



**UNIVERSIDAD NACIONAL DE
CATAMARCA**



**FACULTAD DE TECNOLOGÍA
Y CIENCIAS APLICADAS**

Ingeniería en informática

TRABAJO FINAL

**AGENTE INTELIGENTE CONVERSACIONAL APLICADO A
LA AUTOGESTIÓN DE AFILIADOS DE OBRAS SOCIALES**

Autor:

MIGUEL BERNARDO CEJAS ROMERO

Director:

MG. ANA MARIA DEL PRADO

AGRADECIMIENTOS

A mis padres, por el sacrificio, el acompañamiento y el apoyo incondicional a lo largo de toda mi formación académica. En especial a mi madre, quien no me permitió abandonar cuando, en los primeros años de la carrera, atravesé momentos de duda y dificultad. Su constancia, confianza y esfuerzo fueron fundamentales para que hoy pueda alcanzar este logro.

A mi hermano, por el apoyo permanente y el acompañamiento durante el cursado de la carrera, siendo un sostén importante en los momentos de mayor exigencia.

A mis abuelos, que ya no se encuentran físicamente, pero cuyos valores y enseñanzas me acompañaron siempre. En particular, a mi querida abuela Amalia (“Ñata”), quien tuvo un rol fundamental en mi vida. Lamento profundamente que no haya podido verme culminar esta etapa y convertirme en ingeniero, y dedico este trabajo a su memoria con especial cariño.

A mis amigos y compañeros, quienes fueron un apoyo constante a lo largo de la carrera. El compañerismo, la ayuda mutua y las experiencias compartidas fueron determinantes para alcanzar este objetivo, y con el tiempo muchos de ellos se convirtieron también en grandes amigos.

A mis profesores, grandes personas y profesionales, que me acompañaron y guiaron durante todo el proceso formativo, brindándome no solo conocimientos técnicos, sino también apoyo y motivación para continuar avanzando.

Finalmente, a la directora de este trabajo, por su acompañamiento, dedicación y compromiso a lo largo del desarrollo de la tesis, y por el esfuerzo puesto en guiarme en cada etapa.

A todos ellos, mi más sincero agradecimiento.

ÍNDICE/TABLA DE CONTENIDOS

1.1	Contextualización del proyecto	8
1.2	Planteamiento del problema	10
1.3	Antecedentes	12
1.3.1	Evolución histórica de la salud digital y los asistentes	12
1.3.2	Desarrollo internacional de asistentes conversacionales en salud	14
1.3.3	Desarrollo de asistentes conversacionales de autogestión en salud en Argentina	19
1.3.4	Estado actual de la tecnología aplicada al sector salud	24
1.4	Justificación	25
1.5	Objetivos	27
1.5.1	Objetivo General	27
1.5.2	Objetivos Específicos	27
1.6	Diseño Metodológico	28
1.6.1	Tipo de Estudio	28
1.6.2	Enfoque metodológico	28
1.6.3	Integración y ajustes	30
1.6.4	Consideraciones sobre la flexibilidad del proceso metodológico	31
1.6.5	Técnicas e instrumentos de recolección de datos	32
1.6.6	Procedimiento	33
2.1	Agentes Inteligentes y Chatbots	38
2.2	Procesamiento del Lenguaje Natural	40
2.2.1	Definición general y objetivos	40
2.2.2	Principales etapas del procesamiento	41
2.2.3	Modelos clásicos vs. modelos modernos	42
2.2.4	Rol del NLP en los agentes conversacionales	43
2.3	Modelos de Lenguaje y LLMs	44
2.3.1	Definición general de los LLMs	44
2.3.2	Arquitectura Transformer	45
2.3.3	Modelos representativos	46
2.3.4	Capacidades de los LLMs	48
2.3.5	Limitaciones y desafíos actuales	49
2.3.6	Relevancia en el contexto de los agentes conversacionales	51
2.4	Model Context Protocol	52
2.4.1	Definición general del Model Context Protocol	52
2.4.2	Objetivo y motivación del protocolo	52
2.4.3	Arquitectura y componentes principales	53
2.4.4	Funcionamiento operativo del MCP	58
2.4.5	Ventajas del uso de MCP	60

2.4.6. Comparación con enfoques previos	61
2.5. Seguridad, Privacidad y Ética en Sistemas de Salud	62
2.5.1. Marco legal y normativo	62
2.5.2. Riesgos asociados a la digitalización y uso de IA en salud	63
2.5.3. Importancia de la transparencia y la confianza en la IA aplicada a salud	64
2.6. Perspectivas Futuras de los Agentes Conversacionales en Salud	65
2.6.1 Integración multimodal (voz, imagen y texto)	65
2.6.2. Personalización avanzada mediante perfiles de usuario	66
2.6.3. Interoperabilidad con sistemas de Historia Clínica Electrónica (HCE)	67
2.6.4. Potencial del MCP en ecosistemas de IA médica distribuida	68
3.1 Introducción	71
3.2 Identificación y modelado de procesos	71
3.2.1 Alcance funcional y selección de procesos	72
3.2.2 Procesos excluidos del alcance	73
3.2.3 Fuentes de información utilizadas en el relevamiento	73
3.2.4 Modelado de los procesos seleccionados	74
3.2.5 Simplificaciones adoptadas para el prototipo	75
3.2.6 Validación del modelado	75
3.3 Diseño conceptual del prototipo	75
3.3.1 Arquitectura general del sistema	76
3.3.2 Diseño preliminar de la interfaz	78
3.3.3 Frontend	80
3.3.4 Backend y agente conversacional	81
3.3.5 Servidor MCP	82
3.3.6 Base de datos y modelo de información	83
3.4 Fundamentación de la selección de tecnologías y herramientas	85
3.4.1 Criterios generales de selección	85
3.4.2 Lenguaje y framework base	86
3.4.3 Framework conversacional del agente	86
3.4.4 Modelo de lenguaje	87
3.4.5 Interfaz de usuario	88
3.4.6 Base de datos	88
3.5 Implementación del prototipo	89
3.5.1 Implementación de la lógica conversacional	89
3.5.2 Implementación del servidor MCP y herramientas	90
3.5.3 Implementación del backend	91
3.5.4 Implementación del frontend	92
3.5.5 Implementación de la base de datos simulada	93
3.6 Integración del prototipo y ajuste del comportamiento conversacional	94

3.6.1 Integración entre componentes	94
3.6.2 Iteraciones técnicas durante la integración	96
3.6.2.1 Iteraciones menores derivadas de la validación funcional interna	96
3.6.2.2 Ajustes técnicos durante el desarrollo e integración progresiva del prototipo	97
3.6.2.3 Ajustes derivados de observaciones del tribunal evaluador	98
3.6.2.4 Ajustes derivados de la validación experta	99
3.6.3 Validación funcional interna del prototipo	101
3.6.4 Validación experta del prototipo	103
3.7. Cierre del desarrollo	104
3.7.1. Funcionamiento del prototipo	104
3.7.2. Resultado de las pruebas	105
3.7.3. Alcances y restricciones del desarrollo	105
4.1 Síntesis del trabajo realizado	108
4.2. Cumplimientos de los objetivos	109
4.3. Aportes del proyecto	110
4.4. Justificación y valor del uso de MCP en la arquitectura	111
4.5. Limitaciones del trabajo	113
4.6 Líneas de trabajo futuro	114
4.7 Conclusiones Finales	115
Bibliografía	117
Anexo A: Especificación de requerimientos	130
Anexo B: Acta de Validación de requerimientos	157
Anexo C: Esquema de base de datos	160
Anexo D: Validación Funcional del Prototipo	171
Anexo E: Validación Experta	181

ÍNDICE DE FIGURAS

Figura 1 Modelo en cascada con retroalimentación entre fases.	29
Figura 2 Etapas del preprocesamiento de datos en PLN.	42
Figura 3 Arquitectura general del MCP.	54
Figura 4 Arquitectura lógica del prototipo.	77
Figura 5 Interfaz conversacional web.	79
Figura 6 Interfaz conversacional web durante la carga de documentos.	80
Figura 7 Modelo entidad–relación general del sistema.	84

RESUMEN

El sistema desarrollado consiste en un agente conversacional inteligente orientado a la autogestión de afiliados de obras sociales, diseñado con el objetivo de mejorar el acceso a la información y la gestión de trámites administrativos mediante interacción en lenguaje natural. El agente permite a los usuarios realizar consultas personalizadas e iniciar gestiones administrativas de forma automatizada, reduciendo la dependencia de los canales tradicionales de atención y optimizando la experiencia del afiliado.

La solución se basa en la integración de un modelo de lenguaje de gran escala (LLM) con una base de datos institucional simulada, a través de un backend especializado que incorpora el Model Context Protocol (MCP) como mecanismo de control de acciones. Este enfoque permite estructurar las operaciones posibles del sistema, validar la coherencia de las solicitudes y garantizar trazabilidad y control transaccional durante la ejecución de los trámites iniciados desde la conversación.

El prototipo desarrollado contempla procesos representativos de la interacción habitual entre afiliados y obras sociales, combinando consultas informativas personalizadas con trámites administrativos transaccionales. La arquitectura implementada integra una interfaz web conversacional, un agente basado en LLM, un servidor MCP y una base de datos simulada, validándose su funcionamiento mediante pruebas funcionales y validación experta. Los resultados obtenidos demuestran la viabilidad técnica del enfoque propuesto y su potencial aplicación como herramienta de apoyo a la transformación digital del sector salud, particularmente en contextos donde se requiere mayor accesibilidad, integración y eficiencia en la gestión de servicios.

Palabras clave: agente conversacional, inteligencia artificial, modelos de lenguaje, salud digital, obras sociales, autogestión, MCP.

CAPÍTULO I

Introducción



1.1 CONTEXTUALIZACIÓN DEL PROYECTO

En un contexto de creciente demanda de servicios de salud más accesibles, eficientes y personalizados, las obras sociales enfrentan el desafío de modernizar sus canales de atención para responder adecuadamente a las necesidades actuales de sus afiliados. El acceso a información clara, precisa y oportuna en el ámbito sanitario continúa siendo un problema estructural, especialmente para los beneficiarios de estas entidades, quienes con frecuencia enfrentan barreras informativas y administrativas al momento de realizar consultas sobre trámites, coberturas y requisitos vinculados a gestiones médicas (Tapia et al., 2023). La carencia de herramientas digitales integradas y de fácil acceso contribuye a la saturación de los canales tradicionales de atención y a una experiencia de usuario insatisfactoria.

A nivel global, la transformación digital en el sector salud se ha consolidado como un eje estratégico para mejorar la calidad asistencial y la eficiencia de los sistemas sanitarios. La Organización Mundial de la Salud (OMS) promueve desde hace una década la adopción de tecnologías digitales que faciliten el acceso, la equidad y la sostenibilidad en la atención médica (World Health Organization, 2021b). Este proceso incluye el desarrollo de plataformas interoperables, sistemas de historia clínica electrónica, servicios de telemedicina y soluciones basadas en inteligencia artificial que optimizan la toma de decisiones clínicas y administrativas.

En este marco, la salud digital —incluida la telemedicina— ha adquirido una relevancia significativa al facilitar la prestación de servicios de salud a distancia y apoyar la continuidad del cuidado, especialmente en contextos donde existen barreras geográficas. El uso de tecnologías conectadas, plataformas de comunicación seguras y registros digitales contribuye a ampliar el acceso y mejorar la eficiencia del sistema de salud. De manera complementaria, la interoperabilidad de los sistemas de información en salud constituye un requisito fundamental para el intercambio confiable de datos clínicos y para una atención más coordinada y centrada en la persona. (World Health Organization, 2021b).

Frente a este panorama, las tendencias globales evidencian una adopción creciente de soluciones digitales en el ámbito de la salud, entre ellas la digitalización de trámites, los



portales ciudadanos y los asistentes virtuales basados en inteligencia artificial. En particular, los Modelos de Lenguaje Grande (Large Language Models, LLMs) se han consolidado como una tecnología emergente para la atención automatizada de usuarios, gracias a su capacidad para interpretar consultas en lenguaje natural y generar respuestas coherentes y contextualizadas (Denecke, 2024).

A nivel nacional, iniciativas como la Red Nacional de Salud Digital y el portal “Mi Salud” han impulsado la interoperabilidad de sistemas y el acceso remoto a información sanitaria. No obstante, dichas plataformas aún no contemplan interacciones conversacionales personalizadas (Ministerio de Salud de la Nación, s.f.). Esta brecha tecnológica representa una oportunidad concreta para aplicar los conocimientos de la ingeniería en informática al diseño de soluciones centradas en el usuario, orientadas a mejorar la comunicación entre las instituciones de salud y sus afiliados.

La motivación del presente proyecto se origina en la observación de problemáticas detectadas en el ámbito profesional del desarrollo de software para clínicas, sanatorios y obras sociales. Este contacto directo permitió identificar deficiencias recurrentes en la comunicación con los afiliados, así como oportunidades de mejora mediante la incorporación de tecnologías conversacionales modernas.

En este contexto, el proyecto propone el desarrollo de un agente inteligente conversacional destinado a afiliados de obras sociales, basado en un LLM integrado con una base de datos institucional a través de un backend especializado. A diferencia de otros enfoques centrados exclusivamente en la respuesta informativa, este proyecto se apoya en un paradigma de control conversacional emergente, mediante la incorporación de Model Context Protocol (MCP) que estructura las acciones posibles del sistema según el contexto del diálogo. Esta solución permitirá no solo realizar consultas sobre trámites y prestaciones, sino también iniciar gestiones administrativas reales directamente a través de la interacción en lenguaje natural.



