del sistema de atención automatizada.

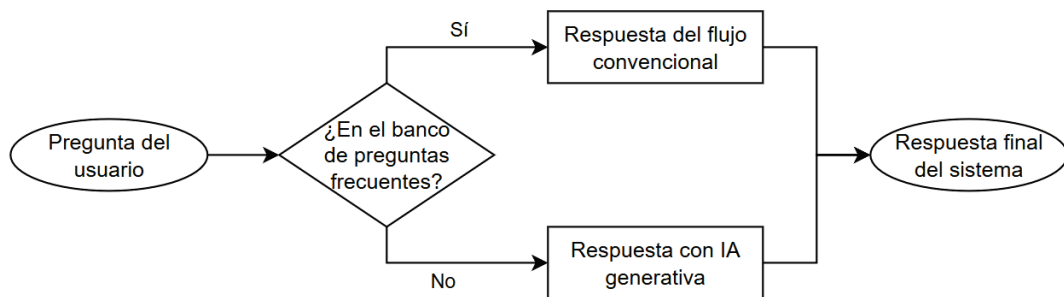


Figura 4.1: Flujo de tratamiento de las preguntas frecuentes

Adicionalmente, se planteó la posibilidad de utilizar **videos como fuente de datos** para enriquecer las respuestas proporcionadas por el bot. Este requerimiento surge a partir de la observación de que muchos contenidos institucionales se encuentran disponibles únicamente en formato audiovisual. La idea consiste en habilitar al bot para que pueda acceder a las transcripciones de dichos videos y, a partir de ellas, ofrecer respuestas basadas en texto que correspondan con la intención detectada en la consulta del usuario. Esta funcionalidad no solo ampliaría el repertorio de respuestas posibles, sino que también permitiría reutilizar recursos ya existentes dentro de la universidad para mejorar la experiencia de atención.

## 4.2. Reuniones con Carlos Gutiérrez

Durante las reuniones sostenidas con Carlos Gutiérrez, Coordinador de análisis de datos del área TI, se identificó un requerimiento enfocado en mejorar el aprovechamiento de las interacciones generadas en el bot mediante la **automatización del registro de leads** dentro del sistema CRM institucional.

Actualmente, una parte significativa de las consultas que llegan a través del canal web corresponde a personas interesadas en programas académicos, procesos de admisión, becas o actividades institucionales. Sin embargo, dicha información no se traduce de forma automática en acciones dentro del sistema de gestión de relaciones con los usuarios, lo cual representa una oportunidad desaprovechada para hacer seguimiento a esos posibles contactos.

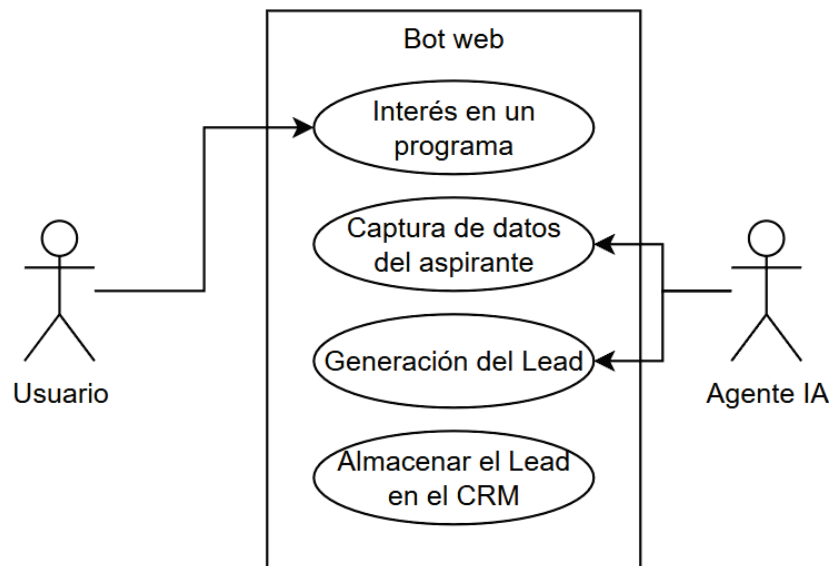


Figura 4.2: Caso de uso de la generación automática de leads

El requerimiento plantea que el bot sea capaz de identificar cuándo una interacción corresponde a un usuario con potencial interés institucional, como un aspirante o futuro estudiante. En esos casos, el sistema debería extraer y almacenar los datos básicos suministrados por el usuario (nombre, correo, interés académico, entre otros) y generar automáticamente un **lead en Salesforce**.

En el contexto del CRM, un *lead* representa un registro preliminar que identifica a una persona o entidad que ha mostrado algún tipo de interés en los servicios de la institución, pero que aún no ha sido contactada formalmente o no ha iniciado un proceso definido. Estos leads son almacenados dentro del módulo de prospección de Salesforce, desde donde pueden ser clasificados, segmentados y transferidos a otras áreas (como admisiones o promoción académica) para su posterior gestión [19].

La propuesta también contempla que el bot sea capaz de recopilar los datos necesarios de manera progresiva durante la conversación, evitando interrupciones en la experiencia del usuario.

### 4.3. Reuniones con Carlos Domínguez

Durante las reuniones con Carlos Domínguez, Ingeniero de proyectos del área de Desarrollo TI, se identificaron dos requerimientos orientados a mejorar la forma en que el bot gestiona situaciones que superan su capacidad de respuesta automatizada. Ambos puntos apuntan a fortalecer la conexión entre el bot y los mecanismos institucionales de soporte humano.

El primer requerimiento consistió en **fortalecer el mecanismo de escalamiento hacia agentes humanos**. Actualmente, el sistema cuenta con una funcionalidad básica que permite transferir algunas conversaciones a personal institucional, sin embargo, esta derivación no siempre se realiza de manera eficiente ni basada en criterios definidos. La propuesta sugiere que el bot sea capaz de reconocer con mayor precisión los casos que requieren intervención humana, ya sea porque el usuario lo solicita explícitamente, porque se detecta frustración en el lenguaje, o porque la consulta no coincide con ninguno de los flujos predefinidos. Ante estas condiciones, el sistema debería activar un protocolo de escalamiento que permita redirigir al usuario a un agente disponible. Además, se considera fundamental que el historial de la conversación sea incluido en el traspaso, evitando que el usuario deba repetir su solicitud.

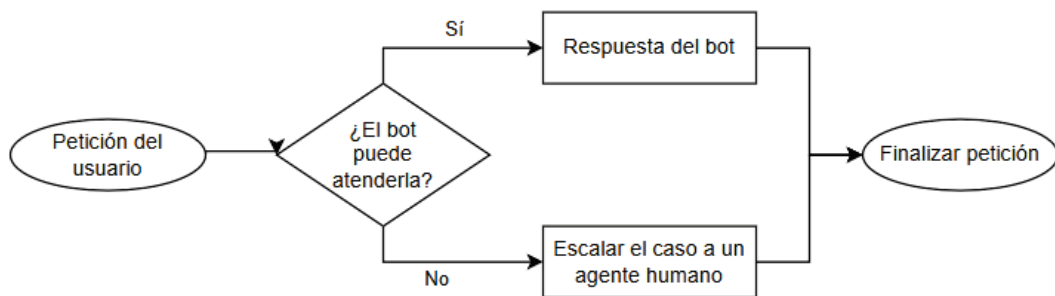


Figura 4.3: Flujo para escalar el caso a un agente humano

El segundo requerimiento está relacionado con la **creación automática de casos tipo PQR (Peticiónes, Quejas o Reclamos)** directamente desde el bot. Actualmente, el proceso de radicación de PQR requiere que el usuario acceda a un formulario externo o realice la gestión por canales adicionales. La propuesta sugiere que el bot sea capaz de registrar un caso PQR en el sistema de gestión de solicitudes

de la universidad cuando detecte que el usuario manifiesta una inconformidad o expresa explícitamente una queja o petición que no puede ser resuelta en ese momento. El registro incluiría los datos del usuario, la descripción del caso, y un identificador vinculado a la conversación previa. Estos casos quedarían almacenados en el CRM institucional, desde donde serían gestionados por las dependencias responsables.

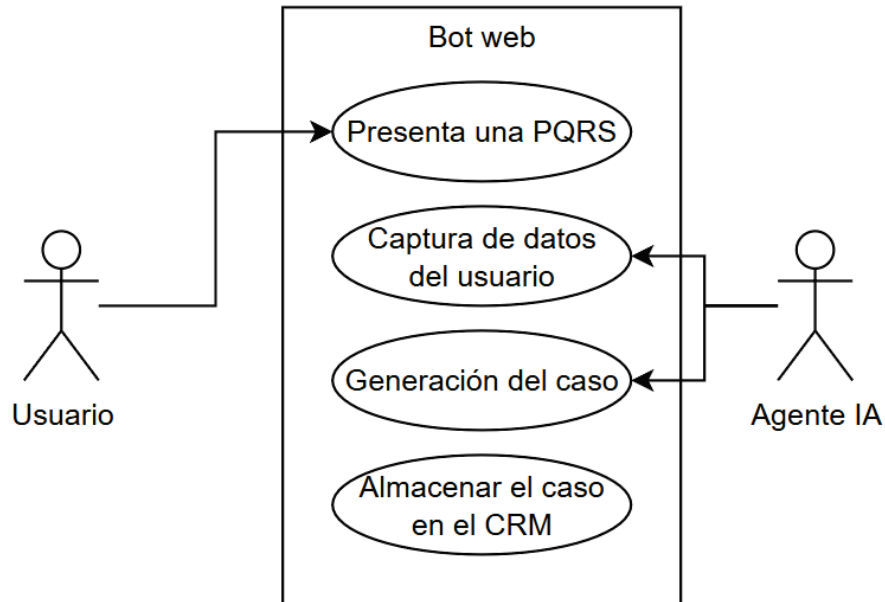


Figura 4.4: Caso de uso de la generación automática de PQRS

## 4.4. Requisitos del proyecto

A partir del análisis de los casos de uso y los requerimientos identificados durante las reuniones con los actores involucrados en la gestión y desarrollo del bot institucional, se definieron los requisitos funcionales y no funcionales que guiarán la implementación de la solución propuesta.

Los requisitos funcionales describen las capacidades que el sistema debe ofrecer para cumplir con los objetivos planteados, mientras que los no funcionales establecen las condiciones de calidad, rendimiento, escalabilidad e integración que debe garantizar la arquitectura del proyecto.

A continuación, se presentan ambos conjuntos de requisitos.

### Requisitos funcionales

Código	Descripción
RF_01	El bot debe interpretar consultas en lenguaje natural, sin depender de coincidencias exactas con palabras clave.
RF_02	El sistema debe registrar automáticamente leads en Salesforce cuando detecte interés en programas académicos.
RF_03	Debe permitir el escalamiento de conversaciones a agentes humanos cuando no pueda resolver la consulta o detecte frustración.
RF_04	El bot debe gestionar la creación automática de casos PQR dentro del CRM.
RF_05	El sistema debe recuperar información institucional desde una base vectorial para enriquecer las respuestas.
RF_06	Debe generar respuestas contextuales mediante IA generativa cuando las consultas excedan los flujos predefinidos.
RF_07	Mantener una memoria conversacional que preserve el contexto en interacciones prolongadas.

Cuadro 4.2: Requisitos funcionales del proyecto

### Requisitos no funcionales

Código	Descripción
RNF_01	El sistema debe cumplir con políticas de seguridad y privacidad institucional para el tratamiento de datos de los usuarios.
RNF_02	La arquitectura debe ser escalable para integrar nuevas fuentes documentales sin reentrenar el modelo.
RNF_03	Debe integrarse nativamente con Salesforce para registrar leads y casos sin procesos manuales.
RNF_04	La solución debe ser mantenible, permitiendo actualizar flujos y documentos sin afectar toda la arquitectura.
RNF_05	Debe estar disponible 24/7, soportando múltiples usuarios simultáneamente sin pérdida de rendimiento.
RNF_06	La interfaz conversacional debe ser clara y accesible para garantizar una experiencia de usuario inclusiva.