1.2 PLANTEAMIENTO DEL PROBLEMA

La interacción de los afiliados con las obras sociales en Argentina se desarrolla en un contexto donde, a nivel global y regional, se reconocen limitaciones estructurales en los sistemas de información en salud, tales como la fragmentación de los datos, la falta de interoperabilidad y las brechas en el acceso efectivo a soluciones digitales (World Health Organization, 2021b; OPS, 2025). Estas condiciones impactan en la eficiencia de los procesos administrativos y en la calidad percibida del servicio, al dificultar la disponibilidad oportuna y confiable de información para la gestión de coberturas y prestaciones. En el ámbito local, esto se refleja en dificultades de accesibilidad y en un aumento de reclamos por problemas de atención sanitaria (Gil Moreira, 2023; Unidad de Auditoría Interna, 2023).

La fragmentación se expresa tanto en el plano organizativo como en el tecnológico. Áreas como afiliaciones, autorizaciones, auditoría médica, farmacia y prestadores externos suelen operar con procedimientos y sistemas que no se encuentran plenamente integrados. La coexistencia de aplicaciones heterogéneas sin estándares comunes ni mecanismos robustos de intercambio de información genera circuitos redundantes, solicitudes reiteradas de datos y dificultades para reconstruir de manera consistente el recorrido de un trámite a través de distintos puntos de contacto (OPS, 2025; World Health Organization, 2021b).

Aunque muchas instituciones han incorporado portales web y aplicaciones móviles, la disponibilidad de canales digitales no garantiza una experiencia fluida si los sistemas subyacentes no están articulados. En estos escenarios, los usuarios pueden enfrentarse a la dispersión de canales, a la ausencia de una visión unificada de su situación y a una integración insuficiente entre los procesos en línea y la atención presencial o telefónica. Esto mantiene una presión elevada sobre los canales tradicionales y contribuye a una percepción de servicio poco previsible (OPS, 2025; Unidad de Auditoría Interna, 2023).

La información necesaria para gestionar coberturas, prestaciones y trámites administrativos suele encontrarse distribuida en múltiples sistemas con criterios de actualización no



siempre homogéneos. Esta desalineación incrementa el riesgo de inconsistencias, duplicidades y demoras en la validación de datos críticos y limita las posibilidades de automatizar verificaciones o consolidar información de manera oportuna (OPS, 2025; World Health Organization, 2021b).

En el caso de las obras sociales y entidades de medicina prepaga, las evaluaciones de auditoría evidencian registros incompletos, falta de documentación respaldatoria y debilidades en los circuitos de información, lo que afecta la capacidad de constatar el cumplimiento normativo y la accesibilidad efectiva a las prestaciones (Unidad de Auditoría Interna, 2023).

Estas deficiencias repercuten tanto en la eficiencia operativa como en la experiencia del afiliado. La ausencia de una infraestructura digital plenamente integrada obliga a destinar recursos significativos a tareas de verificación y conciliación de datos, lo que incrementa la carga administrativa y la probabilidad de errores (OPS, 2025; Unidad de Auditoría Interna, 2023). Bajo la mirada del usuario, la necesidad de utilizar distintos canales para obtener información básica o completar un trámite se asocia con percepciones de complejidad, incertidumbre y escasa previsibilidad, en consonancia con el incremento de reclamos registrados por problemas de acceso a medicamentos, prácticas médicas y prestaciones (Gil Moreira, 2023).

El problema central radica en la limitada capacidad de los sistemas actuales de las obras sociales para ofrecer una experiencia de gestión verdaderamente integrada, accesible y confiable, en un entorno donde persisten la fragmentación de los canales de atención, la ausencia de acceso unificado a la información y las restricciones en la interoperabilidad de los datos sanitarios (World Health Organization, 2021b; OPS, 2025; Unidad de Auditoría Interna, 2023). Esto pone de relieve la necesidad de avanzar en soluciones tecnológicas que articulen consulta, acción y seguimiento en un mismo entorno digital, promoviendo interacciones más directas entre los afiliados y los sistemas institucionales (World Health Organization, 2021b; OPS, 2025).

En este ámbito, las interfaces inteligentes basadas en lenguaje natural y soportadas por infraestructuras de datos interoperables se presentan como una oportunidad para mejorar el



acceso a la información y la gestión de trámites de manera automatizada, segura y contextualizada. Esta orientación es coherente con las recomendaciones de la OMS y de la Organización Panamericana de la Salud (OPS), que subrayan la importancia de fortalecer los sistemas de información para la salud, desarrollar ecosistemas digitales interoperables y promover el uso responsable de tecnologías avanzadas —incluida la inteligencia artificial y el procesamiento del lenguaje natural— como apoyo a la toma de decisiones y a la prestación de servicios centrados en las personas (OPS, 2021; OPS, 2025; World Health Organization, 2021b).

1.3 ANTECEDENTES

1.3.1 EVOLUCIÓN HISTÓRICA DE LA SALUD DIGITAL Y LOS ASISTENTES

En 1966 se desarrolló *ELIZA*, uno de los primeros programas de procesamiento de lenguaje natural, concebido como un experimento en interacción humano–computadora. Uno de sus scripts más conocidos, *DOCTOR*, simulaba el diálogo de un psicoterapeuta mediante reglas simples de coincidencia de patrones, sin comprensión semántica real. Aunque *ELIZA* no fue diseñada como una herramienta clínica, el experimento puso de manifiesto la capacidad de las máquinas para sostener interacciones conversacionales que los usuarios podían percibir como significativas, influyendo posteriormente en el desarrollo de sistemas conversacionales y asistentes digitales, incluidos aquellos aplicados al ámbito de la salud (Weizenbaum, 1966).

Durante la década de 1970 se desarrolló *MYCIN*, un sistema experto en la Universidad de Stanford destinado a apoyar el diagnóstico y tratamiento de infecciones bacterianas, como la bacteriemia y la meningitis. El sistema utilizaba reglas basadas en conocimiento médico y un mecanismo de razonamiento lógico con manejo explícito de la incertidumbre para recomendar antibióticos y dosificaciones adaptadas al perfil del paciente. Aunque *MYCIN* nunca fue implementado en la práctica clínica, principalmente por consideraciones legales y de aceptación profesional, sus resultados experimentales y su enfoque metodológico lo convirtieron en un antecedente fundamental de los sistemas de soporte a la decisión clínica en medicina (Shortliffe, 1976).



En la década de 1980 surgieron INTERNIST-1 y su evolución Quick Medical Reference (QMR), diseñados para asistir el diagnóstico diferencial mediante una base de conocimiento estructurada y un motor de inferencia que, a partir de signos, síntomas y hallazgos, generaba listas ordenadas de posibles diagnósticos. Estos sistemas consolidaron la formalización del conocimiento médico y anticiparon principios de los actuales soportes a la decisión clínica (Miller et al., 1982).

En la década de 1990, con el avance de la digitalización hospitalaria, comenzaron a explorarse los primeros chatbots aplicados en salud, inspirados en el modelo conversacional de ELIZA. Estos sistemas se utilizaron de manera experimental en entornos de salud mental y orientación básica, reproduciendo interacciones textuales que buscaban simular el diálogo terapéutico. Aunque sus capacidades eran limitadas, sentaron las bases para el desarrollo posterior de asistentes virtuales más sofisticados en el ámbito sanitario (Shawar & Atwell, 2007).

En los años 2000 se consolidaron los portales de pacientes, plataformas digitales que permitieron a los usuarios acceder en línea a sus resultados de laboratorio, gestionar turnos médicos y consultar prescripciones. Estos sistemas, aunque aún sin integrar inteligencia artificial avanzada, representaron un cambio significativo en la relación médico-paciente al fomentar el empoderamiento del usuario y la autogestión de la salud. Constituyeron además el antecedente directo de los actuales asistentes inteligentes en salud, que hoy incorporan IA para personalizar la atención y mejorar la experiencia del paciente (Eysenbach, 2001).

A partir de 2010 surgieron asistentes en salud basados en inteligencia artificial y aprendizaje automático, que marcaron un salto cualitativo respecto de los portales tradicionales. Entre los ejemplos más relevantes se encuentra Ada Health (2016), una aplicación que guía al paciente en el autodiagnóstico mediante un sistema de preguntas dinámicas y algoritmos de razonamiento probabilístico. Otro caso destacado es Babylon Health (2017), que combina chatbots médicos con IA para ofrecer consultas remotas y apoyo clínico digital. Estas innovaciones consolidaron la IA como un actor central en la transformación digital de los servicios de salud (Matheny et al., 2019).



Desde 2020 la inteligencia artificial generativa impulsó una nueva etapa en los asistentes en salud, con aplicaciones en triaje automatizado, apoyo al diagnóstico y acompañamiento digital al paciente. Ejemplos recientes incluyen los pilotos con IA generativa en Mayo Clinic (2023) y los asistentes conversacionales en SingHealth (2022), que muestran cómo los sistemas de salud avanzan hacia modelos híbridos de atención, donde la interacción con profesionales se complementa con IA conversacional (Damingo et al., 2025).

1.3.2 DESARROLLO INTERNACIONAL DE ASISTENTES CONVERSACIONALES EN SALUD

La OMS implementó el “WHO Health Alert”, un chatbot disponible en WhatsApp y Facebook Messenger que brinda información sobre síntomas, medidas de prevención, números de casos, preguntas frecuentes y enlaces a recursos oficiales relacionados con la COVID-19. Está disponible en varios idiomas y se utiliza como canal directo para que la población acceda a datos verificados (World Health Organization, 2020).

El Gobierno de India implementó el “MyGov Corona Helpdesk”, un chatbot oficial en WhatsApp diseñado para proporcionar a la ciudadanía información validada sobre la COVID-19. Este asistente permite consultar síntomas, medidas de prevención, ubicación de centros de testeo, números de casos y preguntas frecuentes, además de ofrecer un servicio de verificación de noticias falsas relacionadas con la pandemia. Fue utilizado de manera masiva como canal de comunicación directa entre las autoridades sanitarias y la población (IndiaAI, 2020).

El ABDM Chatbot (Ayushman Bharat Digital Mission), desarrollado por el Ministerio de Salud y Bienestar Familiar de India, funciona como un canal de soporte para la estrategia nacional de salud digital. Este asistente ofrece información sobre el registro de ciudadanos en el sistema, la creación del Ayushman Bharat Health Account (ABHA ID), la vinculación de historiales clínicos digitales, y brinda orientación tanto a pacientes como a profesionales para el uso de los servicios integrados de la misión (APAC News Network, 2024).



El Ministerio de Salud de Brasil implementó un asistente virtual en WhatsApp para apoyar las campañas de vacunación. Este chatbot permite a los ciudadanos consultar información sobre los calendarios de vacunación, recibir recordatorios personalizados, verificar lugares de aplicación de dosis y resolver dudas frecuentes relacionadas con las inmunizaciones. La herramienta fue utilizada especialmente durante la campaña de vacunación contra la COVID-19 como un canal oficial de orientación y contacto directo con la población (Ministério da Saúde, 2023).

El Instituto Mexicano del Seguro Social (IMSS) implementó el “Chatbot Atención VIHrtual”, un asistente en WhatsApp diseñado para brindar información y acompañamiento a personas que viven con VIH. Este chatbot permite consultar sobre prevención, diagnóstico y tratamiento, además de orientar en la localización de servicios especializados y ofrecer recordatorios de citas o toma de medicamentos, funcionando como un canal de apoyo accesible y confidencial (IMSS, 2020).

El Instituto Mexicano del Seguro Social (IMSS) desarrolló “Mi Chat ONCOIMSS”, un asistente conversacional en WhatsApp orientado a pacientes oncológicos y sus familias. La herramienta facilita información sobre prevención, detección temprana y tratamiento del cáncer, además de brindar orientación sobre citas médicas, seguimientos y disponibilidad de servicios especializados dentro del sistema de salud. Su propósito es ofrecer un canal accesible de apoyo y acompañamiento durante el proceso de atención (IMSS, 2021).

La Dubai Health Authority (DHA) de Emiratos Árabes Unidos implementó una línea oficial de atención en WhatsApp, que permite a los ciudadanos realizar reservas de turnos de vacunación, acceder a certificados digitales, consultar información sobre servicios médicos y recibir orientación automatizada mediante inteligencia artificial. Este canal fue especialmente utilizado durante la campaña de vacunación contra la COVID-19 como un medio de autogestión rápida y segura (Government of Dubai Media Office, 2021).

En China, la compañía Ping An Health desarrolló el asistente médico con inteligencia artificial “Ping An Xin Yi”, integrado en su aplicación móvil. Este chatbot permite a los



usuarios realizar consultas de síntomas, recibir orientación sobre posibles diagnósticos, acceder a recomendaciones de autocuidado y coordinar turnos médicos en línea. Además, ofrece un canal para seguimiento de pacientes y gestión de seguros de salud, combinando servicios médicos y administrativos en un único entorno digital (Ping An Health, 2025).

En España, la aseguradora DKV desarrolló la aplicación “Quiero cuidarme Más”, una plataforma de salud digital destinada a sus asegurados. La app permite gestionar turnos médicos, acceder al historial clínico, solicitar recetas electrónicas y realizar consultas médicas en línea mediante chat, llamada o videollamada en cualquier momento. También incorpora un chat médico disponible las 24 horas, orientación en salud mental, programas de coaching y acceso a especialistas según el plan del usuario, integrando un entorno completo de atención médica y apoyo preventivo (DKV Seguros, s. f.).