Cuadro 4.3: Requisitos no funcionales del proyecto

# Capítulo 5

## Implementación

A partir del análisis realizado en los capítulos anteriores, donde se identificaron limitaciones en el funcionamiento del bot institucional en cuanto a comprensión de lenguaje natural, cobertura temática y capacidad de adaptación a distintos tipos de consulta, se dio inicio a la fase de implementación de la solución propuesta. Esta etapa tomó como base los requerimientos definidos a partir de las reuniones con actores clave del proceso, quienes expusieron casos de uso concretos como la creación automática de leads, la derivación de consultas a agentes humanos, la integración de contenido audiovisual como fuente de información, y la incorporación de capacidades de generación de lenguaje natural para atender consultas más complejas. Con base en estos insumos, la implementación tuvo como objetivo incorporar herramientas que permitieran ampliar las capacidades del bot, integrarlo de manera más efectiva con el CRM institucional y ofrecer una experiencia de atención automatizada más flexible y contextualizada.

Como parte del proceso de implementación, se exploraron inicialmente tres alternativas técnicas para ampliar las capacidades del bot. La primera consistió en integrar un modelo generativo al flujo del bot original mediante el uso de la API de Gemini. La segunda alternativa fue el uso de Agentforce, una funcionalidad nativa de Salesforce diseñada para construir agentes conversacionales basados en inteligencia artificial. Finalmente, se optó por una tercera alternativa como solución definitiva: la utilización de n8n, una herramienta de automatización de flujos que permitió orquestar llamadas a modelos generativos externos e integrar resultados de manera controlada dentro del CRM institucional.

Este capítulo inicia con la descripción de las tres aproximaciones evaluadas durante el proceso de desarrollo, junto con los criterios técnicos y funcionales que llevaron a la elección de la solución final implementada.

### 5.1. Exploración de aproximaciones

Se describen las tres alternativas consideradas durante el diseño. Incluyendo ventajas y limitaciones de cada una.

#### 5.1.1. Conexión directa con la API de Gemini a través de Apex

La primera aproximación evaluada para extender las capacidades del bot institucional consistió en integrar de forma directa un modelo generativo de lenguaje mediante la **API de Gemini**, exponiéndolo como un servicio accesible desde el bot Einstein configurado en el *sandbox* institucional.

#### ¿Qué es el sandbox de Salesforce?

En Salesforce, el término *sandbox* hace referencia a un entorno aislado utilizado para pruebas y desarrollo, el cual replica la configuración y parte de los datos del entorno productivo, pero sin afectar la operación real de la organización. En este proyecto, el bot Einstein se encontraba desplegado en dicho sandbox, lo que permitió realizar integraciones y experimentos sin comprometer los procesos activos del CRM institucional.

#### Proceso de acceso al sandbox institucional

Para acceder al *sandbox* institucional de Salesforce, fue necesario realizar un proceso de autorización formal gestionado por el Centro de Servicios Informáticos de la Pontificia Universidad Javeriana Cali. Como parte de este proceso, se suscribió un Acuerdo de Confidencialidad (NDA) que garantizó el manejo adecuado de la información contenida en el entorno de pruebas y estableció las restricciones para la divulgación de configuraciones y datos internos.

Una vez firmado el acuerdo, se concedieron las credenciales temporales necesarias para trabajar en el entorno seguro, permitiendo realizar las integraciones entre el bot Einstein, n8n, Qdrant y Gemini de manera controlada, sin afectar el CRM productivo de la universidad.

En el Apéndice 8.1.8 se incluye el texto completo del Acuerdo de Confidencialidad firmado entre el equipo desarrollador y la Pontificia Universidad Javeriana Cali, documento que respalda el acceso al *sandbox* institucional durante la ejecución del proyecto.

### Descripción del flujo de integración

Para establecer la comunicación entre el bot Einstein y la API de Gemini se implementó un **servicio intermedio** desarrollado con **Flask**, un *framework* ligero para Python que facilita la creación de aplicaciones web y APIs de manera rápida, sin requerir una configuración compleja. Esta característica lo convirtió en una opción adecuada para una prueba de concepto como la que se necesitaba en este proyecto. Mediante Flask se definió una ruta HTTP capaz de recibir solicitudes, invocar la API de Gemini con el mensaje del usuario y retornar el resultado al bot, actuando así como puente entre ambas plataformas.

Este servicio funcionaba como un punto de enlace o endpoint que recibía los mensajes enviados desde Salesforce, los procesaba y devolvía la respuesta generada por el modelo.

El flujo técnico de esta aproximación puede resumirse de la siguiente manera:

1. El usuario ingresaba un mensaje en el bot Einstein del sandbox.
2. Mediante un endpoint generado en Apex (el lenguaje de programación de Salesforce), se enviaba el mensaje en formato JSON hacia el endpoint expuesto por el servicio Flask.
3. El servicio intermedio recibía el mensaje, invocaba la API de Gemini y lo procesaba como *prompt*.
4. Gemini generaba una respuesta en lenguaje natural, devolviendo el resultado al servicio intermedio.
5. Finalmente, el endpoint respondía al bot Einstein, que mostraba el mensaje generado al usuario.

### Implementación del endpoint

Para implementar el servicio intermedio se desarrolló una pequeña aplicación con **Flask** que recibía las solicitudes desde el bot Einstein, invocaba la API de Gemini y retornaba la respuesta generada.

Un fragmento simplificado de la definición del endpoint es el siguiente:

```
@app.route('/process_message', methods=['POST'])
def process_message():
    user_message = request.json.get('message')
    response = generate(user_message)
    return jsonify({'response': response.strip()})
```

El código completo de la solución, incluyendo la configuración de la función `generate()` y los parámetros utilizados para interactuar con el modelo generativo, se encuentra disponible en el repositorio público [20]

### Explicación del código y configuración

Este servicio está compuesto por los siguientes elementos clave:

- **Importación de librerías:**
  - Flask para crear el servidor web y definir el endpoint.
  - Google Generative AI (`genai`) para interactuar con el modelo Gemini.
- **Función `generate(prompt)`:** Recibe el mensaje del usuario (`prompt`) y configura una solicitud al modelo `gemini-2.0-flash-lite`. Se establecieron parámetros de generación con criterios específicos:
  - `temperature=0.5`: se eligió un nivel de creatividad intermedio, permitiendo respuestas variadas pero sin perder coherencia.
  - `top_p=0.95`: asegura que el modelo priorice las palabras más probables, filtrando opciones menos relevantes.
  - `top_k=40`: limita la cantidad de candidatos que el modelo considera por cada paso, equilibrando diversidad y precisión.
  - `max_output_tokens=8192`: define la longitud máxima de respuesta, garantizando suficiente espacio para preguntas largas sin generar texto excesivo.

El modelo devuelve la respuesta en forma de *streaming*, concatenando cada fragmento en la variable `value` para ensamblar el mensaje final.

- **Definición del endpoint:** La ruta `/process_message` recibe solicitudes POST con un JSON que contiene el mensaje del usuario (`message`), invoca la función `generate()` y devuelve la respuesta como JSON al bot Einstein.
- **Ejecución del servidor:** El servidor Flask se inicia en el puerto 5000 en modo depuración, lo que permitió realizar pruebas locales y monitorear los logs durante el desarrollo.

### Ventajas y limitaciones de esta aproximación

La conexión directa con Gemini permitió validar de manera rápida la capacidad del modelo para responder preguntas complejas en lenguaje natural y sirvió como prueba de concepto para evaluar su potencial. Sin embargo, se identificaron varias limitaciones:

- **Seguridad y privacidad de datos:** el envío de información institucional a una API externa implicaba riesgos en cuanto a confidencialidad.
- **Falta de integración nativa:** esta solución funcionaba de forma aislada, sin conexión directa con el CRM para registrar leads o casos PQRS.
- **Mantenimiento adicional:** requería un servidor intermedio en Flask para gestionar las solicitudes, lo que aumentaba la complejidad de la infraestructura.

Por estas razones, aunque fue útil como prueba inicial, esta aproximación no fue seleccionada como solución definitiva. No obstante, permitió identificar la viabilidad de incorporar capacidades generativas al flujo conversacional del bot institucional.

### 5.1.2. Uso de Salesforce Agentforce

La segunda alternativa evaluada para ampliar las capacidades del bot institucional fue el uso de **Salesforce Agentforce**, una herramienta nativa recientemente incorporada en el ecosistema Salesforce para la creación de agentes conversacionales basados en inteligencia artificial generativa.

Desde el inicio, Agentforce se percibía como una alternativa altamente atractiva. Al ser una solución desarrollada por Salesforce, ofrecía una integración directa con el CRM institucional, lo que en teoría permitiría optimizar procesos clave como la creación automática de *leads*, la generación de casos PQRS y la consulta de información institucional almacenada en el propio entorno Salesforce. Además, al estar completamente integrado dentro del ecosistema, prometía reducir la necesidad de infraestructura adicional, simplificando tanto la gestión como el mantenimiento de la solución.

#### Acceso al entorno de pruebas

Dado que Agentforce es una solución relativamente nueva y no estaba disponible de manera estándar en el sandbox institucional, fue necesario solicitar a Salesforce un acceso temporal a un entorno de desarrollo externo provisto exclusivamente para pruebas. Este entorno permitió evaluar las funcionalidades principales de la herramienta y comprobar cómo se comportaba en escenarios reales de uso, aunque con limitaciones en cuanto a espacio y permisos de configuración.

#### Pruebas realizadas y resultados obtenidos

Durante las pruebas iniciales se cargaron documentos institucionales en formato PDF y HTML, tales como reglamentos, preguntas frecuentes y manuales académicos,

para que el modelo generativo pudiera utilizarlos como base de conocimiento. Se configuraron también flujos simples para la detección de consultas relacionadas con programas académicos, procesos de admisión y solicitudes frecuentes.

En general, las respuestas generadas por Agentforce resultaron coherentes y bien redactadas cuando lograba encontrar coincidencias dentro de las fuentes cargadas, como se puede ver en la figura 5.1, donde se le pregunta al bot en lenguaje natural por información sobre la carrera de Diseño de Comunicación Visual ofrecida por la universidad y fue capaz de responder a la pregunta de manera clara y precisa.

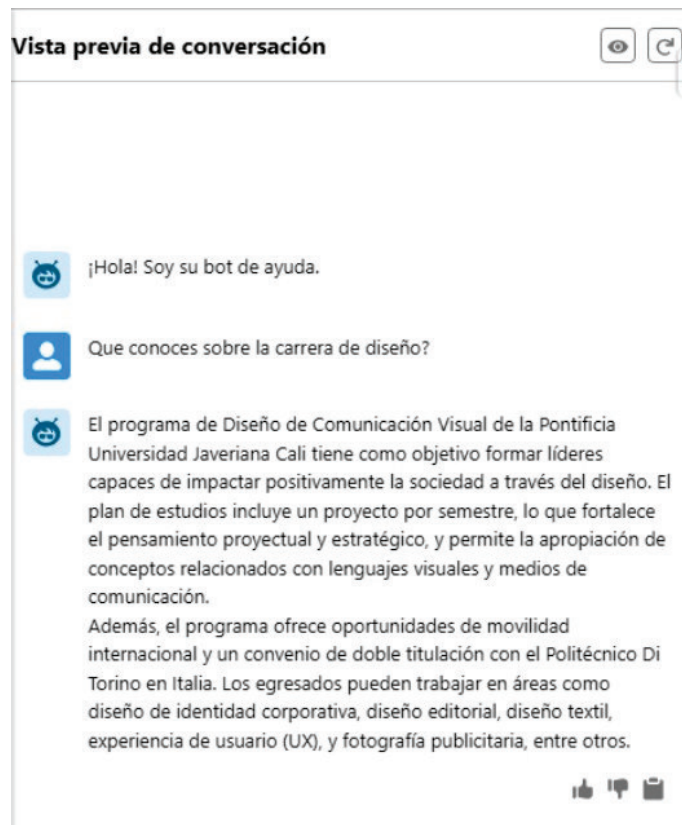


Figura 5.1: Respuesta acertiva del bot de Agentforce

Sin embargo, a medida que se realizaron más pruebas, surgieron varias limitaciones importantes:

- Acceso restringido y costos elevados: Al ser una herramienta emergente dentro del ecosistema Salesforce, el uso de Einstein Bots no está incluido en las licencias estándar. Para habilitarlos, es necesario adquirir la licencia Digital Engagement, que tiene un costo aproximado de 75 dólares por usuario/mes [21], además de que se incluyen solo 25 conversaciones mensuales por usuario. Si se requiere ampliar ese límite, el precio es de aproximadamente 2 dólares por cada conversación adicional [22]. Esto eleva significativamente el costo de implementación, lo que dificulta su adopción en proyectos piloto o pruebas de concepto en entornos institucionales con recursos limitados.

- **Restricciones en los tipos de fuentes:** Agentforce únicamente permite cargar fuentes en formato PDF, TXT y HTML. Esto limita considerablemente el tipo de información que el agente puede consultar, dejando por fuera datos dinámicos como APIs, bases de datos institucionales o transcripciones generadas automáticamente (como las solicitadas en los casos de uso).
- **Uso inconsistente de los documentos cargados:** Aunque los archivos eran aceptados y procesados, se observó que en varias ocasiones el modelo no los utilizaba correctamente para construir las respuestas. Esto generaba resultados incompletos o respuestas genéricas, especialmente cuando se requería información muy específica. Un caso de ello, fue que aunque se cargó la información sobre la maestría en ingeniería de software, al preguntarle por información sobre este programa, el bot no era capaz de responder; esto se puede evidenciar en la figura 5.2.

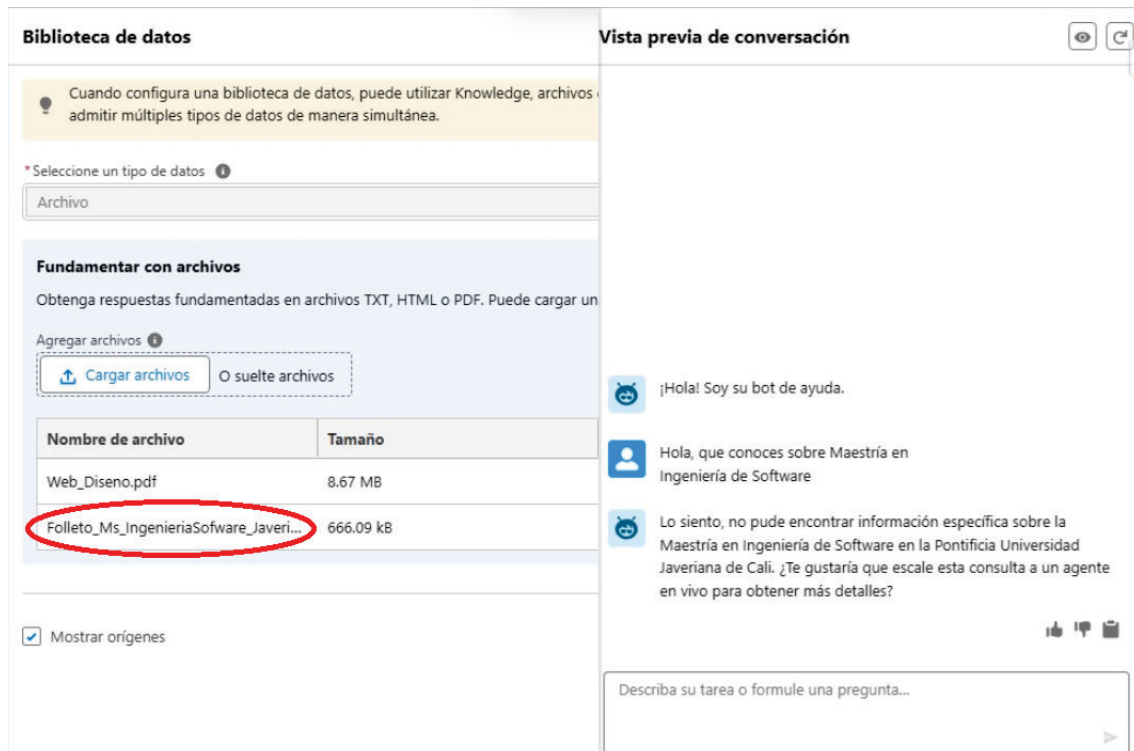


Figura 5.2: Inconsistencia en los documentos cargados a Agentforce

- **Capacidad de almacenamiento reducida:** El entorno de pruebas solo permitía almacenar una cantidad limitada de documentos, lo que restringía la cobertura temática y la actualización del conocimiento disponible para el agente.
- **Tiempos de respuesta elevados:** Si bien las respuestas obtenidas eran de buena calidad en términos de lenguaje y estructura, el tiempo que tardaba en generarlas era excesivo, afectando significativamente la experiencia de usuario. Para consultas simples podían tardar varios segundos, y para preguntas más

complejas los tiempos se volvían imprácticos, llegando a necesitar incluso más de 10 segundos para la obtención de la respuesta final, lo cual no es ideal para un entorno de atención automatizada.

### Conclusiones de esta aproximación

A pesar de sus ventajas potenciales, como la integración nativa con el CRM y la posibilidad de gestionar automáticamente procesos internos como leads y casos PQRS, las limitaciones técnicas y de acceso hicieron que esta alternativa no resultara viable para el contexto del proyecto. La dependencia de licencias costosas (2 USD por conversación aproximadamente), la falta de flexibilidad para incorporar diversas fuentes de datos y los problemas de rendimiento detectados durante las pruebas llevaron a descartar Agentforce como solución definitiva.

No obstante, esta exploración permitió identificar de forma temprana los alcances y restricciones de la herramienta, dejando abierta la posibilidad de evaluarla nuevamente en el futuro cuando su madurez tecnológica y condiciones de licenciamiento sean más favorables para entornos institucionales como el de la universidad.

### 5.1.3. Solución definitiva mediante n8n

Tras evaluar las alternativas iniciales, se optó por implementar la solución definitiva mediante **n8n**, una plataforma de automatización de flujos de trabajo que permitió integrar de manera controlada el bot Einstein, el modelo generativo Gemini, la base de datos y el CRM institucional. Esta herramienta resultó ser la opción más flexible y escalable, ya que no dependía de licencias costosas como las de Agentforce, permitía una personalización completa de los procesos y facilitaba la conexión entre servicios internos y externos.

### Uso de Qdrant como base de datos vectorial

Para implementar el enfoque RAG fue necesario contar con una base de datos especializada capaz de almacenar y buscar **vectores semánticos**. A diferencia de una base de datos tradicional que almacena texto exacto, una base vectorial convierte los documentos en representaciones numéricas (embeddings) que capturan el significado del contenido. Esto permite realizar búsquedas por similitud, de modo que el sistema pueda encontrar información relevante aunque no coincida literalmente con las palabras del usuario.

En este proyecto se utilizó **Qdrant**, una base vectorial optimizada para consultas de alta precisión y bajo tiempo de respuesta. Allí se almacenaron los embeddings generados a partir de documentos institucionales, preguntas frecuentes, reglamentos

y transcripciones de videos, lo que permitió que el bot pudiera recuperar fragmentos de texto relevantes como contexto para la generación de respuestas.

### Estructura general del flujo en n8n

El flujo principal implementado en n8n se diseñó para procesar de manera dinámica cada mensaje recibido desde el bot Einstein. La secuencia general puede dividirse en las siguientes etapas:

1. **Recepción del mensaje:** n8n recibe la consulta del usuario enviada desde el bot Einstein. Esta solicitud llega en formato JSON, incluyendo el mensaje y metadatos básicos de la sesión.
2. **Identificación de intención:** se utiliza el modelo generativo Gemini para analizar el contenido del mensaje y clasificarlo según su intención principal. Las categorías definidas fueron: pregunta frecuente, interés académico (lead), queja o reclamo (PQRS), consulta compleja que requiere generación, o mensaje que debe ser remitido a un agente humano.

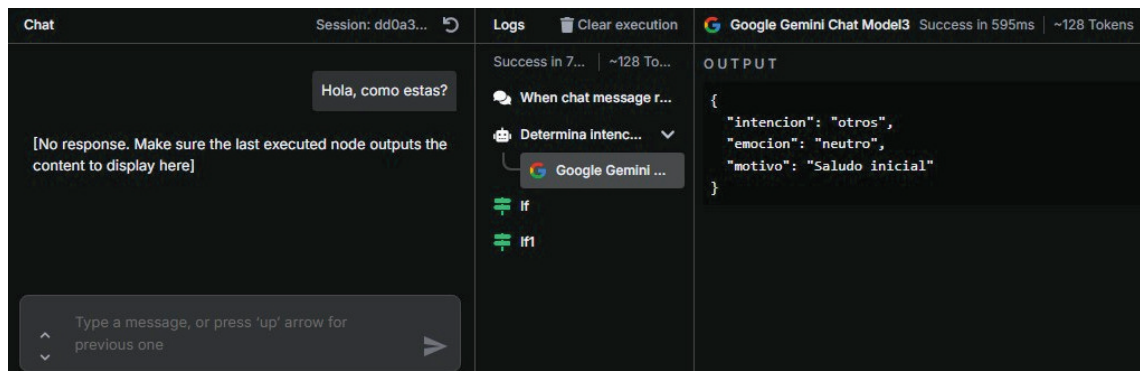


Figura 5.3: Clasificación automática de intención, emoción y motivo del mensaje mediante el modelo Gemini.

3. **Procesamiento según la intención:** en función de la clasificación obtenida, el flujo sigue diferentes rutas:
  - Si es una **pregunta frecuente**, se intenta responder con información predefinida.
  - Si es una **consulta compleja**, se genera un embedding del mensaje y se realiza una búsqueda por similitud en Qdrant. Los fragmentos recuperados se envían como contexto a Gemini, que construye una respuesta contextualizada.
  - Si corresponde a un **interés académico**, n8n captura los datos básicos del usuario (nombre, correo, programa de interés) y crea automáticamente

un lead en Salesforce, utilizando para ello los componentes nativos de integración con el CRM que ofrece n8n.

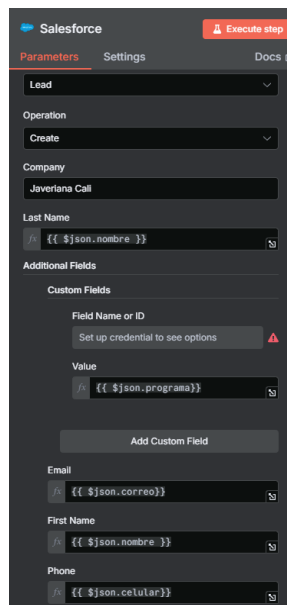


Figura 5.4: Configuración del nodo nativo de Salesforce en n8n para la creación automática de leads a partir de datos capturados en la conversación

- Si el mensaje es una **PQRS**, el flujo solicita datos adicionales y genera automáticamente un caso en Salesforce, también mediante los nodos específicos que n8n provee para esta plataforma.