En Estados Unidos, la aseguradora Aetna ofrece la aplicación “Aetna Health”, que integra un asistente digital basado en canales inteligentes "smart chat". A través de la app, los usuarios pueden consultar información sobre su plan —como cobertura, costos y tarjetas de afiliación— localizar médicos y farmacias dentro de su red, también gestionar y comprender reclamaciones, rastrear gastos y deducibles, y agendar vacunaciones en centros. Asimismo, la aplicación cuenta con un chat inteligente que ofrece respuestas inmediatas y permite conectar con un agente humano si es necesario. La plataforma también incorpora módulos de inteligencia artificial que guían al paciente con recomendaciones personalizadas, opciones de programas de salud y kits de cuidado adaptados a su condición (CVS Health, 2025).

En Estados Unidos, la aseguradora Anthem (Elevance Health) ofrece la aplicación “Sydney Health”, que cuenta con un asistente digital disponible las 24 horas para atender consultas de los afiliados. A través de la app, los usuarios pueden chatear con un bot inteligente para verificar coberturas, acceder a información sobre beneficios, revisar reclamos, consultar saldos y deducibles, y encontrar médicos, laboratorios y farmacias de la red. Además, la app integra servicios de telemedicina para conectarse con médicos vía



chat o videollamada en tiempo real, y permite gestionar recetas, incluyendo la opción de recargas automáticas (Anthem, Inc., s. f.).

En Australia, NIB Group incorporó en su app el “AI-driven Symptom Checker” que guía al usuario mediante un cuestionario de síntomas y lo orienta al nivel de atención apropiado (teleconsulta o consulta con médico general, emergencia hospitalaria o autocuidado). En sus primeros meses (lanzado en febrero de 2024) registró más de 5.000 usos, con derivaciones de 64% a médico general, 15% a emergencias y 21% a monitoreo en casa; está disponible para todos los miembros en Australia y en inglés y chino simplificado dentro de la app (NIB Health, 2024).

En Hong Kong y Tailandia, la aseguradora AXA lanzó el asistente digital “Emma by AXA”, una aplicación que centraliza servicios de gestión de pólizas, reclamos y programas de bienestar. La plataforma permite a los usuarios acceder a su información de seguros, realizar consultas sobre coberturas, tramitar reembolsos, recibir orientación en salud y utilizar herramientas de telemedicina. Además, integra recordatorios y recursos de autocuidado para fortalecer la experiencia digital de los asegurados (AXA, s. f.).

En Estados Unidos, el portal para pacientes Epic MyChart incorporó en 2023 funciones de inteligencia artificial generativa para apoyar la comunicación entre médicos y pacientes. La herramienta permite que, cuando un paciente envía un mensaje a través del portal, la IA redacte un borrador de respuesta que luego es revisado y aprobado por el profesional antes de enviarse. Esta innovación busca reducir la sobrecarga administrativa de los médicos y mantener tiempos de respuesta más ágiles, al mismo tiempo que se garantiza la supervisión humana en todo el proceso (Garcia et al., 2024; Stanford UCSD Health, 2024).

Mayo Clinic implementó herramientas basadas en inteligencia artificial generativa para asistir en la redacción de respuestas a los mensajes que envían los pacientes a través del portal. Estas herramientas facilitan que el personal médico gestione mejor su carga de comunicación, aunque siempre con supervisión humana en el envío final. Investigaciones



internas destacan mejoras en la experiencia del paciente y la eficiencia del equipo sanitario gracias a este soporte automatizado (MeekaC, 2024)

En Colorado, Estados Unidos, el sistema de salud UCHealth implementó en su portal de pacientes “My Health Connection” un proyecto piloto de inteligencia artificial orientado a traducir informes radiológicos al lenguaje cotidiano. El sistema genera explicaciones comprensibles de estudios como resonancias y tomografías junto al informe técnico original, siempre con supervisión médica. Esta innovación busca reducir la ansiedad de los pacientes y mejorar su comprensión clínica, fortaleciendo el rol activo del usuario en la gestión de su salud (UCHealth Today, 2025).

En Estados Unidos, Kaiser Permanente implementó en 2023 un chatbot basado en inteligencia artificial para ayudar a sus afiliados a navegar opciones de atención: el usuario escribe un texto sobre su problema, el sistema responde con preguntas de seguimiento como lo haría un médico, y luego sugiere posibles condiciones y opciones de atención —todo con supervisión humana. Esta herramienta no sustituye al profesional médico sino que hace más accesible y eficiente el acceso a servicios de salud (Becker’s Hospital Review (Díaz), 2023).

En India, Apollo Hospitals lanzó la plataforma digital Apollo 24/7, que facilitó consultas médicas, compras de medicamentos y pruebas diagnósticas desde el hogar. Como parte de su evolución tecnológica, en 2020 también incorporó un chatbot impulsado por inteligencia artificial que evalúa el riesgo de COVID-19 mediante análisis de la calidad de la tos del usuario, integrando el conocimiento clínico de los médicos de Apollo, datos institucionales y fuentes externas como la OMS (Dev, 2020)

En el Reino Unido, la empresa Babylon Health desarrolló un sistema digital de triaje que utiliza inteligencia artificial para apoyar la evaluación inicial de síntomas, el cual fue integrado en el servicio *GP at Hand* del National Health Service (NHS). A través de esta herramienta, los usuarios pueden describir sus síntomas y recibir orientaciones preliminares sobre el tipo de atención más adecuada, como autocuidado o consulta médica.



El servicio fue lanzado en 2017 y alcanzó posteriormente una base de más de 100 000 pacientes registrados. No obstante, el uso de estas herramientas generó críticas y preocupaciones por parte de expertos y reguladores, particularmente en relación con la precisión clínica y la seguridad del sistema (Kobie, 2022; Wikipedia, s. f.).

El análisis de las iniciativas internacionales relevadas muestra un escenario de alta adopción de asistentes digitales en salud, con soluciones que priorizan la disponibilidad de información, la orientación inicial del usuario y, en algunos casos, la automatización de gestiones administrativas simples. A pesar de su madurez tecnológica, estas plataformas tienden a segmentar la interacción en funciones específicas —informar, derivar, sugerir o asistir— y mantienen una separación explícita entre la conversación y la ejecución efectiva de acciones institucionales. Incluso en los casos donde se incorporan modelos avanzados de inteligencia artificial, la capacidad del asistente suele restringirse al apoyo informativo, a la redacción asistida o al triaje preliminar, mientras que la toma de decisiones operativas y la gestión transaccional permanecen bajo control humano o de sistemas externos no conversacionales. En este contexto, se observa que aún no se ha consolidado un enfoque que combine comprensión profunda del lenguaje natural con la autonomía necesaria para coordinar y ejecutar procesos administrativos completos dentro del propio flujo conversacional. La propuesta desarrollada en este trabajo se orienta precisamente a cubrir esta vacancia, al plantear un agente conversacional capaz de integrar diálogo, control de acciones y validación transaccional en un mismo entorno, avanzando hacia un modelo de interacción más autónomo y cohesivo.

1.3.3. DESARROLLO DE ASISTENTES CONVERSACIONALES DE AUTOGESTIÓN EN SALUD EN ARGENTINA

En la Ciudad Autónoma de Buenos Aires se desarrolló “Boti”, un chatbot oficial basado en WhatsApp que constituye un canal transversal de atención ciudadana. Este asistente digital permite a los vecinos realizar múltiples gestiones, entre ellas la solicitud de turnos médicos en hospitales públicos, trámites administrativos y consultas generales, todo a través de un flujo conversacional guiado. La incorporación de Boti en la estrategia de gobierno digital busca acercar los servicios de la ciudad a la población de manera más ágil, intuitiva y



disponible las 24 horas, posicionándose como una de las experiencias pioneras en el país en materia de asistentes conversacionales en el sector público (Buenos Aires Ciudad, s. f.).

La Provincia de Buenos Aires implementó el portal “Mi Salud Digital Bonaerense”, complementado por la plataforma clínica Historia de Salud Integrada (HSI). Esta solución permite a los ciudadanos acceder a su historia clínica unificada, recetas electrónicas, registros de vacunación, solicitud de turnos y servicios de telesalud, además de integrar un chatbot como canal de acceso. El sistema busca mejorar la continuidad del cuidado y la interoperabilidad entre hospitales y centros de salud provinciales, facilitando a los pacientes un mayor control sobre su información y agilizando la gestión de trámites médicos. Se trata de una de las iniciativas provinciales más robustas en materia de salud digital en Argentina (Secretaría de Salud Provincia de Buenos Aires, 2022).

En la Provincia de Mendoza se desarrolló el asistente virtual “MendoBot”, disponible a través de WhatsApp y articulado con la línea telefónica 148, que funciona como centro de contacto ciudadano. Esta herramienta permite gestionar turnos hospitalarios, acceder a información de trámites de salud y recibir orientación sobre servicios provinciales. El uso de WhatsApp como canal central busca acercar la administración pública a la población mediante un medio cotidiano y ampliamente difundido, reduciendo las barreras de acceso digital y ofreciendo disponibilidad permanente. Con esta estrategia, Mendoza se posiciona como una de las provincias pioneras en utilizar la mensajería instantánea para facilitar el acceso a turnos y servicios sanitarios de forma automatizada (Gobierno de Mendoza, 2021).

La Provincia de Neuquén implementó la plataforma ANDES (Aplicaciones Neuquinas de Salud), que integra la Historia Única Digital de Salud (HUDS) y un Portal del Paciente disponible tanto en versión web como aplicación móvil. Esta solución permite a los ciudadanos gestionar turnos médicos, acceder a resultados de estudios, consultar su esquema de vacunación y actualizar datos personales, constituyéndose en un modelo de autogestión sanitaria provincial. El sistema busca fortalecer la interoperabilidad entre los diferentes niveles de atención y garantizar la continuidad del cuidado, al tiempo que ofrece a los usuarios un mayor control sobre su información clínica. ANDES se destaca como una



de las iniciativas más avanzadas de historia clínica digital en el país (Gobierno de Neuquén, 2017).

La Provincia de Santa Fe desarrolló el sistema SICAP (Sistema de Consulta y Administración de Turnos), que permite a los ciudadanos gestionar en línea la solicitud y consulta de turnos médicos en hospitales y centros de salud públicos. Esta herramienta busca simplificar el acceso a la atención sanitaria y reducir las demoras administrativas, ofreciendo un canal digital disponible de manera continua para los usuarios. El SICAP se enmarca en la estrategia de modernización de la gestión de salud provincial, priorizando la accesibilidad y la eficiencia en la organización de la demanda de servicios médicos (Gobierno de Santa Fe, 2024).

En la Provincia del Chaco se implementó un ensayo controlado con aproximadamente 250.000 personas para evaluar un chatbot bidireccional vía WhatsApp, diseñado mediante principios de la ciencia del comportamiento, para promover la vacunación contra el COVID-19. Esta herramienta permitió ofrecer mensajes personalizados, informar sobre elegibilidad, planificar visitas a centros de vacunación y enviar recordatorios. La intervención *triplicó* la tasa de vacunación comparado con no hacer nada, y *duplicó* la efectividad respecto a un mensaje estático. Realizada en un contexto con baja cobertura (solo el 33 % había recibido el refuerzo), esta estrategia demostró la viabilidad y eficacia de la mensajería instantánea como herramienta escalable de salud pública, constituyendo una experiencia destacada en el país en el uso de chatbots para campañas de prevención (Brown et al., 2024; INECO, 2023).

En PAMI, el asistente virtual PAME, disponible a través de WhatsApp, permite a los afiliados y sus familiares realizar una amplia variedad de trámites y consultas las 24 horas del día. Entre otras funcionalidades, posibilita el acceso a cartilla médica, consulta de turnos, credenciales digitales, trámites web y seguimiento de expedientes —todo sin necesidad de atención presencial—. La experiencia fue lanzada tras una alianza con Meta (WhatsApp) en agosto de 2022, tras una etapa piloto que superó las 150.000 interacciones con miles de usuarios activos (PAMI, 2022; La Mañana Neuquén, 2023).



El Instituto de Obra Médico Asistencial (IOMA) de la Provincia de Buenos Aires implementó la aplicación IOMA Digital, que incluye un chatbot disponible las 24 horas como canal de atención a sus afiliados. A través de esta herramienta es posible realizar consultas frecuentes, gestionar autorizaciones médicas, utilizar la credencial digital, solicitar turnos y acceder a recetas electrónicas. La solución busca optimizar los procesos de autogestión, reducir trámites presenciales y agilizar la comunicación entre los afiliados y la obra social. (IOMA, s. f).

La Obra Social de la Ciudad de Buenos Aires (ObSBA) incorpora el asistente virtual “ObSBI”, disponible tanto en WhatsApp como en la web institucional. Este canal permite a los afiliados gestionar turnos médicos, consultar la cartilla de prestadores, localizar farmacias habilitadas y acceder a servicios de telemedicina, además de realizar diversos trámites administrativos en línea. Su diseño apunta a optimizar la experiencia del usuario mediante un acceso rápido y autogestionado a la información y servicios más demandados. Con esta implementación, ObSBA moderniza sus procesos de atención y fortalece la relación digital con los afiliados (OBSBA, 2023).

El grupo de medicina prepaga Swiss Medical desarrolló el asistente virtual “Swity”, disponible a través de WhatsApp, como parte de su estrategia de atención digital a los afiliados. Este bot permite agendar turnos médicos, acceder a recetas electrónicas, consultar la cartilla de prestadores por geolocalización, utilizar la credencial digital, gestionar autorizaciones y reintegros, e incluso establecer un chat con profesionales clínicos o pediátricos en determinados casos. (Swiss Medical, 2024).

La empresa de medicina prepaga OMINT incorporó el asistente virtual “Mint”, disponible tanto en su sitio web como en WhatsApp, para brindar atención continua a sus socios. Esta herramienta permite resolver consultas frecuentes, realizar gestiones simples y acceder a servicios de salud de manera remota. Además, se complementa con la aplicación móvil de la prepaga, que ofrece funcionalidades de telemedicina y autogestión. (OMINT, s. f.).

La empresa de medicina prepaga Sancor Salud implementó el asistente virtual Sandy que opera principalmente a través de WhatsApp y su portal digital, ofreciendo a los afiliados la posibilidad de gestionar credenciales, realizar autorizaciones y tramitar reintegros de



manera inmediata. Este canal de autogestión está disponible las 24 horas, lo que permite a los socios resolver trámites sin necesidad de interacción presencial o llamadas telefónicas. (Sancor Salud, s.f.).

La empresa de medicina prepaga Medifé implementó el asistente virtual “Medi”, disponible en WhatsApp, como canal de autogestión de trámites. A través de esta herramienta, los afiliados pueden consultar facturas, verificar saldos, obtener tokens de validación y acceder a servicios frecuentes de manera rápida y sencilla. “Medi” busca mejorar la experiencia de los socios al ofrecer un canal digital siempre disponible, que reduce la necesidad de interacciones presenciales o telefónicas. Con esta incorporación, Medifé se suma a la tendencia de digitalización de los servicios de salud privados en Argentina, priorizando la accesibilidad y la eficiencia (Medifé, s. f.).

La obra social OSDEPYM anunció el desarrollo de una plataforma omnicanal basada en inteligencia artificial que incluye asistentes virtuales en WhatsApp, chatbots en línea y sistemas de voz interactivos, con el objetivo de ofrecer atención continua las 24 horas. Este enfoque busca centralizar los diferentes canales de contacto en una misma infraestructura tecnológica, permitiendo a los afiliados resolver consultas y trámites de manera automatizada y en cualquier momento. La iniciativa representa un esfuerzo por ampliar la disponibilidad de servicios digitales y mejorar la comunicación con los usuarios mediante soluciones conversacionales integradas (OSDEPYM, 2022).

Las iniciativas desarrolladas en Argentina muestran un énfasis predominante en la digitalización de puntos de contacto específicos —como la gestión de turnos, la consulta de cartillas o el acceso a credenciales digitales— y en la adopción de canales de mensajería como WhatsApp para facilitar la autogestión básica de los afiliados. Sin embargo, estas soluciones suelen operar como extensiones digitales de procesos preexistentes, reproduciendo lógicas fragmentadas y delegando la coordinación efectiva de los trámites en el propio usuario. En este contexto, el asistente actúa principalmente como un intermediario informativo o un disparador de acciones aisladas, sin una capacidad real para orquestar el flujo completo de una gestión administrativa dentro de una interacción conversacional continua. La propuesta desarrollada en este trabajo aborda directamente



esta limitación al introducir un agente conversacional diseñado para el contexto de las obras sociales locales, capaz de conducir al afiliado a lo largo de un trámite completo mediante diálogo en lenguaje natural y de ejecutar las acciones correspondientes de forma controlada y verificable, integrando lógica institucional, validación de datos y registro de operaciones en un único proceso conversacional.

1.3.4. ESTADO ACTUAL DE LA TECNOLOGÍA APLICADA AL SECTOR SALUD

En el ámbito de la salud, la transformación digital ha impulsado el desarrollo de portales y asistentes conversacionales que buscan mejorar la relación entre usuarios e instituciones. Estas herramientas se han convertido en un eje estratégico para reducir la saturación de canales tradicionales de atención y brindar información accesible y oportuna. A nivel global, organismos internacionales han destacado la importancia de la digitalización como medio para garantizar continuidad asistencial, interoperabilidad de datos y empoderamiento de los pacientes en la gestión de su salud (World Health Organization, 2021b).

La incorporación de interfaces conversacionales responde a la necesidad de ofrecer interacciones más naturales y personalizadas. A diferencia de los portales tradicionales, basados en esquemas de navegación estructurados, los asistentes virtuales permiten realizar consultas en lenguaje natural y facilitan el acceso a información y servicios de manera más flexible. Investigaciones recientes indican que los LLMs presentan una mayor capacidad para interpretar las intenciones del usuario y apoyar la realización de tareas complejas dentro de entornos digitales, lo que los posiciona como una tecnología con alto potencial para la atención digital de pacientes y asegurados (Wen et al., 2024; Z. N. & W. P., 2024).

Los sistemas más avanzados han evolucionado desde simples chatbots informativos hacia plataformas transaccionales capaces de gestionar autorizaciones, turnos o reembolsos. Este cambio refleja una tendencia hacia la integración de bases de datos institucionales con interfaces conversacionales, lo que permite optimizar procesos administrativos y reducir tiempos de espera. No obstante, persisten desafíos relacionados con la fiabilidad de la información, la seguridad en el acceso a datos sensibles y la necesidad de marcos normativos que respalden estas prácticas (Xie et al., 2024).



En el contexto latinoamericano, la adopción de estas tecnologías presenta un desarrollo desigual. Si bien varios países han implementado portales de salud y servicios de teleasistencia, la mayoría de las iniciativas se concentran en la digitalización de trámites y no en la conversación inteligente. En Argentina, los esfuerzos institucionales han estado dirigidos principalmente a la interoperabilidad —como lo evidencian la Red Nacional de Salud Digital y el portal “Mi Salud”—, así como a la creación de bases de datos compartidas y registros electrónicos (Ministerio de Salud de la Nación, 2022). Sin embargo, el despliegue de asistentes virtuales basados en IA se encuentra aún en una etapa incipiente. (Ministerio de Salud de la Nación, 2022; Hospital Italiano de Buenos Aires, 2024). Esto evidencia la existencia de una brecha tecnológica frente a países que ya integran soluciones conversacionales avanzadas con infraestructura interoperable. La falta de herramientas capaces de combinar comprensión del lenguaje natural con acceso seguro a datos institucionales limita el potencial de transformación digital del sistema sanitario argentino.

1.4. JUSTIFICACIÓN

La presente propuesta se justifica por la necesidad concreta de mejorar el acceso a la información y la experiencia de los afiliados en su vínculo con las obras sociales en Argentina. Aunque la digitalización ha avanzado en diversos sectores, el ámbito de la salud aún presenta deficiencias en sus canales de comunicación entre usuarios y prestadores, especialmente en lo referido a la interoperabilidad, la integración de sistemas y el diseño de servicios centrados en las personas. Estos desafíos coinciden con los señalados en los lineamientos internacionales para la transformación digital del sector salud, que destacan la importancia de sistemas integrados, accesibles y orientados al usuario (OPS, 2021; OPS, s.f.; World Health Organization, 2021b). Estas limitaciones se manifiestan en la falta de claridad, disponibilidad y facilidad de uso de los servicios informativos, afectando la accesibilidad y la calidad percibida de la atención.

Desde la perspectiva del afiliado, contar con un canal conversacional que no solo brinde información, sino que además permita iniciar y gestionar trámites reales, representa un avance sustancial en términos de accesibilidad, empoderamiento y usabilidad. Desde el



enfoque institucional, la herramienta propuesta contribuye a optimizar recursos humanos y técnicos mediante la automatización de consultas reiterativas, la validación previa de datos y la reducción de errores administrativos, promoviendo una gestión más eficiente y trazable.

El valor diferencial del proyecto radica en la incorporación del MCP, un enfoque emergente que permite controlar la ejecución de acciones derivadas de la conversación, garantizando coherencia, validación y trazabilidad antes de efectuar cualquier operación. Este diseño combina la comprensión del lenguaje natural propia de los LLM con lógicas transaccionales seguras y auditables, ampliando la aplicabilidad práctica de los agentes inteligentes en entornos sanitarios.

A nivel internacional, plataformas como NHS App (Reino Unido), My Health Record (Australia) y MyChart (Estados Unidos) evidencian avances significativos en la autogestión digital de los pacientes, al ofrecer acceso en línea a información clínica, resultados de estudios, gestión de turnos y diversos trámites administrativos (MyChart, s.f.; Australian Digital Health Agency, s.f.; NHS, 2024). aunque aún con limitaciones en el uso de asistentes conversacionales inteligentes.

En Argentina y en gran parte de América Latina, la adopción de tecnologías de salud digital continúa siendo heterogénea y de avance gradual. Aunque existen iniciativas orientadas a la estandarización y a la interoperabilidad —como la Red Nacional de Salud Digital— la región aún enfrenta desafíos significativos en infraestructura, gobernanza, madurez de los sistemas de información y capacidad institucional (Ministerio de Salud de la Nación, 2022; Bastias-Butler & Ulrich, 2019). En este escenario, el desarrollo de un agente conversacional orientado al sector salud local adquiere especial relevancia, tanto por su aplicabilidad práctica como por su potencial de integración con los servicios y lineamientos definidos para la transformación digital del sistema sanitario.

Desde el punto de vista académico, la investigación constituye un aporte al campo de la informática en salud al explorar la integración de tecnologías conversacionales de última generación con arquitecturas de control orientadas a la acción. Mediante un enfoque



centrado en el usuario y validado a través de simulaciones realistas, se busca sentar bases conceptuales y técnicas para futuras implementaciones operativas.

1.5. OBJETIVOS

1.5.1 OBJETIVO GENERAL

Desarrollar un agente conversacional inteligente para afiliados de obras sociales, basado en un LLM integrado a una base de datos institucional mediante un backend controlado por un MCP, con el fin de habilitar la consulta de información y la iniciación estructurada de trámites administrativos a través del diálogo en lenguaje natural, asegurando validación, trazabilidad y autonomía en la interacción.

1.5.2 OBJETIVOS ESPECÍFICOS

1. Delimitar y modelar procesos de autogestión seleccionados, representativos de la interacción de afiliados con obras sociales.
2. Diseñar una interfaz web accesible que permita la comunicación fluida con el sistema en lenguaje natural.
3. Implementar un sistema basado en LLM capaz de interpretar las intenciones del usuario y canalizarlas mediante el backend a través de un MCP que regule las acciones posibles.
4. Configurar el MCP como capa de validación lógica que asegure la coherencia semántica, la integridad de las gestiones y el control sobre la base de datos.
5. Integrar funcionalidades que permitan registrar gestiones administrativas reales en una base de datos simulada.
6. Validar conceptualmente la utilidad, usabilidad y eficacia del prototipo mediante pruebas con escenarios simulados representativos de casos reales, complementadas por una validación experta.



1.6. DISEÑO METODOLÓGICO

1.6.1. TIPO DE ESTUDIO

El trabajo se clasifica como un estudio de desarrollo tecnológico. El objetivo central es diseñar, implementar y validar un prototipo de agente conversacional inteligente integrado con el MCP y una base de datos simulada, aplicado a la gestión de trámites administrativos de obras sociales.

1.6.2 ENFOQUE METODOLÓGICO

El proyecto adopta un enfoque metodológico híbrido basado en un modelo de cascada con retroalimentación controlada. Esta elección se fundamenta en las características del dominio abordado, en la naturaleza del prototipo desarrollado y en la distinción conceptual existente entre el desarrollo de modelos de aprendizaje automático y el desarrollo de sistemas de software que integran modelos preentrenados.

En primer lugar, resulta necesario diferenciar este proyecto de los enfoques iterativos característicos del aprendizaje automático. En este tipo de sistemas, los ciclos experimentales son indispensables debido a la necesidad de entrenar y reentrenar modelos, ajustar hiperparámetros, realizar evaluaciones continuas basadas en métricas y gestionar la dependencia de datos que pueden variar a lo largo del tiempo (Breck et al., 2017). Ninguna de estas actividades está presente en este trabajo, dado que el agente conversacional se construye sobre un modelo de lenguaje preentrenado accesible mediante una interfaz de programación de aplicaciones (API). Dicho modelo no se entrena, no se ajusta ni se optimiza dentro del alcance del proyecto, y su comportamiento permanece estable durante todo el proceso de desarrollo. En consecuencia, no se justifica la adopción de metodologías propias del aprendizaje automático, tales como MLOps, la experimentación iterativa o la validación estadística de modelos.