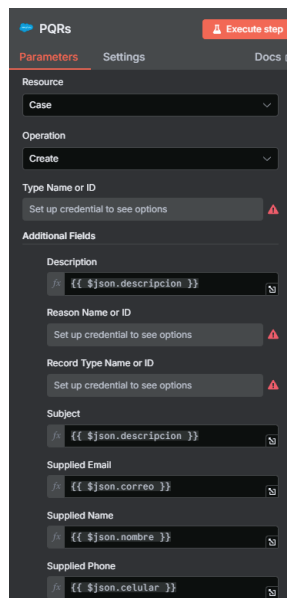


Figura 5.5: Configuración del nodo nativo de Salesforce en n8n para la creación automática de casos a partir de datos capturados en la conversación

- Si ninguna ruta puede resolver el mensaje, se activa el **escalamiento a**

un agente humano, enviando el historial de la conversación para dar continuidad a la atención.

4. **Envío de la respuesta:** n8n retorna el mensaje generado al bot Einstein a través de un endpoint configurado en Apex, el cual recibe la respuesta procesada y la entrega al usuario final dentro del entorno conversacional de Salesforce.

### Carga y actualización de contenido

Para mantener la base vectorial actualizada se implementó un único flujo complementario en n8n. Este detecta el tipo de archivo y lo procesa en consecuencia:

- Si es **MP4/MP3**, aplica transcripción *speech-to-text* para generar texto.
- Si es **PDF/DOCX**, lo procesa directamente.
- Luego, divide el contenido en fragmentos, genera sus embeddings y los carga automáticamente en Qdrant.

Adicionalmente, se desarrolló una interfaz gráfica en React que permite a los administradores cargar nuevos archivos de manera sencilla, sin necesidad de interactuar directamente con el script de carga.



Figura 5.6: Vista de la interfaz gráfica para carga de archivos.

### Implementación y referencia al repositorio

La configuración completa de estos flujos en n8n, junto con el script y el desarrollo de la interfaz gráfica de carga de archivos, se encuentra disponible en el repositorio público del proyecto [23]. Este repositorio documenta paso a paso la implementación, permitiendo replicar o adaptar la solución en otros entornos.

### Ventajas de la solución definitiva

Esta arquitectura basada en n8n permitió:

- Integrar múltiples componentes (bot Einstein, Gemini, Qdrant y Salesforce) sin depender de licencias adicionales.
- Mantener trazabilidad de cada interacción, asegurando control sobre los datos enviados y recibidos.
- Escalar fácilmente el sistema agregando nuevas fuentes de conocimiento o ajustando las rutas del flujo según necesidades futuras.

Gracias a esta flexibilidad, n8n se consolidó como la herramienta más adecuada para cubrir los requerimientos institucionales, permitiendo un equilibrio entre personalización, rendimiento y facilidad de mantenimiento.

## 5.2. Arquitectura de la solución final

La solución final se diseñó bajo el enfoque de arquitectura RAG, que permite combinar la capacidad generativa de un modelo de lenguaje con la recuperación dinámica de información institucional. Esta integración garantiza respuestas precisas, actualizadas y contextualizadas con las fuentes oficiales de la universidad, evitando depender únicamente del conocimiento preentrenado del modelo.

A diferencia de un bot tradicional basado en flujos rígidos, esta arquitectura incorpora módulos inteligentes que detectan la intención del usuario, consultan bases de conocimiento vectoriales y automatizan procesos administrativos cuando es necesario. En conjunto, el flujo general está compuesto por cuatro bloques principales:

1. **Detección de intención y análisis de tono**, que clasifica las consultas y permite priorizar reclamos o quejas.
2. **Recuperación semántica**, mediante la búsqueda de información relevante en Qdrant cuando la consulta requiere contexto adicional.

3. **Generación contextualizada de respuestas**, usando Gemini con los fragmentos recuperados como base.
4. **Integración CRM**, que registra automáticamente leads o casos en Salesforce cuando la interacción implica seguimiento administrativo.

La figura 5.7 muestra una vista global de este flujo, donde cada componente interactúa para ofrecer una experiencia conversacional más robusta y adaptativa.

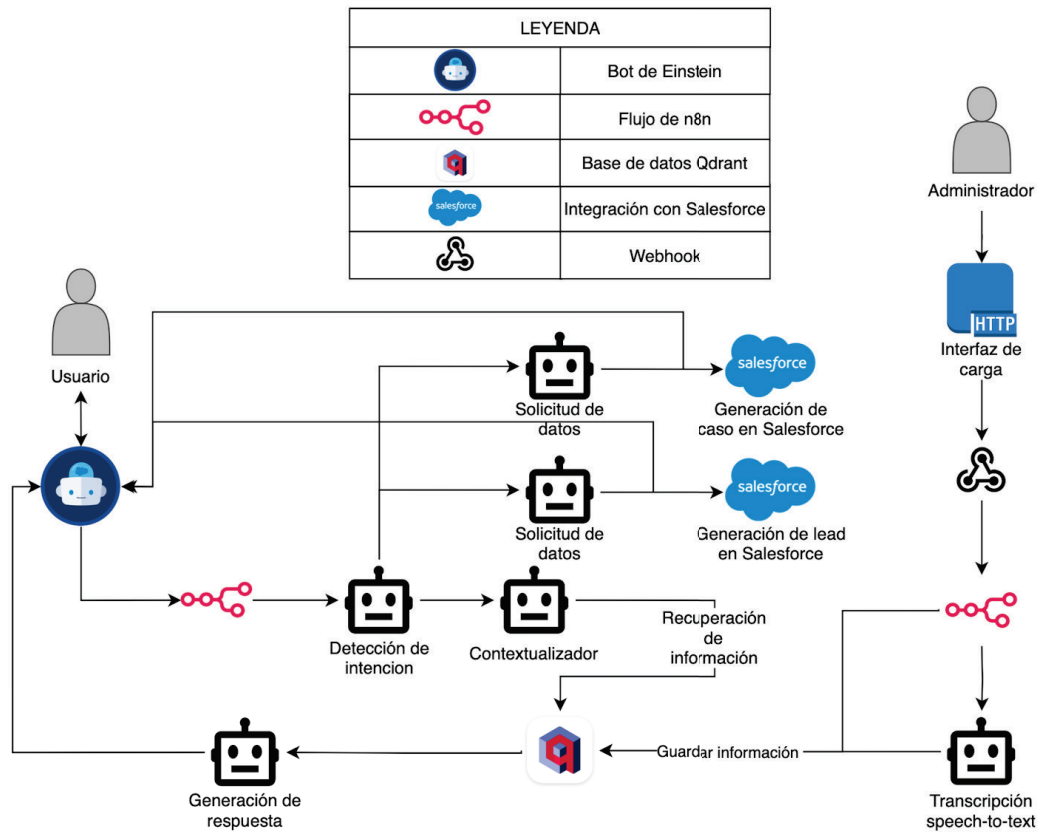


Figura 5.7: Diagrama de flujo general con enfoque RAG e integración CRM

### 5.2.1. Análisis de intención

Cada mensaje enviado por el usuario se recibe en n8n a través de un webhook y se procesa en una primera capa de análisis. Este módulo utiliza Gemini para identificar:

- **Tipo de intención:** pregunta frecuente, interés académico, PQRS o consulta compleja.

- **Sentimiento asociado:** neutral, positivo, negativo o urgente.

Esta clasificación permite enrutar el mensaje hacia el módulo adecuado y priorizar automáticamente casos sensibles, como quejas o reclamos. Por ejemplo, si el mensaje expresa frustración, el flujo puede escalarlo de inmediato a un agente humano o generar un caso en Salesforce.

### 5.2.2. Procesamiento según tipo de intención

En función de la intención detectada, el flujo toma distintas rutas:

- Si se trata de una **pregunta frecuente**, se intenta responder directamente mediante el agente configurado del bot anterior para generar respuestas estándar.
- Si la pregunta excede las capacidades del flujo tradicional o se identifica como **compleja o ambigua**, el mensaje es contextualizado y redirigido hacia el componente RAG.
- Si el usuario manifiesta un **interés en un programa académico**, se activa un módulo de extracción de datos (nombre, correo, celular, programa) y, si la información es suficiente, se genera automáticamente un *lead* en Salesforce, aprovechando los conectores nativos de n8n.

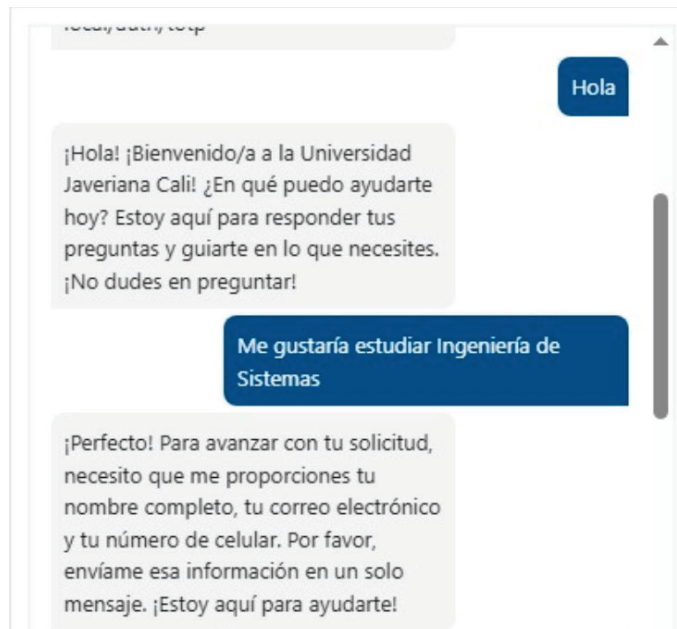


Figura 5.8: Solicitud de datos adicionales para generar el lead en Salesforce

- Si el mensaje corresponde a una **queja o PQRS**, el sistema solicita datos complementarios y, al completarse, genera un caso directamente en Salesforce.

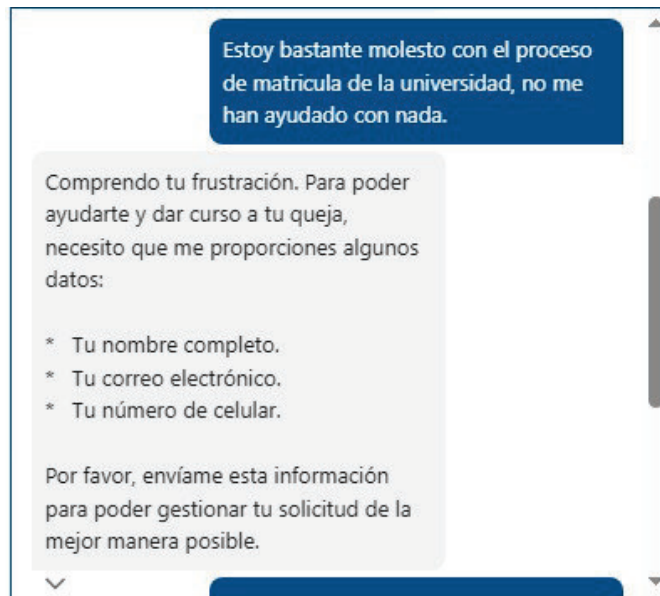


Figura 5.9: Solicitud de datos adicionales para generar el caso en Salesforce

Este ruteo dinámico optimiza recursos: consultas simples no saturan el motor generativo, mientras que las complejas reciben un tratamiento más profundo.

### 5.2.3. Componente RAG: recuperación y generación

Cuando una consulta no puede resolverse mediante respuestas predefinidas, el sistema recurre al enfoque RAG. El flujo es el siguiente:

1. **Normalización y embedding de la consulta:** Se limpia el texto y se genera una representación vectorial del mensaje usando Gemini.
2. **Búsqueda semántica en Qdrant:** Este vector se utiliza para realizar una búsqueda por similitud dentro de la base vectorial construida con Qdrant. Esta base contiene documentos institucionales previamente cargados y divididos por tokens, incluyendo preguntas frecuentes, reglamentos, calendarios, y transcripciones de video.
3. **Construcción de contexto:** Se recuperan los fragmentos de texto más relevantes y se entregan como contexto al modelo generativo (Gemini).
4. **Generación de respuesta:** Gemini recibe este contexto y produce una respuesta coherente, fundamentada en los documentos consultados.