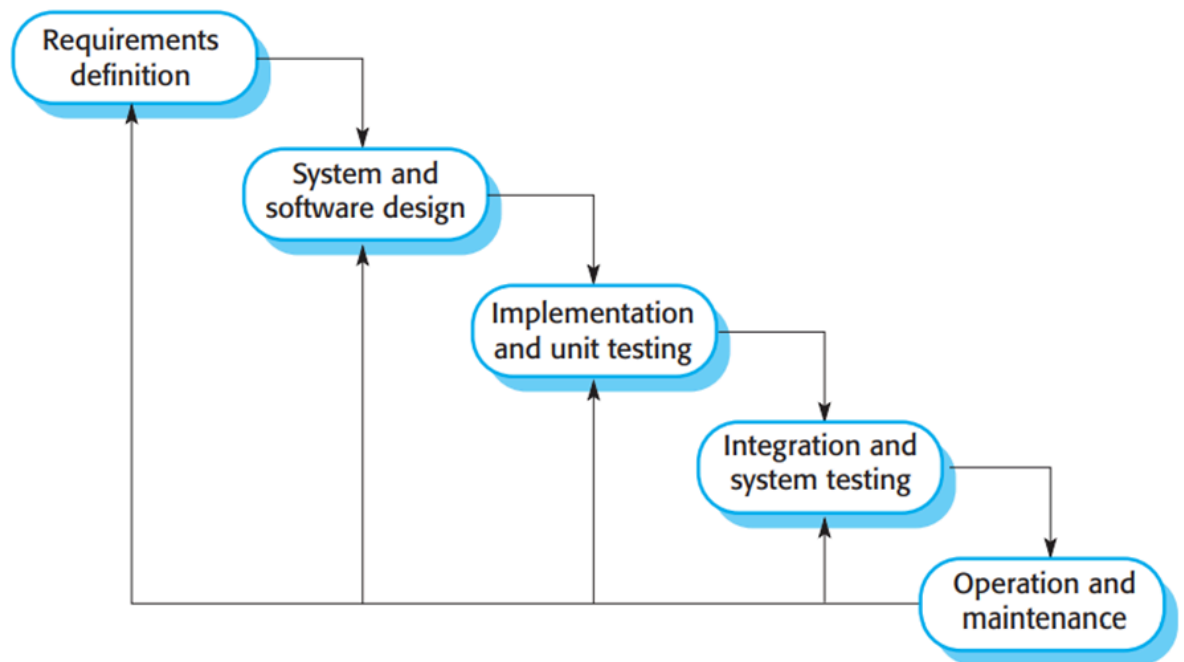
En segundo lugar, las características del dominio favorecen una estructuración del desarrollo por fases. Los procesos administrativos seleccionados para el prototipo —reintegros, afiliaciones, comunicaciones y consultas informativas— corresponden a procedimientos institucionales estables, regulados y con baja variabilidad operativa. Esta

condición permite definir requerimientos funcionales completos desde el inicio del proyecto, escenario que resulta adecuado para la aplicación de metodologías de desarrollo secuenciales (Sommerville, 2016).

No obstante, el modelo en cascada no debe interpretarse como un proceso estrictamente rígido. Aunque su representación es lineal, en la práctica admite retroalimentación entre fases y revisiones acotadas cuando se detectan inconvenientes durante etapas posteriores como la implementación, la integración o la validación (Sommerville, 2016). Esta concepción se ajusta a la dinámica del presente proyecto, en el cual se mantuvo una estructura metodológica por fases, incorporando ajustes puntuales en etapas avanzadas únicamente cuando la evidencia técnica así lo requiere.

Figura 1

Modelo en cascada con retroalimentación entre fases.



Nota. Adaptado de Sommerville (2016).

Este modelo refleja con precisión la metodología adoptada en el presente trabajo. Si bien el desarrollo avanzó de manera predominantemente secuencial, se incorporaron ajustes



acotados cuando se identificaron oportunidades de mejora. Dichos ajustes surgieron principalmente durante la integración técnica del backend, el servidor MCP, el modelo de lenguaje y el frontend, así como durante las instancias de validación funcional interna y validación experta. En todos los casos, las modificaciones realizadas fueron localizadas y no implican la redefinición del alcance ni la reestructuración de fases completas del proceso de desarrollo (Sommerville, 2016).

El enfoque metodológico adoptado puede describirse como un modelo de cascada con retroalimentación entre fases: una estructura secuencial orientada a garantizar orden, trazabilidad y estabilidad, complementada con ciclos de retroalimentación controlados que permiten introducir correcciones cuando las pruebas técnicas o la retroalimentación experta así lo requieren. Este esquema resulta adecuado para el desarrollo de un prototipo que no presenta la incertidumbre ni la variabilidad propias de los procesos de entrenamiento de modelos de aprendizaje automático, pero que sí demanda ajustes técnicos derivados de la integración entre componentes de software heterogéneos (Breck et al., 2017; Sommerville, 2016).

La elección del enfoque metodológico se sustentó en tres factores principales: la estabilidad del dominio abordado, la ausencia de procesos propios del machine learning y la necesidad de contar con una estructura ordenada capaz de incorporar ajustes puntuales sin comprometer las decisiones fundamentales adoptadas en las fases iniciales del proyecto. Bajo estas condiciones, el modelo en cascada con retroalimentación controlada constituye una alternativa metodológica adecuada para el desarrollo del prototipo propuesto (Sommerville, 2016).

1.6.3. INTEGRACIÓN Y AJUSTES

La integración entre los distintos componentes del sistema se desarrolló de manera progresiva y ordenada, siguiendo la secuencia definida en el diseño del prototipo. Debido a la naturaleza heterogénea de estos elementos, la integración requirió ciclos sucesivos de prueba y verificación destinados a asegurar su funcionamiento conjunto. En este proceso surgieron ajustes técnicos puntuales, orientados principalmente a afinar la interacción entre el agente conversacional y las herramientas MCP, alinear las validaciones internas con la



lógica del modelo preentrenado y garantizar la coherencia de los flujos operativos implementados.

Los ajustes realizados durante esta etapa constituyeron iteraciones internas propias del ensamblado de componentes en sistemas de software. En todos los casos se trató de modificaciones acotadas, focalizadas en resolver inconsistencias o mejorar comportamientos específicos sin alterar la arquitectura general ni los requerimientos funcionales ya establecidos. Su propósito fue asegurar la estabilidad del prototipo a medida que los módulos se integraban y comenzaban a operar de manera conjunta.

La retroalimentación técnica aportada por expertos en sistemas informáticos utilizados por obras sociales también contribuye a este proceso. Sus observaciones permitieron aclarar condiciones operativas de ciertos trámites y ajustar el comportamiento conversacional del agente cuando resultó necesario para reflejar con mayor precisión los circuitos administrativos reales. Estas intervenciones complementaron las pruebas internas y permitieron detectar oportunidades de mejora que no habían sido evidentes durante la integración puramente técnica.

1.6.4. CONSIDERACIONES SOBRE LA FLEXIBILIDAD DEL PROCESO METODOLÓGICO

Si bien el proyecto se estructuró a partir de una secuencia metodológica ordenada, su ejecución requirió incorporar un grado acotado de flexibilidad para responder a hallazgos surgidos durante la construcción del prototipo. Esta necesidad es característica de desarrollos que integran tecnologías emergentes.

La flexibilidad introducida no implicó modificaciones al alcance del proyecto ni a la arquitectura conceptual definida, sino que se manifestó en la posibilidad de ajustar decisiones técnicas específicas dentro de cada fase, manteniendo la trazabilidad y el control del proceso. De este modo, las iteraciones realizadas se encuadran como refinamientos internos compatibles con un enfoque por etapas, y no como redefiniciones del proceso metodológico.

Este enfoque resulta consistente con la perspectiva planteada por Sommerville (2016), quien señala que los modelos secuenciales admiten retroalimentación entre fases cuando



emergen nuevos conocimientos durante el desarrollo. En el marco de este trabajo, dicha flexibilidad permitió preservar la coherencia metodológica general, al tiempo que se garantizó la adaptación necesaria para alcanzar un prototipo funcional y alineado con los objetivos establecidos.

1.6.5. TÉCNICAS E INSTRUMENTOS DE RECOLECCIÓN DE DATOS

La recolección de datos tuvo como propósito reunir la información necesaria para definir y modelar los procesos institucionales que servirían como base del prototipo. Dado que el proyecto se orientó al desarrollo de un agente conversacional integrado con un modelo de lenguaje preentrenado y no al entrenamiento de dicho modelo, las técnicas utilizadas se enfocaron exclusivamente en la comprensión del dominio y en la identificación precisa de los trámites a representar.

Para ello se emplearon tres técnicas principales: revisión documental, análisis comparativo de procesos y entrevistas semiestructuradas con expertos del dominio. Estas técnicas se complementaron entre sí y permitieron establecer un conjunto de procesos estable, representativo y viable de implementar en un entorno prototípico.

Revisión documental

Se realizó una revisión de reglamentos, instructivos, normativas vigentes y descripciones operativas de diversas obras sociales. El objetivo fue reconstruir la estructura formal de los trámites más frecuentes para los afiliados, identificar requisitos administrativos, documentación asociada y condiciones de inicio de cada proceso.

Este relevamiento permitió elaborar un listado preliminar de posibles trámites y comprender las variaciones normativas que podrían presentarse entre instituciones, lo que facilitó la posterior selección de un subconjunto manejable y representativo para el prototipo.

Análisis comparativo de procesos



A partir de la documentación recopilada, se llevó a cabo un análisis comparativo orientado a identificar patrones comunes entre obras sociales y a reducir la variedad de trámites a un conjunto coherente para el proyecto. Este análisis consideró tres criterios:

- Frecuencia de uso y relevancia operativa.
- Diversidad funcional, combinando trámites transaccionales y consultas informativas personalizadas.
- Viabilidad de implementación dentro del alcance de un trabajo final individual.

Con base en estos criterios se seleccionaron seis procesos —tres transaccionales y tres informativos personalizados— que definieron el alcance funcional estable del prototipo. Esta selección temprana permitió estructurar el resto del proyecto sin replanificaciones posteriores.

Entrevistas semiestructuradas con expertos del dominio

Para complementar el relevamiento documental se realizaron entrevistas semiestructuradas con personal técnico vinculado al desarrollo y operación de sistemas informáticos utilizados por obras sociales. Estas entrevistas se centraron en:

- Verificar la correspondencia entre los procesos modelados y la operatoria real.
- Aclarar condiciones particulares o excepciones habituales.
- Identificar requisitos que no se encuentran explícitos en la documentación pública.

Las entrevistas no tuvieron por finalidad redefinir los procesos ni modificar el alcance funcional, sino validar y precisar la representación conceptual construida previamente. Los ajustes derivados consistieron en correcciones menores en los flujos operativos y en la interpretación de ciertos requisitos administrativos, sin alterar la selección de procesos ni los límites del prototipo.

1.6.6. PROCEDIMIENTO

El procedimiento seguido en el proyecto se organizó en seis fases consecutivas, coherentes con el enfoque metodológico híbrido por fases adoptado. Cada fase produjo resultados



concretos que sirvieron de insumo para la siguiente, manteniendo una estructura general ordenada. No obstante, el desarrollo incorporó iteraciones técnicas acotadas en etapas posteriores, principalmente orientadas a la validación y ajuste de la implementación, sin reabrir fases conceptuales ya consolidadas ni modificar el alcance funcional del prototipo.

Fase 1

Esta etapa estuvo orientada a delimitar y formalizar el alcance funcional del prototipo mediante el análisis del dominio y el modelado de los requerimientos del sistema. Inicialmente, a partir de la revisión documental y de un análisis preliminar de los trámites institucionales, se identificó un conjunto amplio de procesos potenciales. Dicho conjunto fue depurado aplicando criterios de frecuencia de uso, relevancia operativa y viabilidad de implementación en el contexto de un trabajo final individual.

Como resultado de este análisis se definió un subconjunto estable de seis procesos, compuesto por tres trámites transaccionales y tres consultas informativas personalizadas, que constituyó el alcance funcional definitivo del prototipo. Esta definición permite acotar el dominio del problema y establecer un marco claro para el diseño posterior del sistema.

Sobre la base de este alcance, se realizó el análisis y modelado detallado de cada proceso seleccionado, describiendo sus pasos, condiciones de inicio, datos de entrada, validaciones requeridas y resultados esperados, diferenciando según su naturaleza transaccional o informativa. El modelado fue contrastado mediante entrevistas semiestructuradas con personal técnico con experiencia en el desarrollo y operación de sistemas de obras sociales, lo que permitió verificar la correspondencia entre los flujos propuestos y la operatoria real. Los ajustes surgidos de esta validación se limitaron a precisiones sobre condiciones particulares y excepciones frecuentes, sin modificar el alcance funcional definido. El resultado final fue un modelo de procesos coherente y validado, que sirvió como base conceptual para el diseño del sistema.

Fase 2

La segunda fase se centró en el diseño conceptual del prototipo. Sobre la base del modelo de procesos se definió la estructura general del sistema, identificando los componentes



principales —interfaz web, backend, servidor MCP, agente conversacional y base de datos— y las relaciones entre ellos.

En esta etapa se estableció también el modelo de información necesario para representar los datos indispensables del dominio en un entorno prototípico, mediante un esquema entidad-relación simplificado que describe afiliados, prácticas, topes, consumos y solicitudes. El diseño conceptual fijó la organización interna del sistema y la distribución de responsabilidades entre componentes, manteniéndose estable durante el resto del proyecto, salvo ajustes operativos acotados introducidos durante la integración.

Fase 3

En esta fase se fundamentó y consolidó la selección de las tecnologías y herramientas utilizadas en la implementación del prototipo, en coherencia con los requerimientos analizados y el diseño conceptual del sistema. Se definió el uso de Python y FastAPI para el backend, React para el frontend, PostgreSQL como motor de base de datos, un servidor MCP para la ejecución de operaciones del dominio y LangGraph como framework para el agente conversacional.

Asimismo, se llevó a cabo un proceso exploratorio orientado a justificar la elección del modelo de lenguaje a integrar, evaluando distintas alternativas accesibles mediante API en función de su estabilidad, capacidad de seguimiento de instrucciones y adecuación al dominio. Este proceso permitió seleccionar un modelo preentrenado concreto para la versión final del prototipo. Si bien estas pruebas se extendieron parcialmente sobre fases posteriores, no implican cambios en la arquitectura general ni en el alcance del proyecto, sino iteraciones técnicas propias de la integración de IA generativa en sistemas de software.

Fase 4

La cuarta fase correspondió a la implementación de los componentes del prototipo. Se construyó la base de datos simulada de acuerdo con el modelo de información definido, se desarrolló el backend con las funciones necesarias para coordinar la interacción entre la interfaz y el agente, se implementó el servidor MCP con sus herramientas asociadas a los



procesos del dominio y se materializó la interfaz conversacional en React siguiendo los mockups diseñados previamente.

Durante esta fase se realizaron ajustes propios del desarrollo, tales como correcciones en validaciones, manejo de errores y adaptación de ciertos mensajes a las necesidades del flujo conversacional. Estas modificaciones se mantuvieron dentro de la fase de implementación y no implican replantear el diseño conceptual ni reabrir decisiones metodológicas anteriores.

Fase 5

La quinta fase abarcó la integración del sistema y su validación funcional y experta. En primer lugar, se integraron frontend, backend, agente conversacional, servidor MCP y base de datos, verificando que el flujo completo de interacción operará de forma coherente. Sobre el prototipo integrado se ejecutó luego una validación funcional interna, definida a partir de casos de prueba representativos de los seis procesos seleccionados. Esta validación permitió detectar ajustes menores en el comportamiento conversacional, que se incorporaron como iteraciones técnicas acotadas dentro de la misma fase.

Finalmente, se realizó una validación experta con profesionales vinculados a sistemas utilizados por obras sociales, quienes revisaron los flujos implementados bajo un enfoque del dominio. Sus observaciones dieron lugar a precisiones adicionales en el comportamiento del agente y a ciertos refuerzos operativos, sin modificar la arquitectura ni el alcance funcional del prototipo.

Esta fase permitió confirmar que el sistema implementado no solo funcionaba correctamente desde el punto de vista técnico, sino que también era coherente con los procesos institucionales modelados y con las prácticas habituales del sector.

CAPÍTULO II

.

Marco Teórico



El desarrollo del presente proyecto se apoya en un conjunto de conceptos clave que permiten comprender tanto la problemática como la solución propuesta. Estos marcos conceptuales orientan la formulación, el diseño y la validación de la herramienta tecnológica.

2.1. AGENTES INTELIGENTES Y CHATBOTS

Los agentes conversacionales inteligentes son sistemas de software diseñados para interactuar con las personas mediante lenguaje natural —ya sea por texto o voz— con el propósito de interpretar las consultas del usuario, generar respuestas pertinentes y, en algunos casos, ejecutar acciones específicas. Estos sistemas integran técnicas de procesamiento del lenguaje natural (PLN) y modelos de aprendizaje automático (AA) preentrenados, lo que les permite adaptarse a nueva información, optimizar su desempeño y mejorar la calidad de la interacción de manera progresiva (Radziwill y Benton, 2017).

Desde una perspectiva arquitectónica, un agente conversacional inteligente suele estructurarse en tres módulos principales:

1. Módulo de percepción o comprensión del lenguaje, encargado de procesar la entrada del usuario y transformarla en una representación semántica interpretable por el sistema.
2. Módulo de razonamiento o procesamiento, responsable de inferir la intención del usuario, consultar bases de conocimiento y decidir la acción o respuesta más adecuada.
3. Módulo de generación de salida, que produce la respuesta en lenguaje natural, garantizando coherencia, claridad y continuidad en la interacción comunicativa.

El objetivo fundamental de estos sistemas es simular conversaciones humanas significativas, manteniendo la coherencia contextual y la adecuación semántica a lo largo del intercambio. A diferencia de los sistemas basados en reglas fijas, los agentes inteligentes poseen la capacidad de aprender de interacciones previas, ajustar sus estrategias de diálogo y razonar sobre el contexto, lo que los acerca a un comportamiento más flexible, autónomo y adaptativo.



Dentro del espectro de los sistemas conversacionales, los chatbots tradicionales pueden considerarse una etapa inicial de desarrollo, caracterizada por su funcionamiento determinístico. Estos sistemas se basan en reglas predefinidas o flujos de diálogo rígidos que asocian palabras clave con respuestas específicas. Este enfoque resulta funcional para tareas limitadas, pero presenta importantes restricciones frente a variaciones lingüísticas, ambigüedades o consultas no contempladas, reduciendo así su capacidad para interpretar el contexto o las intenciones implícitas del usuario (Caldarini, Jaf y McGarry, 2022). En consecuencia, los chatbots de este tipo responden de manera estática ante entradas conocidas y carecen de mecanismos de aprendizaje, inferencia o comprensión contextual profunda.

La evolución de estos sistemas ha dado lugar a una categoría más avanzada: los asistentes virtuales, los cuales incorporan mayores niveles de autonomía y razonamiento contextual. Estos sistemas no solo comprenden y responden a consultas, sino que también poseen capacidad de agencia, es decir, la posibilidad de ejecutar acciones en representación del usuario. A diferencia de los chatbots tradicionales, cuyo propósito se limita a la entrega de información o a la gestión de diálogos acotados, los asistentes virtuales integran PLN, módulos de gestión de tareas y conexión con servicios externos o dispositivos inteligentes.

Ejemplos ampliamente difundidos son Siri, Alexa y Google Assistant, que actúan como interfaces conversacionales multimodales. Mediante lenguaje natural, permiten acceder a una amplia gama de servicios y APIs distribuidos en múltiples dominios: desde la programación de recordatorios hasta el control de dispositivos domésticos o la ejecución de acciones automatizadas. Estas plataformas representan una evolución hacia sistemas conversacionales más integrados, contextualmente conscientes y dotados de capacidades de razonamiento más sofisticadas (Rastogi et al., 2019; Kot & Nguyen, 2020).

Los agentes conversacionales inteligentes han encontrado aplicaciones transversales en diversos ámbitos sociales y productivos. En el sector educativo, se emplean como tutores virtuales capaces de ofrecer retroalimentación personalizada, responder consultas académicas o asistir en procesos de aprendizaje adaptativo. En el comercio y los servicios, los chatbots automatizan la atención al cliente, gestionan pedidos y mejoran la experiencia de usuario mediante recomendaciones personalizadas basadas en análisis de comportamiento.



En el ámbito de la salud, los asistentes virtuales apoyan tareas de orientación médica, recordatorios de medicación y triage inicial, contribuyendo a una atención más accesible y eficiente. Finalmente, en el sector gubernamental, estos sistemas facilitan la comunicación entre ciudadanos e instituciones, automatizando consultas, trámites y servicios públicos digitales. En conjunto, estas aplicaciones evidencian el potencial transformador de los agentes conversacionales para optimizar procesos, reducir cargas operativas y mejorar la calidad de la interacción entre humanos y sistemas tecnológicos.

2.2. PROCESAMIENTO DEL LENGUAJE NATURAL

2.2.1. DEFINICIÓN GENERAL Y OBJETIVOS

El Procesamiento del Lenguaje Natural (PLN, o en inglés Natural Language Processing, NLP) es una rama de la inteligencia artificial y de la lingüística computacional que se centra en la interacción entre los sistemas informáticos y el lenguaje humano. Su propósito es estudiar cómo las máquinas pueden comprender, interpretar y generar lenguaje humano, tanto en forma escrita como hablada (IBM, 2024).

El objetivo principal del PLN es permitir una comunicación fluida entre personas y computadoras, posibilitando que las máquinas comprendan el lenguaje natural mediante el análisis de sus estructuras sintácticas y semánticas. Este campo busca que los sistemas informáticos capten significados, intenciones y contextos, reduciendo así la ambigüedad inherente al lenguaje humano (TechTarget, 2024).

Asimismo, el PLN se aplica en una amplia variedad de tareas, tales como la traducción automática, el análisis de sentimientos, la clasificación de texto, la extracción de información y la generación automática de respuestas. A través de estas aplicaciones, el procesamiento del lenguaje natural contribuye a disminuir la ambigüedad propia del lenguaje humano y a dotar a las computadoras de capacidades de comprensión y generación lingüística. De esta manera, las tecnologías informáticas logran integrarse más eficazmente en procesos de comunicación, análisis y toma de decisiones (Amazon Web Services, s.f.).



2.2.2. PRINCIPALES ETAPAS DEL PROCESAMIENTO

El PLN se desarrolla mediante una serie de fases que conforman un *pipeline* o flujo de procesamiento. Estas etapas permiten analizar el lenguaje humano de forma estructurada y comprensible para las máquinas, abarcando desde la segmentación del texto hasta la identificación de información relevante (GeeksforGeeks, 2025).

La primera etapa es la tokenización, proceso mediante el cual el texto se divide en unidades mínimas denominadas tokens, que pueden ser palabras, frases o símbolos. Este paso es esencial, ya que constituye la base sobre la cual se construyen las fases posteriores del análisis lingüístico (GeeksforGeeks, 2025).

Luego se aplican técnicas de lematización y stemming, que tienen como objetivo normalizar las palabras reduciéndolas a su forma base o raíz. El stemming elimina afijos para obtener la raíz morfológica, mientras que la lematización analiza la categoría gramatical para devolver la palabra a su forma canónica. Ambas técnicas facilitan la comparación semántica y el procesamiento eficiente del texto (Khan, 2025).

Posteriormente, se realiza el etiquetado gramatical o Part-of-Speech Tagging (POS), que consiste en asignar a cada token una categoría sintáctica, como sustantivo, verbo o adjetivo. Esta fase permite comprender la estructura de las oraciones y la función de cada palabra en el contexto del texto (GeeksforGeeks, 2025).

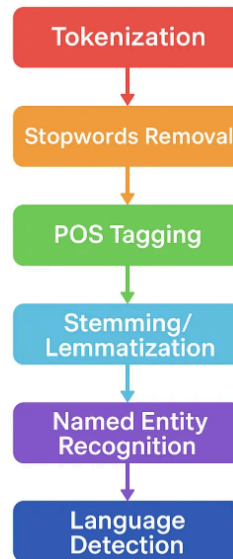
El reconocimiento de entidades nombradas (Named Entity Recognition, NER) representa otra tarea importante dentro del pipeline de PLN. Este proceso identifica y clasifica información clave en el texto, como nombres de personas, lugares u organizaciones. Su función principal es agregar estructura lingüística —por ejemplo, reconocer “Nueva York” como una ubicación— que luego puede utilizarse como característica para otras tareas de análisis o extracción de información (Khan, 2025).

Finalmente, la detección de idioma permite identificar en qué idioma está escrito un texto. Esta etapa es especialmente útil cuando se trabaja con grandes volúmenes de datos multilingües, ya que posibilita aplicar modelos específicos según el idioma detectado, garantizando una mejor precisión en las etapas posteriores del procesamiento (Khan, 2025).

Figura 2

Etapas del preprocesamiento de datos en PLN.

NLP Data Preprocessing



Nota. Adaptado de *Understanding the NLP pipeline*, Khan, 2025.

Como se muestra en la Figura 2, el flujo típico de procesamiento del lenguaje natural incluye fases de tokenización, eliminación de stopwords, etiquetado gramatical, lematización o stemming, reconocimiento de entidades y detección de idioma (Khan, 2025).

2.2.3. MODELOS CLÁSICOS VS. MODELOS MODERNOS

A lo largo del desarrollo del PLN, los métodos empleados han evolucionado significativamente, pasando de enfoques basados en reglas y estadísticas a modelos neuronales de gran escala capaces de comprender el contexto y generar lenguaje natural con alta precisión.

Los modelos clásicos del PLN se apoyaban principalmente en reglas lingüísticas diseñadas manualmente o en métodos estadísticos simples. Entre ellos se encuentran los modelos de n-gramas y el modelo de bolsa de palabras (bag-of-words), los cuales representaban el texto como conjuntos de palabras independientes, ignorando el orden y las relaciones semánticas



entre ellas. Aunque estos métodos permitían analizar patrones de frecuencia y realizar tareas básicas como clasificación o filtrado de texto, carecían de comprensión contextual y generalización semántica (GeeksforGeeks, 2025).

Posteriormente, la aparición de los modelos neuronales supuso un avance importante. Las redes neuronales recurrentes (Recurrent Neural Networks, RNN) y sus variantes como las LSTM (Long Short-Term Memory) introdujeron la capacidad de procesar secuencias de palabras y mantener una representación del contexto a lo largo del texto. Gracias a estas arquitecturas, los sistemas de PLN pudieron modelar dependencias temporales y generar predicciones más coherentes dentro de una oración o párrafo.

El siguiente gran salto fue la introducción de la arquitectura Transformer propuesta por Vaswani et al. (2017). A diferencia de las redes recurrentes, los Transformers emplean mecanismos de atención, que permiten analizar simultáneamente todas las posiciones de una secuencia y capturar dependencias a largo plazo de manera más eficiente. Esta arquitectura se convirtió en la base de modelos modernos de lenguaje como BERT, GPT y T5, los cuales han revolucionado el campo al generar representaciones contextualizadas y producir texto con un nivel de coherencia similar al humano (Vaswani et al., 2017).

Esta transición marca el paso de un PLN basado en reglas y estadísticas hacia un PLN fundamentado en el aprendizaje profundo y contextualizado, en el que las representaciones vectoriales y los modelos de atención han reemplazado los enfoques tradicionales, permitiendo avances sustanciales en comprensión, traducción y generación de lenguaje natural.

2.2.4. ROL DEL NLP EN LOS AGENTES CONVERSACIONALES

El PLN constituye el eje funcional de los agentes conversacionales modernos. En las arquitecturas típicas de diálogo, la disciplina del PLN vincula los módulos de comprensión del lenguaje (NLU), gestión del estado o del diálogo (Dialogue Manager) y generación del lenguaje (NLG), de modo que el sistema pueda interpretar entradas en lenguaje natural y producir respuestas coherentes con el objetivo conversacional (Jurafsky & Martin, 2025).