5. **Entrega al usuario:** La respuesta generada se retorna al bot Einstein mediante un endpoint en Apex, mostrándose en el mismo canal de interacción original.

### 5.2.4. Memoria conversacional y enriquecimiento

Para mantener la coherencia en conversaciones prolongadas, se implementó un módulo de memoria de contexto. Este almacena interacciones previas dentro de una misma sesión y permite que el bot recuerde datos ya proporcionados por el usuario, evitando preguntas repetitivas. Por ejemplo:

- Si el usuario ya indicó su nombre o programa de interés en turnos anteriores, el flujo puede reutilizar esa información.
- Si una nueva consulta depende de un mensaje previo (“¿Y cuáles son las fechas para ese programa?”), el bot recupera el contexto de la interacción anterior para responder correctamente.

### 5.2.5. Ventajas de la arquitectura implementada

La integración de RAG con automatización conversacional y CRM proporciona varias ventajas clave:

- Aprovechar modelos generativos manteniendo control sobre las fuentes de información utilizadas, las respuestas se basan en fuentes oficiales, reduciendo errores y contradicciones.
- Escalar el sistema de información incorporando nuevas fuentes documentales sin necesidad de reentrenar el modelo.
- Leads y PQRS se registran automáticamente en Salesforce, integrando la atención conversacional con la gestión institucional.
- Se pueden agregar nuevas fuentes documentales, nodos en n8n o procesos CRM sin rediseñar toda la solución.
- Ahora es posible para los usuarios interactuar con el bot utilizando lenguaje natural, sin necesidad de seleccionar opciones rígidas o escribir palabras clave exactas.

Con esta arquitectura, el bot pasa de ser una herramienta estática basada en flujos predefinidos, a un asistente inteligente capaz de responder a necesidades más amplias, adaptarse a la intención del usuario y generar valor mediante su conexión con los procesos internos de la universidad.

### 5.3. Herramientas utilizadas

Para la implementación de la solución se emplearon diferentes herramientas que permitieron orquestar los flujos de automatización, gestionar el conocimiento institucional y conectar las distintas capas del sistema conversacional. A continuación, se describen las principales:

- **n8n:** plataforma de automatización de flujos que actuó como núcleo de la solución. Permitió orquestar la interacción entre el bot Einstein, el modelo generativo de Gemini y las demás fuentes de información. Además, facilitó la integración con Salesforce CRM para la creación automática de leads y casos PQRS.
- **Qdrant database:** base de datos vectorial utilizada para almacenar las representaciones semánticas (*embeddings*) de documentos institucionales. Gracias a Qdrant fue posible realizar búsquedas por similitud, recuperando fragmentos relevantes que luego se usaban como contexto en el enfoque RAG.
- **Salesforce CRM:** sistema de gestión de relaciones institucionales en el que se registran leads, casos y procesos relacionados con aspirantes, estudiantes y usuarios externos. Se integró al flujo del bot para automatizar la creación de registros de interés académico y casos de atención (PQRS).
- **Einstein Bot Builder:** entorno de Salesforce para el diseño y despliegue del bot conversacional. En esta herramienta se configuraron los flujos iniciales, las intenciones más comunes y las condiciones para derivar mensajes a los módulos externos de análisis.
- **Modelo generativo (Gemini):** se eligió Gemini 2.0 Flash Lite por su rapidez, coherencia en español, fácil integración mediante API con n8n y compatibilidad con el enfoque RAG usando Qdrant. Su alto límite de tokens sin coste lo hizo ideal para un proyecto piloto, ofreciendo un equilibrio entre velocidad y calidad sin generar gastos. A futuro, se considera viable probar modelos más avanzados en proyectos con mayor presupuesto para ampliar las capacidades y cobertura del bot.
- **Herramientas de transcripción de video:** soluciones externas que permitieron extraer texto de contenido audiovisual institucional. En este proyecto se utilizó la API de *Lemonfox*, que ofreció un procesamiento rápido y preciso de archivos en formatos MP4 y MP3. Las transcripciones obtenidas se integraron a la base documental para ampliar la cobertura temática del bot.
- **Docker:** se utilizó para desplegar y ejecutar los flujos de trabajo de n8n en un entorno local, garantizando portabilidad, facilidad de configuración y aislamiento del entorno de desarrollo. Esto permitió mantener una infraestructura controlada y replicable durante todo el proceso de implementación.

## 5.4. Configuración del bot y funcionalidades implementadas

Con base en los requerimientos definidos por los actores clave y las limitaciones detectadas en el bot original, se configuraron nuevas funcionalidades orientadas a mejorar la experiencia del usuario y la conexión con los procesos internos de la universidad. Estas funcionalidades se integraron al flujo conversacional a través de n8n y el bot Einstein.

- **Reorganización de preguntas frecuentes (FAQ):** se actualizaron y limpiaron los flujos preexistentes para mejorar su precisión y facilitar la cobertura de consultas comunes.
- **Análisis de intención:** se incorporó un módulo basado en IA generativa para clasificar las consultas según su objetivo (pregunta frecuente, interés académico, queja/PQRS, solicitud compleja). Esta clasificación determinaba la ruta que seguiría cada mensaje dentro del flujo.
- **Activación de IA generativa por condición:** cuando una consulta no coincidía con los flujos preconfigurados, se activaba el componente RAG para buscar información en Qdrant y, de ser necesario, generar una respuesta contextualizada mediante Gemini.
- **Respuestas desde transcripciones de video:** el bot fue habilitado para acceder a las transcripciones de videos institucionales, permitiéndole responder preguntas basadas en contenido audiovisual previamente indexado.
- **Creación automática de leads:** cuando el usuario manifestaba interés en programas académicos u ofertas institucionales, el bot capturaba datos básicos y registraba automáticamente un lead en Salesforce, clasificándolo por nivel académico o facultad.
- **Escalamiento a agentes humanos:** se implementó un flujo que permitía derivar conversaciones a personal de soporte cuando el bot no podía resolver la consulta, incluyendo el historial de mensajes para dar continuidad a la atención.
- **Creación de casos PQRS:** para peticiones, quejas o reclamos que requerían seguimiento formal, el bot generaba automáticamente un caso en Salesforce, asociándolo al usuario y su mensaje original.

La implementación descrita en este capítulo permitió transformar el bot institucional en un sistema más flexible y capaz de interactuar con distintas fuentes de información, combinando la recuperación semántica con la generación de respuestas en lenguaje natural. Además, la integración con Salesforce CRM facilitó la automatización de procesos clave como la creación de leads y casos PQRS.

En el siguiente capítulo se presentan los resultados de la fase de validación funcional, evaluando la efectividad del sistema en escenarios reales de uso y analizando el impacto de las funcionalidades desarrolladas en la experiencia de los usuarios.

# Capítulo 6

## Validación del Sistema y Resultados

### 6.1. Plan de Pruebas

El propósito del plan de pruebas fue asegurar que el sistema conversacional basado en arquitectura RAG funcionara de manera correcta, validando tanto su integración técnica como su capacidad para mejorar la atención institucional. Se buscó garantizar que el flujo entre bot Einstein, n8n, Qdrant, Gemini y Salesforce operara de forma confiable, respondiendo consultas con precisión y automatizando procesos administrativos sin errores.

Estas pruebas permitieron identificar posibles puntos de mejora y ajustar la solución antes de su posible despliegue definitivo.

### 6.2. Alcance de las Pruebas

El alcance de las pruebas incluyó la evaluación integral del sistema en condiciones controladas, analizando:

- **Clasificación de intención y análisis de tono emocional**, verificando que las consultas fueran correctamente categorizadas.
- **Recuperación semántica** desde Qdrant para preguntas complejas.
- **Calidad de las respuestas generadas por Gemini**, considerando coherencia, claridad y relevancia.
- **Automatización de procesos CRM**, validando la creación de leads y PQRS en Salesforce mediante n8n.

- **Escalamiento a agente humano**, comprobando que los casos no resueltos se transfirieran con el historial conversacional.
- **Experiencia del usuario**, verificando la interacción en lenguaje natural, sin depender de menús rígidos ni palabras clave exactas.

### 6.3. Validación de Requisitos

Se validaron los requisitos funcionales y no funcionales definidos en la sección 4.4, asegurando que la solución cumpliera con los objetivos planteados.

#### 6.3.1. Requisitos Funcionales

Requisito	Nombre	Criterio de Éxito	Validación
RF1	Detección de intención	Al menos el 90% de las consultas fueron clasificadas en la categoría correcta.	Se probaron mensajes de FAQ, consultas complejas, PQRS e interés académico.
RF2	Recuperación semántica	Qdrant devolvió fragmentos relevantes para consultas no previstas en los flujos tradicionales.	Se consultaron reglamentos, calendarios y documentos cargados previamente.
RF3	Generación contextualizada	Gemini produjo respuestas claras, precisas y fundamentadas en los fragmentos recuperados.	Se comparó la respuesta generada con la información oficial.
RF4	Registro en CRM	Leads y PQRS se crearon automáticamente en Salesforce con los datos capturados.	Se verificó el registro correcto mediante los nodos nativos de n8n.
RF5	Escalamiento a agente humano	Las consultas no resueltas fueron transferidas con historial conversacional.	Se simuló una interacción ambigua para activar el escalamiento.

Cuadro 6.1: Validación de Requisitos Funcionales del sistema RAG

### 6.3.2. Requisitos No Funcionales

Requisito	Nombre	Criterio de Éxito	Validación
RNF1	Tiempo de respuesta	$\leq 3$ segundos para preguntas frecuentes y $\leq 6$ segundos para consultas complejas.	Se midieron tiempos en interacciones reales.
RNF2	Escalabilidad	Se podían añadir nuevos documentos sin modificar el modelo.	Qdrant cargó nuevos archivos procesados automáticamente por $n8n$ .
RNF3	Trazabilidad	Cada interacción conservó historial y registros auditables.	Las conversaciones se almacenaron en $n8n$ y Salesforce.
RNF4	Seguridad	Se garantizó que no se expusieran datos sensibles.	Se usaron conexiones HTTPS y tokens de autenticación.
RNF5	Disponibilidad 24/7 y concurrencia	Disponibilidad continua y manejo de múltiples usuarios simultáneos sin degradación del rendimiento.	Se realizaron pruebas de carga simulando varios usuarios concurrentes, verificando que los tiempos de respuesta se mantuvieran dentro de los límites definidos y sin interrupciones en el servicio.
RNF6	Accesibilidad y claridad conversacional	Interfaz clara e inclusiva que facilite la comprensión y uso para todo tipo de usuarios.	Se evaluó la experiencia conversacional con un grupo de usuarios de distintos perfiles.

Cuadro 6.2: Validación de Requisitos No Funcionales del sistema RAG

## 6.4. Escenarios de Prueba

Se diseñaron casos de prueba representativos, que cubrieron los principales flujos conversacionales:

- **FAQ:** Preguntas simples sobre matrículas, becas y programas.

- **Consulta compleja:** Requirió recuperar documentos de reglamentos o calendarios.
- **Interés académico:** Simuló la generación automática de un lead.
- **PQRS:** Creó un caso en Salesforce tras solicitar datos adicionales.
- **Escalamiento:** Mensajes ambiguos o sin intención clara para derivar a un asesor.

Cada escenario documentó: entrada del usuario, ruta esperada, respuesta generada y estado final (aprobado/ajuste requerido).

## 6.5. Métricas de Evaluación

Además de validar el cumplimiento funcional, se definieron métricas para evaluar la **calidad de las respuestas y la eficiencia del sistema:**

- **Precisión de clasificación:** porcentaje de mensajes correctamente categorizados.
- **Tasa de recuperación relevante:** porcentaje de fragmentos útiles en la búsqueda semántica.
- **Tiempo promedio de respuesta:** medido por tipo de consulta.
- **Tasa de éxito en CRM:** porcentaje de leads y PQRS creados sin errores.
- **Métrica de calificación de calidad de respuesta:** evaluada en una escala de 1 a 5 según:
  - **Claridad:** ¿Fue fácil de entender?
  - **Relevancia:** ¿Respondió a lo que se preguntó?
  - **Utilidad:** ¿Entregó información que realmente ayudó al usuario?