En los sistemas de diálogo basados en PLN, cuando el usuario emite una utterance —es decir, una expresión en lenguaje natural, ya sea una frase, pregunta o comando— el NLU lleva a cabo tareas como la detección de intención, la extracción de entidades (por ejemplo, servicios, fechas o nombres) y la resolución de ambigüedades apoyada en el contexto y el estado del diálogo. Esta información se integra en una representación semántica que el gestor del diálogo utiliza para determinar la siguiente acción del sistema. Posteriormente, el LNG transforma dicha acción en una respuesta formulada en lenguaje humano, ajustando su contenido, registro y coherencia al historial conversacional y al objetivo de la tarea. La calidad de este proceso condiciona la naturalidad percibida y la eficacia comunicativa del agente.

La precisión de los componentes de PLN determina la coherencia del intercambio, la detección correcta de intenciones y el manejo adecuado de ambigüedades. Fallos en estos procesos producen respuestas fuera de contexto, malinterpretaciones y ciclos de reparación (repair turns) que degradan la experiencia conversacional (Jurafsky & Martin, 2025). Estas limitaciones en los enfoques tradicionales motivaron el desarrollo de arquitecturas más integradas y adaptables.

Los avances recientes en LLMs han intensificado el papel del PLN al integrar comprensión semántica profunda y generación contextualizada en un mismo modelo, incrementando la robustez ante la variación lingüística y favoreciendo la transferencia entre tareas. Estas mejoras han ampliado el espectro de dominios donde los agentes conversacionales pueden desplegarse de forma más fiable, contextualizada y adaptable, como en salud, educación o servicios públicos (Zhao et al., 2025).

2.3. MODELOS DE LENGUAJE Y LLMs

2.3.1. DEFINICIÓN GENERAL DE LOS LLMs

Los modelos de lenguaje son sistemas de inteligencia artificial diseñados para predecir la probabilidad de una secuencia de palabras, aprendiendo patrones estadísticos, sintácticos y semánticos a partir de grandes corpus de texto. Durante su entrenamiento, estos modelos



identifican regularidades del lenguaje natural, lo que les permite generar texto coherente, completar frases y realizar inferencias básicas sobre el significado (Jurafsky & Martin, 2025).

Los LLMs representan una evolución significativa de estos enfoques tradicionales. Se caracterizan por estar entrenados con volúmenes masivos de datos textuales y por utilizar redes neuronales de muy alta capacidad, que pueden alcanzar miles de millones de parámetros. Esta escala computacional y de datos permite que los LLMs capturen dependencias lingüísticas complejas, relaciones semánticas profundas e incluso fragmentos de conocimiento general del mundo, superando ampliamente el rendimiento de los modelos estadísticos o neuronales de menor tamaño (Zhao et al., 2025).

Gracias a esta capacidad para modelar el lenguaje de forma contextualizada y de alto nivel, los LLMs se han convertido en la base tecnológica de múltiples aplicaciones modernas. Entre ellas se encuentran sistemas de diálogo, traducción automática, resumen de textos, clasificación semántica, razonamiento asistido y diversas formas de asistencia virtual. Su versatilidad y desempeño han impulsado un cambio de paradigma en el procesamiento del lenguaje natural, ubicándolos como la tecnología central en la mayoría de los sistemas contemporáneos de comprensión y generación de lenguaje. (Zhao et al., 2025)

2.3.2. ARQUITECTURA TRANSFORMER

La aparición de la arquitectura Transformer, propuesta por Vaswani et al. (2017), representó un punto de inflexión en el desarrollo de los modelos de lenguaje actuales. Hasta ese momento, la mayoría de los sistemas utilizaban redes que procesan el texto palabra por palabra, siguiendo un orden estricto. Esta forma de trabajo era costosa y limitaba la capacidad de los modelos para entender relaciones entre palabras distantes.

El enfoque Transformer introdujo un mecanismo diferente y mucho más eficiente, basado exclusivamente en un principio llamado atención, que permite analizar todas las palabras de una frase al mismo tiempo. Esto mejoró la velocidad, la capacidad de comprender el contexto completo y la posibilidad de escalar los modelos a tamaños mucho mayores (Jurafsky & Martin, 2025).



La arquitectura se organiza en tres componentes principales que, combinados, explican su funcionamiento:

1. Mecanismo de atención.

Un componente fundamental de esta arquitectura es el mecanismo de atención, que permite al modelo determinar qué palabras resultan más relevantes para interpretar cada parte de una oración. Por ejemplo, en la frase “el paciente retiró su medicación”, el modelo puede identificar que “paciente” y “su” están relacionados, aun cuando no se encuentren adyacentes. Esta capacidad, denominada *self-attention*, elimina la dependencia del procesamiento estrictamente secuencial y facilita la comprensión de relaciones complejas dentro del lenguaje (Vaswani et al., 2017).

2. Codificadores (*encoders*).

El bloque de codificación recibe el texto de entrada y lo transforma en representaciones internas contextualizadas, donde cada token incorpora información tanto semántica como relacional respecto del resto de la oración. Estas representaciones sirven como base para las tareas posteriores de comprensión, clasificación o razonamiento (Jurafsky & Martin, 2025).

3. Decodificadores (*decoders*).

Los decodificadores utilizan esa información procesada para generar la salida del modelo: una respuesta, una traducción o una continuación del texto. Esta etapa combina la información de entrada con lo que el modelo ya ha producido hasta ese momento, asegurando respuestas coherentes y alineadas con el contexto global (Vaswani et al., 2017).

La combinación de estos elementos hizo posible realizar entrenamiento altamente paralelo, ya que el modelo no depende de procesar la secuencia palabra por palabra. Esta capacidad de paralelización, sumada a su eficiencia para capturar relaciones complejas, fue determinante para la aparición de los LLMs, cuyo tamaño y desempeño serían impracticables sin los principios introducidos por la arquitectura Transformer. (Vaswani et al., 2017).

2.3.3. MODELOS REPRESENTATIVOS

El desarrollo reciente del PLN ha estado fuertemente influido por una serie de modelos de lenguaje representativos que marcaron hitos en tareas de comprensión y generación de texto.



Entre los más relevantes se encuentran GPT, BERT, LLaMA y Mistral, cada uno asociado a avances distintos en capacidad, eficiencia u orientación hacia la investigación abierta.

En primer lugar, GPT (Generative Pre-trained Transformer), desarrollado por OpenAI, constituye una familia de modelos basada en la arquitectura Transformer y entrenada mediante generative pre-training, un enfoque que permite aprender a predecir la siguiente palabra en una secuencia (Radford et al., 2018). En sus versiones de gran escala, como GPT-3, estos modelos demostraron la capacidad de adaptarse a múltiples tareas sin reentrenamiento, únicamente mediante instrucciones textuales (prompting) o con unos pocos ejemplos proporcionados en el propio mensaje de entrada (few-shot learning) (Brown et al., 2020). Gracias a esta versatilidad, GPT se ha convertido en una herramienta ampliamente utilizada en aplicaciones de diálogo, redacción y asistencia automatizada.

Por otro lado, BERT (Bidirectional Encoder Representations from Transformers), introducido por Google, se centra principalmente en la comprensión profunda del lenguaje. A diferencia de GPT, BERT procesa el texto de manera bidireccional, es decir, interpreta cada palabra considerando simultáneamente el contexto previo y posterior. Además, utiliza únicamente el componente encoder del Transformer, lo cual le permite capturar con mayor precisión relaciones semánticas complejas. Este enfoque ha demostrado una eficacia notable en tareas como clasificación de texto, respuesta automática a preguntas y análisis semántico (Devlin et al., 2019).

En cuanto a los modelos abiertos, LLaMA (Large Language Model Meta AI), desarrollado por Meta, consiste en una familia de modelos de diversos tamaños diseñada específicamente para investigación abierta. Una de sus principales características es su alto rendimiento respecto a la cantidad de parámetros, lo que facilita su ejecución en entornos con recursos limitados sin comprometer las capacidades del modelo. Esto convirtió a LLaMA en una base ampliamente adoptada para proyectos académicos y variantes optimizadas (Touvron et al., 2023).

De forma complementaria, Mistral representa una línea reciente de modelos abiertos orientados a la eficiencia y la modularidad. En particular, Mistral 7B incorpora técnicas como grouped-query attention y sliding-window attention, que permiten procesar secuencias largas



de texto con menor costo computacional y mejorar la velocidad de inferencia. Aunque su tamaño es relativamente reducido, logra un rendimiento competitivo frente a modelos cerrados de mayor escala (Jiang et al., 2023), reforzando así la tendencia hacia arquitecturas compactas y altamente optimizadas.

Estos modelos han impulsado distintas dimensiones del avance en NLP: GPT potenció la generación fluida de texto; BERT transformó la comprensión contextual; LLaMA amplió el acceso a modelos potentes para la investigación abierta; y Mistral demostró la viabilidad de arquitecturas eficientes y escalables. La diversidad de estos enfoques ha sido fundamental para consolidar un ecosistema capaz de abordar tareas de comprensión, generación y adaptación contextual del lenguaje en una amplia variedad de dominios.

2.3.4. CAPACIDADES DE LOS LLMs

Los LLMs se distinguen por su capacidad para generar texto de forma contextual, manteniendo coherencia en el significado y continuidad temática a lo largo de varios turnos de conversación o párrafos. Gracias a su entrenamiento sobre enormes volúmenes de texto y al uso de la arquitectura Transformer, estos modelos pueden producir respuestas fluidas, relevantes y adaptadas al contexto de la interacción (Minaee et al., 2024; Zhao et al., 2025).

Una de las capacidades más relevantes de los LLMs es su habilidad para realizar razonamiento básico e inferencias a partir de la información proporcionada. Investigaciones recientes describen la aparición de habilidades emergentes como:

- Multi-step reasoning, donde el modelo puede encadenar varios pasos de razonamiento.
- Chain-of-thought, una forma de resolver problemas dividiéndolos en sub-tareas lógicas.
- El uso de conocimiento general del mundo para responder preguntas o tomar decisiones.

Estas capacidades permiten que los LLMs no solo generen texto, sino que también puedan justificar, relacionar y contextualizar la información en escenarios complejos (Minaee et al., 2024; Zhao et al., 2025).



Los LLMs también se utilizan de forma eficaz en una amplia variedad de tareas del lenguaje: clasificación semántica, extracción de información, generación de resúmenes, análisis de sentimientos y traducción multilingüe. Una de sus ventajas es que, al compartir una misma arquitectura basada en Transformers, un solo modelo puede abarcar tareas muy distintas sin requerir estructuras especializadas para cada caso, alcanzando frecuentemente un rendimiento comparable al de sistemas diseñados específicamente para una única función (Minaee et al., 2024).

La fortaleza más significativa de los LLMs es su capacidad de transferencia entre tareas. Un modelo entrenado de manera generalista puede ajustarse a nuevos dominios y problemáticas sin reentrenamiento exhaustivo. Esto se logra mediante técnicas como:

- Zero-shot learning, donde el modelo responde adecuadamente sin haber visto ejemplos previos.
- Few-shot learning, en el que basta incluir algunos ejemplos dentro del *prompt* para guiar su comportamiento.

Estos mecanismos, vinculados a fenómenos como in-context learning (aprender del contexto inmediato) y instruction following (seguir instrucciones en lenguaje natural), permiten que los LLMs se comporten como herramientas versátiles y reutilizables, capaces de adaptarse a múltiples aplicaciones con mínimos ajustes adicionales (Zhao et al., 2025; Minaee et al., 2024).

2.3.5. LIMITACIONES Y DESAFÍOS ACTUALES

A pesar de sus avances, los LLMs presentan una serie de limitaciones que deben considerarse para garantizar un uso seguro y responsable. Una de las más relevantes es la presencia de sesgos cognitivos, sociales y culturales. Estos modelos aprenden a partir de grandes volúmenes de texto obtenidos de internet y otras fuentes, donde pueden existir estereotipos, prejuicios o patrones discriminatorios. En consecuencia, los LLMs pueden reproducir e incluso amplificar dichos sesgos, generando respuestas injustas o desfavorables hacia determinados grupos. En ámbitos como la educación, la salud o la toma de decisiones, esto representa un riesgo para la equidad y la confiabilidad del sistema (Kasneci et al., 2023).



Otra limitación ampliamente documentada es la tendencia de los modelos a producir errores factuales o alucinaciones. Los LLMs pueden generar información incorrecta o no verificada que, sin embargo, es presentada como plausible y bien formulada en términos lingüísticos. Este fenómeno es especialmente problemático en contextos donde la precisión de la información es crítica, como en tareas educativas, sanitarias o jurídicas, ya que puede inducir a conclusiones erróneas o generar desinformación (Kasneci et al., 2023; Zhao et al., 2025).

La falta de interpretabilidad constituye también un desafío importante. Debido a la complejidad de su arquitectura y a la naturaleza estadística de sus procesos internos, los LLMs suelen funcionar como sistemas de tipo “caja negra”, en los que resulta difícil comprender cómo se generan determinadas respuestas o identificar el origen de posibles errores. Esta falta de transparencia limita la posibilidad de auditar su comportamiento, dificulta la detección de sesgos y complica la toma de decisiones informada sobre su uso (Zhao et al., 2025).

A nivel operativo, los LLMs presentan además un alto coste computacional. Su entrenamiento y funcionamiento requieren infraestructuras especializadas —como GPU o TPU— y un consumo energético significativo. Esto aumenta los costos económicos y restringe el acceso a organizaciones con capacidad de cómputo avanzada, lo que plantea desafíos relacionados con la escalabilidad, la sostenibilidad y la democratización del acceso a estas tecnologías (Zhao et al., 2025).

Finalmente, se identifican importantes desafíos éticos y de gobernanza. Entre ellos se destacan la falta de transparencia respecto a los datos utilizados para entrenar los modelos, la capacidad de generar contenido sintético difícil de distinguir del producido por humanos y el potencial uso malintencionado para la creación de desinformación, contenidos dañinos o spam. Estas cuestiones demandan supervisión humana, pautas claras de uso responsable y mecanismos de gobernanza que acompañen la adopción de estas tecnologías, especialmente en contextos sensibles como la educación y la salud (Kasneci et al., 2023).