La siguiente tabla resumió cómo se evaluó cada respuesta:

Criterio	Descripción
Claridad	Lenguaje comprensible, sin ambigüedades ni términos innecesariamente técnicos.
Relevancia	La respuesta se ajustó al contexto de la pregunta realizada.
Utilidad	Ofreció datos concretos o próximos pasos claros para el usuario.

Cuadro 6.3: Criterios de evaluación de calidad de respuesta

Cada interacción recibió una calificación promedio sobre 5 puntos, permitiendo medir la mejora frente al bot original.

### 6.6. Validación de Integración

Finalmente, se ejecutaron pruebas de extremo a extremo para validar la comunicación entre componentes:

1. El bot Einstein recibió la consulta y la envió a n8n mediante Apex.
2. n8n analizó la intención, consultó Qdrant y generó respuesta con Gemini.
3. Si correspondía, n8n creó leads o PQRS en Salesforce usando sus nodos nativos.
4. La respuesta final retornó al bot, conservando trazabilidad y contexto.

Se verificó que no existieran errores de autenticación, tiempos muertos o duplicidad de registros en Salesforce.

### 6.7. Resultados y Análisis

### 6.8. Resultados y Análisis

#### 6.8.1. Resultados de las pruebas

Antes de presentar los resultados obtenidos en las pruebas del sistema conversacional, es importante facilitar la lectura de las tablas mediante abreviaciones para las métricas evaluadas. A continuación, se emplearán las siguientes abreviaciones:

- **PC:** Precisión de clasificación (%), que mide el porcentaje de mensajes correctamente categorizados.
- **TR:** Tiempo de respuesta (segundos), correspondiente al promedio de procesamiento por consulta.
- **TRR:** Tasa de recuperación relevante (%), que indica qué tan útiles fueron los fragmentos recuperados desde Qdrant.
- **TEC:** Tasa de éxito en CRM (%), que evalúa el correcto registro de leads o PQRS en Salesforce.

- **C**: Claridad de la respuesta, calificada en una escala de 1 a 5.
- **R**: Relevancia de la respuesta, calificada en una escala de 1 a 5.
- **U**: Utilidad de la respuesta, calificada en una escala de 1 a 5.

Con esta notación, se presentan las tablas que consolidan los resultados para cada categoría de prueba: preguntas frecuentes (FAQ), consultas complejas, generación de leads, PQRS y casos de escalamiento. Cada tabla resume los datos recogidos durante la validación por varios usuarios, permitiendo un análisis comparativo del desempeño del sistema en distintos escenarios de interacción. Adicionalmente, se incluye la evaluación de los tiempos de respuesta bajo concurrencia para verificar el cumplimiento del requisito RNF5.

### 6.8.2. Usuario del equipo desarrollador

#### Pruebas: preguntas frecuentes (FAQ)

Caso FAQ	PC	TR	C	R	U	Síntesis
Solicitud de información de carreras	100	6.33	5	5	5	Se le preguntó sobre la carrera de enfermería y retornó información importante como los perfiles, la duración y los costes.
Becas de pregrado	100	5.3	5	5	5	El agente listó las becas existentes y al preguntar por una específica brindó los detalles necesarios.
Programas de pregrado	100	5.7	5	5	5	El agente listó los programas ofrecidos por la universidad y además fue capaz de dividirlos cuando se le preguntó por facultades y departamentos

**Pruebas: consultas complejas**

Consulta Compleja	PC	TRR	TR	C	R	U	Síntesis
Proceso de inscripción	100	100	6.6	4	5	5	El agente entregó los pasos detallados para completar la inscripción, incluyendo llenar un formulario. Aunque el enlace no apareció en la primera respuesta, fue proporcionado en un mensaje inmediato posterior, garantizando una respuesta completa y útil.
Calendario administrativo	100	100	5.13	3	4	4	El agente logró dar el enlace correcto al calendario administrativo, pero solo después de un segundo intento. Inicialmente interpretó la solicitud de forma ambigua como un calendario genérico y respondió que no tenía acceso, corrigiendo luego y entregando la información adecuada.

**Pruebas: generación de leads**

Caso Lead	PC	TR	TEC	C	R	U	Síntesis
Interés en Ingeniería de Sistemas	100	10.7	100	5	5	5	El lead se generó correctamente en el sandbox con la información brindada.

**Pruebas: generación de casos PQRS**

Caso PQRS	PC	TR	TEC	C	R	U	Síntesis
Queja sobre atención administrativa	100	9.3	100	5	5	5	El caso se generó correctamente en el sanbox registrando la queja adecuadamente y los datos del usuario.
Reclamo sobre inscripción	100	10.4	100	5	5	5	El reclamo se generó correctamente en el sanbox registrando la queja adecuadamente y los datos del usuario.

**Pruebas: escalamiento a agente humano**

Caso Escalamiento	PC	TR	C	R	U	Síntesis
Petición de escalamiento	100	1.75	5	5	5	El agente detectó correctamente que la solicitud requería intervención humana y activó el flujo de escalamiento sin errores. Además, transfirió el historial de conversación para que el asesor pudiera continuar con el contexto completo, asegurando una experiencia fluida para el usuario.

### 6.8.3. Usuario perteneciente a la comunidad estudiantil

#### Pruebas: preguntas frecuentes (FAQ)

Caso FAQ	PC	TR	C	R	U	Síntesis
Solicitud de horarios de clases	100	5.4	5	5	5	El bot entregó los horarios completos de los cursos consultados y permitió filtrarlos por semestre.
Información sobre bibliotecas	100	6.1	5	5	5	Respondió con claridad sobre ubicación, horarios y servicios, incluyendo préstamo de portátiles.
Beneficios de bienestar universitario	100	5.9	5	5	5	Listó los programas de bienestar disponibles y ofreció enlaces para inscribirse.

**Pruebas: consultas complejas**

Consulta Compleja	PC	TRR	TR	C	R	U	Síntesis
Solicitud de reglamento de prácticas	100	100	7.2	5	5	5	El bot entregó el reglamento en PDF y resumió los puntos clave en la respuesta inicial.
Calendario de grados	100	100	6.5	5	5	5	Indicó las fechas de grados del semestre, con enlaces a la sección oficial de la universidad.

**Pruebas: generación de leads**

Caso Lead	PC	TR	TEC	C	R	U	Síntesis
Interés en Psicología	100	9.8	100	5	5	5	El lead se creó en Salesforce con todos los datos de contacto y el programa seleccionado.

**Pruebas: generación de casos PQRS**

<b>Caso PQRS</b>	<b>PC</b>	<b>TR</b>	<b>TEC</b>	<b>C</b>	<b>R</b>	<b>U</b>	<b>Síntesis</b>
Queja sobre servicio en cafetería	100	9.5	100	5	5	5	Registró el caso correctamente y entregó número de radicado para seguimiento.
Solicitud de certificado académico	100	10.2	100	5	5	5	El bot creó el caso de forma correcta y brindó instrucciones de seguimiento en el portal.

**Pruebas: escalamiento a agente humano**

<b>Caso Escalamiento</b>	<b>PC</b>	<b>TR</b>	<b>C</b>	<b>R</b>	<b>U</b>	<b>Síntesis</b>
Duda no contemplada sobre intercambios	100	2.1	5	5	5	El bot transfirió la conversación a un asesor humano y entregó el historial completo.

#### 6.8.4. Usuario del equipo técnico universitario

##### Pruebas: preguntas frecuentes (FAQ)

Caso FAQ	PC	TR	C	R	U	Síntesis
Solicitud de reglamento interno	100	7.2	5	5	5	El bot recuperó el reglamento y resaltó secciones importantes.
Procesos de inscripción en línea	90	8.4	4	3	3	Confundió inicialmente inscripción académica con admisión. Requirió segunda interacción para corregirse.
Información sobre convenios internacionales	85	8.1	3	3	3	La respuesta fue incompleta y no incluyó enlaces oficiales a las convocatorias de movilidad.

##### Pruebas: consultas complejas

Consulta Compleja	PC	TRR	TR	C	R	U	Síntesis
Pregunta sobre requisitos de posgrado	95	85	11.3	4	4	3	El bot recuperó información relevante sobre requisitos generales, pero no incluyó detalles más específicos como certificaciones de idioma.
Procedimiento para homologaciones	90	80	12.5	3	3	3	Ofreció una respuesta útil pero incompleta, ya que no mencionó toda la documentación necesaria ni los pasos exactos del proceso.

**Pruebas: generación de leads**

Caso Lead	PC	TR	TEC	C	R	U	Síntesis
Interés en Maestría en Ingeniería	95	12.7	100	5	5	5	El lead se creó de forma correcta en Salesforce con los datos principales.

**Pruebas: generación de casos PQRS**

Caso PQRS	PC	TR	TEC	C	R	U	Síntesis
Reclamo sobre notas en plataforma	90	13.4	100	4	4	4	El caso se registró sin errores y con trazabilidad, aunque se puede mejorar la explicación para el seguimiento del caso.
Queja sobre matrícula extraordinaria	85	14.1	95	4	2	3	El sistema creó el caso en Salesforce, pero el bot se limitó a registrar la frase del usuario (" <i>tengo un problema</i> ") sin solicitar más detalle, lo que redujo la claridad de la información almacenada.

**Pruebas: escalamiento a agente humano**

Caso Escalamiento	PC	TR	C	R	U	Síntesis
Pregunta crítica sobre política de datos	90	3.5	5	5	5	El bot activó el flujo de escalamiento correctamente.

### 6.8.5. Comparación de tiempos bajo concurrencia

Usuarios simultáneos	Tiempo promedio (PF)	Tiempo promedio (CC)	Errores detectados
1	5.8 s	7.9 s	Ninguno. Respuestas fluidas y sin pérdida de trazabilidad.
5	6.2 s	9.4 s	Se detectó un leve retraso en consultas complejas, pero dentro del rango aceptable.
10	7.1 s	12.3 s	Dos respuestas de consultas complejas llegaron incompletas, aunque el bot entregó mensajes de respaldo para continuar la interacción.

Cuadro 6.4: Comparación de tiempos de respuesta bajo diferentes niveles de concurrencia

### 6.8.6. Análisis de los resultados de las pruebas

Los resultados obtenidos en las distintas categorías de pruebas permiten identificar fortalezas y aspectos a mejorar en el sistema implementado. En primer lugar, se observó que en las consultas de tipo *FAQ* el desempeño fue sobresaliente: las respuestas fueron claras, completas y rápidas, lo que confirma que el modelo es adecuado para cubrir la mayoría de preguntas frecuentes de estudiantes, aspirantes y demás usuarios de la universidad.

En las consultas complejas, el comportamiento fue positivo pero con ciertos retos. Aunque el bot logró recuperar información útil y en muchos casos entregar enlaces oficiales, en algunas ocasiones la primera interpretación fue ambigua o incompleta, lo que obligó al usuario a reformular su consulta. Este hallazgo muestra la necesidad de seguir afinando los mecanismos de recuperación y la forma en que el bot solicita aclaraciones.

La generación de *leads* mostró un rendimiento muy sólido: los registros fueron creados en Salesforce de manera exitosa y con la información suficiente para que la institución pueda dar continuidad al proceso. Este resultado evidencia que la integración con el CRM es confiable y cumple con los objetivos de captar nuevos interesados.

En el caso de las PQRS, también se logró un desempeño satisfactorio, aunque se identificaron oportunidades de mejora. El sistema fue capaz de registrar los casos con trazabilidad en Salesforce, pero en algunos escenarios no profundizó en la solicitud del usuario y almacenó descripciones muy generales. Esto resalta la importancia de implementar validaciones adicionales que incentiven al bot a pedir más detalle cuando la consulta sea demasiado corta o ambigua.

En cuanto a los tiempos de respuesta, las métricas bajo concurrencia mostraron que el sistema mantiene un rendimiento estable con pocos usuarios simultáneos y que, aunque aparecen leves retrasos e incluso respuestas incompletas en escenarios de alta concurrencia, el bot fue capaz de mantener la interacción con mensajes de respaldo. Esto indica que el sistema es escalable y funcional, pero que podría optimizarse aún más con mejoras en la gestión de cargas elevadas.

En conjunto, estos resultados reflejan que el sistema alcanza de manera satisfactoria los objetivos planteados: responde adecuadamente en los escenarios más frecuentes, se integra de forma efectiva con Salesforce y ofrece una experiencia fluida a los usuarios, al tiempo que deja en evidencia los espacios de mejora que pueden guiar futuros trabajos.

# Capítulo 7

## Conclusiones

Este trabajo permitió abordar la mejora del bot web institucional, cumpliendo los objetivos planteados y, al mismo tiempo, generando aprendizajes significativos sobre la implementación de inteligencia artificial generativa en entornos académicos. En primer lugar, se logró diagnosticar con claridad las limitaciones del sistema original, evidenciando su rigidez en el manejo de consultas, la falta de comprensión semántica y su poca integración con los procesos administrativos de la universidad. Este análisis inicial sirvió como base para diseñar una arquitectura más flexible que respondiera a esas necesidades.

Con la integración de un enfoque RAG y la orquestación de procesos mediante n8n, se logró optimizar la experiencia conversacional incorporando recuperación semántica y generación contextualizada de respuestas. Además, se cumplió con la meta de conectar el flujo con Salesforce, automatizando tareas críticas como la creación de leads e incidencias PQRS, algo que el bot original no podía gestionar. Finalmente, las pruebas en entorno controlado validaron la solución, confirmando que los tiempos de respuesta, la precisión de clasificación y la calidad de la información entregada cumplían con los criterios definidos.

Más allá de estos logros, los aprendizajes fueron clave para entender las particularidades de aplicar IA generativa en un contexto institucional. Se comprobó, por ejemplo, que el ajuste de las instrucciones y el control del contenido son imprescindibles para mantener la coherencia con los lineamientos de la universidad. También se evidenció que la actualización constante de la base documental es esencial para brindar respuestas confiables, y que la adopción de este tipo de soluciones implica no solo un componente técnico, sino también una gestión de expectativas y procesos a nivel organizacional.

El proyecto también permitió reconocer la importancia de mantener una comunicación constante con los actores clave de la universidad. Su acompañamiento fue fundamental, no tanto para definir requisitos técnicos, sino para ayudarnos a identificar y priorizar las interacciones más relevantes que debían ser optimizadas en el bot. Gracias a su experiencia directa con las necesidades de los usuarios, pudimos

enfocar los esfuerzos en los flujos conversacionales de mayor impacto. Además, su retroalimentación constante permitió realizar ajustes puntuales durante el desarrollo, garantizando que la solución respondiera de manera correcta a las dinámicas reales de atención institucional.

Como cierre del desarrollo del proyecto, se realizó una reunión de socialización del producto final con algunos directivos de la universidad y los actores clave involucrados en las fases anteriores del proyecto. En esta sesión participaron:

Nombre	Cargo
Alberto Arias Sandoval	Director de Relacionamiento
Luis Eduardo Rojas Becerra	Jefe de Desarrollo e Innovación TI
Luisa Fernanda Rincón Pérez	Directora del Proyecto
Alexander Valencia Altamirano	Jefe CRM
Carlos Arturo Domínguez Díaz	Ingeniero proyectos de Desarrollo TI
Carlos Augusto Gutiérrez Montilla	Coordinador de análisis de datos TI

Cuadro 7.1: Participantes en la reunión de socialización y validación final del sistema

Durante la reunión se expuso el funcionamiento del sistema conversacional, explicando la solución implementada. Asimismo, se realizó una demostración en vivo del bot, mostrando su capacidad para responder en lenguaje natural, recuperar información institucional de forma contextualizada y automatizar procesos administrativos como la creación de leads y PQRS en Salesforce.

La retroalimentación por parte de todos los asistentes fue altamente positiva. Los directivos destacaron el impacto potencial de la solución para mejorar los canales de atención al público y expresaron su interés en iniciar las labores necesarias para implementar el sistema de forma definitiva en la infraestructura tecnológica de la universidad.

La satisfacción expresada por los directivos y demás asistentes confirma que la solución cumple con los objetivos planteados y aporta valor real a la institución. Como siguiente paso, se acordó iniciar las labores de planeación para su despliegue definitivo dentro del ecosistema tecnológico de la universidad.

# Capítulo 8

## Trabajo Futuro

### 8.1. Trabajos Futuros Propuestos

La entrega de una solución funcional que optimiza la atención al usuario mediante inteligencia artificial y flujos de trabajo inteligentes representó el objetivo principal del presente trabajo de grado. Sin embargo, el desarrollo de este prototipo abre nuevas oportunidades para su evolución y ampliación en el futuro, ya sea mejorando sus capacidades actuales, integrando nuevas funcionalidades o incluso extendiéndolo a otros canales de comunicación institucional. A continuación, se presentan algunas líneas de trabajo que podrían explorarse para fortalecer y expandir el sistema implementado.

#### 8.1.1. Implementación en el Ambiente Productivo Institucional

Como paso siguiente inmediato, está prevista la puesta en marcha del sistema en el entorno productivo de la universidad, de manera que el asistente virtual comience a operar en el canal web institucional real. Este despliegue permitirá:

- Validar el comportamiento del bot en condiciones de uso real con un mayor volumen de consultas.
- Monitorear métricas de desempeño en un contexto operativo continuo.
- Identificar posibles ajustes para optimizar la interacción con la comunidad académica y administrativa.

La implementación en el ambiente productivo representará un punto clave para consolidar la solución como un componente oficial del ecosistema digital institucional y servirá como base para futuras extensiones a otros canales.

### 8.1.2. Ampliación de Cobertura a Todas las Áreas de Acción del Bot

Actualmente, el asistente virtual cubre principalmente las consultas relacionadas con aspirantes y el registro de PQRSF. Un trabajo futuro consistiría en extender su base de conocimiento y sus flujos conversacionales para atender de manera integral otras áreas institucionales que aún no han sido automatizadas, como:

- Servicios académicos y administrativos para estudiantes activos (acceso a calendarios académicos, recursos tecnológicos y bibliotecas).
- Información y procesos de gestión humana para colaboradores y profesores.
- Beneficios, trámites y oportunidades de vinculación para egresados.

Con esta ampliación, el bot podría consolidarse como un canal único de atención que cubra todas las áreas de interacción institucional, reduciendo la necesidad de redirigir al usuario a otros medios y mejorando su experiencia de uso.

### 8.1.3. Entrenamiento Continuo y Aprendizaje Adaptativo

Aunque el sistema actual emplea datos históricos para mejorar su comprensión del lenguaje natural, sería deseable incorporar un mecanismo de aprendizaje continuo. Esto permitiría:

- Retroalimentar el modelo automáticamente con nuevas interacciones.
- Ajustar las respuestas con base en métricas de satisfacción de los usuarios.
- Mejorar la cobertura temática de forma progresiva sin intervención manual constante.

Este enfoque aumentaría la precisión y adaptabilidad del bot a largo plazo.

### 8.1.4. Implementación de Caché para Optimizar Costos

Una posible mejora técnica consiste en la incorporación de un sistema de caché para almacenar temporalmente respuestas previamente generadas por el modelo. Esto permitiría:

- Evitar llamadas redundantes a la API del modelo generativo.

- Reducir costos operativos asociados al consumo de tokens.
- Mejorar los tiempos de respuesta en consultas recurrentes.

De este modo, se optimizaría el uso de recursos y se garantizaría un desempeño más eficiente para las consultas más frecuentes.

### 8.1.5. Análisis Predictivo y Métricas de Impacto

Otra línea futura consiste en utilizar modelos de *machine learning* para identificar patrones de consulta y anticipar necesidades recurrentes de los usuarios. Además, sería valioso implementar un módulo de métricas que permita:

- Medir el tiempo promedio de atención.
- Evaluar la reducción en la carga operativa del personal humano.
- Analizar la satisfacción de los usuarios mediante encuestas automáticas.

Estos datos servirían para mejorar la toma de decisiones institucionales en torno a la atención al usuario.

### 8.1.6. Integración con Fuentes de Conocimiento Dinámicas

Si bien el bot ya incorpora recuperación de información mediante una base vectorial, podría extenderse para conectarse con repositorios documentales más dinámicos, tales como:

- Bases de datos académicas y reglamentarias actualizadas.
- APIs institucionales para consultar información en tiempo real.
- Sistemas externos que brinden soporte a procesos administrativos.

Esto garantizaría respuestas siempre actualizadas y contextualizadas.

### 8.1.7. Bots para Llamadas de Voz Basados en Transcripción

El modelo de transcripción utilizado para procesar contenidos audiovisuales también podría servir como base para crear un bot que atienda llamadas telefónicas en tiempo real. Mediante esta extensión sería posible:

- Transcribir y comprender la consulta hablada del usuario.
- Generar una respuesta dinámica y devolverla como audio.
- Atender consultas de manera inclusiva para usuarios que prefieran comunicación por voz.

Con esta mejora, la atención automatizada podría expandirse más allá de los canales textuales, brindando soporte mediante llamadas de voz.

### 8.1.8. Mejoras en la Experiencia Conversacional

Finalmente, se podrían explorar mejoras en la experiencia de usuario, como:

- Incorporar interacción por voz para facilitar la accesibilidad.
- Enriquecer las respuestas con contenido multimedia como imágenes, videos o formularios interactivos.
- Implementar una curva de respuesta más dinámica según el tipo de usuario (aspirante, estudiante activo, egresado, etc.).

Estos avances harían que el asistente virtual sea más inclusivo y atractivo para todos los públicos.

## **Apéndice A. Acuerdo de Confidencialidad para el acceso al entorno sandbox**

A continuación se incluye el Acuerdo de Confidencialidad (NDA) firmado entre el equipo desarrollador y la Pontificia Universidad Javeriana Cali, documento que respalda el acceso al entorno *sandbox* institucional utilizado durante el proyecto.

## **ACUERDO DE CONFIDENCIALIDAD Y RESERVA EN EL USO DE LA INFORMACIÓN**

Entre los suscritos a saber: **VICENTE DURAN CASAS, S.J.**, mayor de edad, vecino de Cali e identificado con la cedula de ciudadanía No. 3.227.972 de Usaquen, quien obra en nombre y representación de la **PONTIFICIA UNIVERSIDAD JAVERIANA, SECCIONAL CALI**, institución de educación superior privada, de utilidad común, sin ánimo de lucro, con personería jurídica reconocida por medio de la Resolución N° 73 de 1933, emanada del Ministerio de Gobierno, reconocida como Universidad mediante el Decreto 1297 del 30 de mayo de 1964, sometida a la vigilancia del Ministerio de Educación Nacional, en virtud del poder general que le fue conferido por medio de la escritura pública 497 otorgada el 02 de abril de 2014 en la Notaría 26 del Círculo de Bogotá, quien para efectos de este documento se denominará **LA UNIVERSIDAD** por una parte, y por la otra **JUAN MIGUEL ROJAS MEJÍA** y **RAFAEL HERMIDA TOLEDO**, identificado (a) como aparece al pie de su firma y que para todos los efectos se llamarán **LOS ESTUDIANTES**, se celebra el presente acuerdo de confidencialidad y reserva en el uso de la Información, previo a:

### **CONSIDERACIONES**

1. Como parte del desarrollo académico y formativo de los estudiantes de pregrado en Ingeniería de Sistemas y Computación, se establece como requisito para la obtención del título la realización de un proyecto de grado.
2. En este contexto, los estudiantes desarrollaron el anteproyecto “*Mejora de la Atención al Usuario en el bot de atención web de la Pontificia Universidad Javeriana: Implementación de IA y Flujos de Trabajo Inteligentes en Salesforce*” bajo la supervisión de la profesora Luisa Rincón, aplicando y fortaleciendo los conocimientos adquiridos a lo largo de su formación para abordar problemáticas y desafíos en el ámbito de la ingeniería de sistemas y computación.
3. Actualmente, los estudiantes se encuentran en la fase de elaboración de su trabajo de grado, para lo cual requieren acceso a información confidencial almacenada en nuestro CRM. Dicha información es fundamental para el desarrollo de su investigación y análisis.

Teniendo en cuenta lo anterior, las partes acuerdan celebrar el presente acuerdo de confidencialidad, el cual se registrá por las siguientes disposiciones:

### **CLÁUSULAS**

**PRIMERA.- Objeto:** En virtud del presente acuerdo **LOS ESTUDIANTES** asumen la obligación de no revelar, divulgar, o exhibir información financiera, técnica, comercial o científica relacionada con los proyectos en los que participa o tenga conocimiento y otros relacionados o derivados de estos a cargo del Centro de Consultoría y Educación Continua de **LA UNIVERSIDAD**, a persona alguna natural o jurídica, (incluyendo parientes en cualquier grado de consanguinidad o afinidad), ni utilizar o emplear dicha información en su favor o en el de terceros, ya sea de manera directa o indirecta y en perjuicio o no de las anteriores.

**SEGUNDA.- Autorización previa:** Mediante la firma del presente documento, **LOS ESTUDIANTES** aceptan que sin la previa autorización escrita de **LA UNIVERSIDAD** u orden de autoridad competente en ejercicio de funciones legales y en desarrollo de actuación administrativa o judicial, no podrá revelar, divulgar, exhibir y, en general, dar a conocer a ningún tercero, información

o documento alguno que haya recibido o esté por recibir de parte de **LA UNIVERSIDAD**, en razón de sus funciones, relativo a la información legal, financiera, técnica o comercial relacionada con el giro ordinario de su quehacer o de los proyectos a desarrollar.

En consecuencia, **LOS ESTUDIANTES** reconocen y aceptan que **LA UNIVERSIDAD** y/o **LA EMPRESA** son titulares exclusivos de cualquier derecho sobre dicha información, y que no tendrá derecho alguno sobre la misma, obligándose a no copiarla, duplicarla, sustraerla o comunicarla, para sí o para terceros.

**CUARTA.- Cláusula penal:** En adición a lo establecido en la cláusula precedente, el incumplimiento por parte de **LOS ESTUDIANTES** les hará incurrir automáticamente en una pena a favor de **LA UNIVERSIDAD** por una suma equivalente cien (100) salarios mínimos legales mensuales vigentes al momento en que se dé el incumplimiento del presente acuerdo, sin perjuicio de que **LA UNIVERSIDAD** pueda adelantar las acciones legales que tiendan al resarcimiento total de los perjuicios causados. El pago de la suma estipulada como pena, no releva a **LOS ESTUDIANTES** del cumplimiento de la obligación de confidencialidad contraído mediante este acuerdo; en consecuencia, deberá pagar esta penalidad a **LA UNIVERSIDAD** cada vez que se presente un incumplimiento de esta índole. En los términos del inciso segundo del artículo 1595 del Código Civil, **LOS ESTUDIANTES** reconocen expresamente que incurrirá en la pena establecida en la presente cláusula a partir del momento en que ejecute cualquiera de los hechos o conductas que se ha obligado a abstenerse.

**Parágrafo.** Sin perjuicio de la pena pactada en la presente cláusula, **LOS ESTUDIANTES** se harán responsable en forma integral por todos los perjuicios que llegare a causar por el incumplimiento de las obligaciones contraídas dentro de los proyectos en los que intervenga, y por el mal manejo de la información confidencial que le sea confiada o a la que llegare a tener acceso cualquiera que sea su causa.

**QUINTA.- Permanencia de las obligaciones:** La terminación del contrato de prestación de servicios entre **LA UNIVERSIDAD** y **LOS ESTUDIANTES** no lo exonera del cumplimiento de las obligaciones previstas en este acuerdo, las cuales subsisten durante el tiempo de su permanencia en la universidad. En caso de incumplimiento de alguna de estas obligaciones dentro de dicho período, **LA UNIVERSIDAD** se reserva el derecho de reclamar, judicial o extrajudicialmente, la pena y la indemnización plena de los perjuicios causados.

**SEXTA.- Vigencia:** El presente acuerdo regirá durante la vigencia de los proyectos y cinco (5) años más contados a partir de su terminación, cualquiera que sea la causa que dé lugar a ella.

**SÉPTIMA.- Mérito Ejecutivo:** El presente documento junto con la prueba sumaria del incumplimiento por parte de **LOS ESTUDIANTES** prestará mérito ejecutivo para el cobro de las sanciones pecuniarias que en virtud de él se establecen. Bastará también para proceder al cobro ejecutivo, el original del presente acuerdo acompañado de copia auténtica de la carta donde la parte cumplida manifieste a la parte incumplida las causas y motivos de su incumplimiento.

**OCTAVA.- Diferencias:** Toda controversia o diferencia que surja entre las partes con ocasión de la interpretación, ejecución, modificación, suspensión, terminación o incumplimiento de este acuerdo se resolverá mediante conciliación. De no llegarse a un acuerdo conciliatorio, las diferencias se solucionarán por vía judicial.

La conciliación estará sujeta a las siguientes reglas:

- a) La conciliación se llevará a cabo en el Centro de Arbitraje y Conciliación de la Cámara de Comercio de Cali.
- b) El conciliador será escogido de común acuerdo de la lista de conciliadores del Centro de Arbitraje y Conciliación de la Cámara de Comercio de Cali.
- c) La conciliación deberá intentarse dentro de los tres meses siguientes a la presentación en debida forma de la solicitud de conciliación.
- d) Los gastos generados a partir del procedimiento de conciliación serán asumidos por partes iguales entre **LOS ESTUDIANTES** y **LA UNIVERSIDAD**.

**NOVENA.- Notificaciones:** Para los efectos a que haya lugar en el desarrollo del presente contrato, las partes recibirán notificaciones en las siguientes direcciones:

- **LA UNIVERSIDAD** en la calle 18 N.º 118-250, vía Pance de la ciudad de Cali (Valle).
- **LOS ESTUDIANTES**, en la:

<b>Estudiante:</b>	<b>Dirección:</b>	<b>Correo:</b>
Juan Miguel Rojas Mejía	Carrera 2c #47-47	juanrx19@javerianacali.edu.co
Rafael Hermida Toledo		

En constancia de todo lo anterior, se firma en dos originales, en la ciudad de Santiago de Cali, el


\_\_\_\_\_  
LA UNIVERSIDAD

\_\_\_\_\_  
EL ESTUDIANTE

\_\_\_\_\_  
**VICENTE DURAN CASAS, S.J.**  
Rector Seccional Cali

  
\_\_\_\_\_  
**Nombre: Juan Miguel Rojas Mejía**  
**C.C. 1005898078**

\_\_\_\_\_  
EL ESTUDIANTE

  
\_\_\_\_\_  
**Nombre: Rafael Hermida Toledo**  
**C.C. 1078266074**

# Bibliografía

- [1] Javier Rosales Vázquez. Unidad iii. tipos de investigación. <https://sites.google.com/site/metoddelainvest1/unidad-iii-tipos-de-investigacion>, 2010.
- [2] X. Zhang. The application of natural language processing technology based on deep learning in japanese sentiment analysis. <https://ieeexplore.ieee.org/document/10390437>, 2023.
- [3] A. Sharma and S. Sinha. Statistical exploration and sentiment analysis of whatsapp chats using supervised learning. <https://dl.acm.org/doi/10.1145/3647444.3647924>, 2023.
- [4] IBM. What are large language models (llms)? <https://www.ibm.com/topics/large-language-models>, 2023.
- [5] Encyclopaedia Britannica. Large language model (llm). <https://www.britannica.com/topic/large-language-model>, 2025.
- [6] N. Sambhe, G. Yenurkar, and L. Umate. Enhancing customer 360 with better service management using salesforce crm. <https://ieeexplore.ieee.org/document/10093576>, 2022.
- [7] Salesforce. Agentforce: Cree potentes agentes con ia. <https://www.salesforce.com/es/agentforce/>, 2024.
- [8] Anh D. Tran, Jason I. Pallant, and Lester W. Johnson. Exploring the impact of chatbots on consumer sentiment and expectations in retail. <https://www.sciencedirect.com/science/article/pii/S0969698921002848>, 2021.
- [9] R. Sujala and P. Vishalakshi. Role of artificial intelligence in request management: A comprehensive review. <https://ieeexplore.ieee.org/document/10461869>, 2023.
- [10] IBM Research. What is retrieval-augmented generation? <https://research.ibm.com/blog/retrieval-augmented-generation-RAG>, 2023.
- [11] NVIDIA. What is retrieval-augmented generation, aka rag. <https://blogs.nvidia.com/blog/what-is-retrieval-augmented-generation/>, 2025.

- [12] Humanloop. Rag architectures: How retrieval-augmented generation works. <https://humanloop.com/blog/rag-architectures>, 2024.
- [13] Mohammad Taher Pilehvar and Jose Camacho-Collados. *Embeddings in Natural Language Processing: Theory and Advances in Vector Representations of Meaning*. Synthesis Lectures on Human Language Technologies. Springer, 2021.
- [14] Erhan Sezerer and Selma Tekir. A survey on neural word embeddings. *arXiv*, 2021.
- [15] P. K. Singh Thakur and N. Krishnaraj. Whatsapp group chat personality analysis using ai and machine learning techniques. <https://ieeexplore.ieee.org/document/10537307>, 2024.
- [16] Synebo. What is salesforce flow and why do you need it? <https://www.synebo.io/blog/what-is-salesforce-flow-and-why-do-you-need-it/#:~:text=Basically%2C%20Salesforce%20Flow%20enables%20the,or%20run%20on%20a%20schedule.,> 2023.
- [17] Salesforce. About einstein bots in service cloud dialog. [https://help.salesforce.com/s/articleView?id=service.bots\\_service\\_dialog\\_about.htm&type=5](https://help.salesforce.com/s/articleView?id=service.bots_service_dialog_about.htm&type=5), 2023.
- [18] Salesforce. Learn about einstein bots (trailhead module). [https://trailhead.salesforce.com/content/learn/modules/service\\_bots\\_basics/learn-about-einstein-bots](https://trailhead.salesforce.com/content/learn/modules/service_bots_basics/learn-about-einstein-bots), 2023.
- [19] Salesforce. Lead management - what is a lead? <https://www.salesforce.com/products/sales-cloud/features/lead-management/>, 2024.
- [20] R. Hermida y J. Rojas. Repositorio de integración del bot einstein con la api de gemini mediante flask. <https://github.com/rafaaga/flask-llm-gateway>, 2025.
- [21] Salesforce. Digital customer engagement platform pricing. <https://www.salesforce.com/service/digital-customer-engagement-platform/pricing/>, 2024.
- [22] Salesforce. Salesforce add-ons pricing guide. [https://c1.sfdcstatic.com/content/dam/web/en\\_us/www/documents/pricing/all-add-ons.pdf](https://c1.sfdcstatic.com/content/dam/web/en_us/www/documents/pricing/all-add-ons.pdf), 2024.
- [23] R. Hermida y J. Rojas. Repositorio de integración del flujo en n8n. <https://github.com/rafaaga/GenAI-RAG-Chatbot>, 2025.