Estas limitaciones evidencian que, si bien los LLMs representan un avance significativo en el procesamiento del lenguaje natural, su utilización requiere criterios éticos explícitos,



supervisión adecuada, transparencia y marcos de gobernanza que permitan maximizar sus beneficios y mitigar sus riesgos.

2.3.6. RELEVANCIA EN EL CONTEXTO DE LOS AGENTES CONVERSACIONALES

Los LLMs se han convertido en el componente central de los agentes conversacionales modernos, principalmente porque permiten integrar en un solo modelo tareas que antes requerían varios módulos separados. Tradicionalmente, los sistemas de diálogo utilizaban componentes independientes para la comprensión del lenguaje, la gestión del diálogo y la generación de respuestas, los cuales debían configurarse y ajustarse de forma manual. La adopción de modelos de lenguaje entrenados a gran escala ha permitido unificar estas funciones dentro de una misma arquitectura, marcando una evolución significativa respecto de los enfoques modulares clásicos (Nsaif et al., 2024).

Con los LLMs, estas funciones se realizan de manera unificada dentro del mismo modelo, que es capaz de:

- Interpretar la intención del usuario.
- Mantener coherencia entre los distintos turnos de conversación.
- Generar respuestas claras, fluidas y ajustadas al contexto.

Esta integración supone un cambio importante respecto de los sistemas tradicionales basados en reglas o en flujos rígidos de conversación. En estos enfoques anteriores, las respuestas dependían de patrones predefinidos y de configuraciones estáticas, lo que limitaba la naturalidad de las interacciones y la capacidad del sistema para adaptarse a nuevas situaciones.

En contraste, los LLMs pueden entender variaciones del lenguaje, manejar expresiones ambiguas, inferir significados implícitos y adaptar su estilo comunicativo según el contexto o el perfil del usuario. Esto permite que la interacción sea más flexible, más cercana al lenguaje humano y menos dependiente de estructuras conversacionales preprogramadas.

Gracias a estas capacidades, los LLMs amplían significativamente el alcance de los agentes conversacionales, permitiendo su aplicación en escenarios más complejos:



- Acompañamiento y tutoría educativa.
- Asistencia técnica especializada.
- Análisis y síntesis de información.
- Apoyo en procesos de toma de decisiones.
- Interacción en contextos institucionales, como salud, gobierno o servicios públicos.

La relevancia de los LLMs en los agentes conversacionales radica en su capacidad para transformarlos en sistemas verdaderamente inteligentes, capaces de sostener diálogos más ricos, naturales y contextualizados, lo que marca una diferencia sustancial respecto de generaciones anteriores de chatbots.

2.4. MODEL CONTEXT PROTOCOL

2.4.1. DEFINICIÓN GENERAL DEL MODEL CONTEXT PROTOCOL

El MCP es un protocolo abierto diseñado para que los modelos de lenguaje puedan interactuar con herramientas y sistemas externos —como APIs, bases de datos, servicios corporativos o repositorios de información— de manera estandarizada, estructurada y segura. Propuesto por Anthropic en noviembre de 2024, el MCP funciona como un puente que permite a los modelos no limitarse a generar texto a partir de su entrenamiento, sino también acceder a datos actualizados y ejecutar acciones reales sobre sistemas externos cuando la aplicación lo requiere (Anthropic, 2024).

El propósito central del protocolo es organizar y simplificar el flujo de información entre un modelo de lenguaje y los recursos externos que necesita para realizar una tarea. Antes de su aparición, cada integración entre un modelo y una fuente de datos debía desarrollarse de manera personalizada, generando una gran complejidad cuando se combinaban múltiples modelos con múltiples sistemas externos. El MCP elimina esta multiplicidad de conectores ad hoc al proponer una forma unificada y común de comunicación, reduciendo la carga de desarrollo y facilitando la interoperabilidad (Google Cloud, 2025; Model Context Protocol, 2025).

2.4.2. OBJETIVO Y MOTIVACIÓN DEL PROTOCOLO



El MCP surge como respuesta a la necesidad de normalizar cómo los modelos de lenguaje acceden al contexto externo, ya sea información almacenada en sistemas internos, archivos corporativos, servicios web o módulos especializados. En modelos previos, la interacción con estos recursos dependía de integraciones hechas a medida, lo que generaba una estructura de complejidad $N \times M$ (distintos modelos \times distintos sistemas) difícil de mantener y escalar. (Model Context Protocol, 2025).

Al establecer un marco común, el MCP permite que los agentes basados en LLMs puedan:

- Consultar datos.
- Invocar funciones externas.
- Utilizar herramientas específicas.
- Generar respuestas fundamentadas en información actualizada.

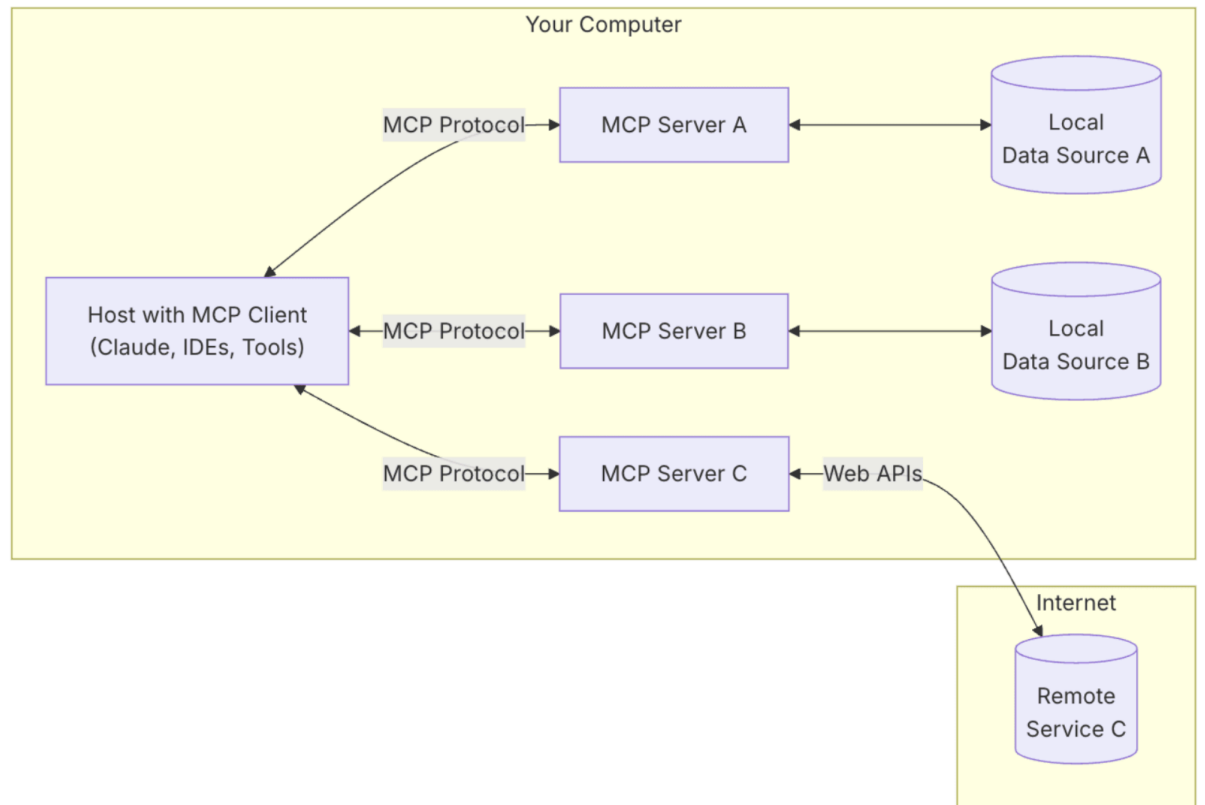
sin estar limitados al conocimiento estático del modelo ni depender de interfaces rígidas.

2.4.3. ARQUITECTURA Y COMPONENTES PRINCIPALES

El MCP adopta una arquitectura basada en un modelo cliente–servidor que permite a los modelos de lenguaje interactuar con herramientas, recursos y sistemas externos mediante un conjunto de reglas y estructuras estandarizadas. Su objetivo principal es proporcionar un mecanismo uniforme para ampliar las capacidades de los agentes conversacionales sin necesidad de desarrollar integraciones ad hoc para cada aplicación. De acuerdo con la documentación oficial, el protocolo define con precisión los roles de los componentes involucrados, los tipos de mensajes intercambiados y el flujo operativo de la interacción (Model Context Protocol, 2025).

Figura 3

Arquitectura general del MCP.



Nota. Adaptado de *Model Context Protocol* (2025).

Servidor MCP

El servidor MCP es el componente responsable de exponer las capacidades que pueden ser utilizadas por el modelo de lenguaje. Estas capacidades incluyen herramientas (tools), recursos (resources) y prompts reutilizables. Desde el punto de vista funcional, el servidor actúa como puente entre el modelo y el entorno externo, permitiendo la ejecución de acciones concretas tales como:

- Consultas a bases de datos.
- Llamadas a servicios web.
- Lectura o escritura de archivos.



- Interacción con sistemas corporativos o APIs internas.

El servidor recibe solicitudes formuladas según el protocolo —normalmente mediante JSON-RPC sobre WebSockets—, ejecuta la operación correspondiente y retorna una respuesta estandarizada. Además, proporciona mecanismos de descubrimiento automático de capacidades (por ejemplo, `tools/list` o `resources/list`), lo que favorece la interoperabilidad y la portabilidad entre distintas arquitecturas y entornos de ejecución (Model Context Protocol, 2025).

Cliente LLM

El cliente LLM corresponde al componente situado del lado del modelo de lenguaje. Este integra internamente el runtime MCP, una capa de software encargada de implementar el protocolo, gestionar la conexión con el servidor, formatear las solicitudes y procesar las respuestas.

A diferencia del servidor —que es un sistema externo y autónomo—, el cliente MCP no existe como módulo independiente. En su lugar, forma parte del entorno del modelo y es proporcionado por cada proveedor (OpenAI, Anthropic, Meta, implementaciones open-source, entre otros). Esta integración asegura compatibilidad nativa y evita que los desarrolladores deban crear adaptaciones específicas para cada modelo.

Operativamente, el cliente cumple funciones esenciales como:

- Traducir las intenciones del modelo en solicitudes MCP estructuradas.
- Enviar y recibir mensajes mediante el protocolo.
- Convertir las respuestas del servidor en contexto utilizable por el modelo durante la generación subsiguiente.
- Gestionar el estado conversacional, incluyendo flujos de contexto y resultados previos.

El cliente actúa como un intérprete entre el modelo de lenguaje y el servidor MCP, permitiendo la ejecución segura y controlada de operaciones externas sin modificar la arquitectura interna del modelo (Anthropic, 2024).



A diferencia del servidor —que es un componente independiente— el cliente MCP no existe como módulo autónomo, sino que está integrado dentro del entorno del modelo. Esto permite que cada proveedor de modelos (OpenAI, Anthropic, Meta o implementaciones open-source) incluya su propia versión del runtime, asegurando compatibilidad sin necesidad de desarrollar integraciones específicas para cada modelo o aplicación.

En términos operativos, el cliente LLM funciona como un intérprete entre el modelo y el servidor:

- Convierte las intenciones del modelo en solicitudes MCP estructuradas.
- Transforma la respuesta del servidor en contexto utilizable por el modelo.

Esta dinámica posibilita que los LLMs amplíen sus capacidades con integraciones dinámicas y seguras hacia servicios externos sin modificar su arquitectura interna.(Model Context Protocol, 2025

Herramientas (tools)

Las tools son definiciones estructuradas que especifican acciones ejecutables por el agente. Cada herramienta representa una operación concreta y encapsula la lógica necesaria para interactuar con un sistema externo. Su estructura incluye:

- Un identificador.
- Una descripción.
- Un esquema de parámetros (normalmente en JSON Schema).
- Un esquema de respuesta.
- La implementación del lado del servidor.

Entre las operaciones típicas se encuentran la ejecución de scripts, el acceso a APIs, la manipulación de datos o la consulta de servicios corporativos. Esta abstracción permite agregar o modificar capacidades sin alterar el funcionamiento principal del modelo de lenguaje, promoviendo modularidad y escalabilidad (Model Context Protocol, 2025). En este sentido, las tools funcionan como componentes plug-and-play que pueden ser utilizadas dinámicamente por el agente.



Recursos (Resources)

Los recursos constituyen otro tipo de capacidad ofrecida por el servidor MCP. A diferencia de las herramientas —que ejecutan acciones—, los recursos exponen información accesible de manera estática, como documentos, configuraciones, fragmentos de código o datos estructurados. El modelo puede solicitar la lectura de un recurso cuando necesite consultar información sin ejecutar un procedimiento activo.

Este mecanismo permite:

- Evitar duplicación de información.
- Centralizar datos disponibles para el agente.
- Facilitar que el LLM opere sobre bases de conocimiento locales sin depender exclusivamente de su entrenamiento (Model Context Protocol, 2025).

Prompts

Los **prompts** son plantillas reutilizables definidas en el servidor que pueden ser invocadas por el modelo a través del protocolo. Su finalidad es estandarizar instrucciones, estructuras conversacionales o formatos de salida, fomentando la consistencia entre diferentes agentes o componentes. Los prompts pueden incluir variables parametrizables, lo que permite adaptarlos a distintos contextos sin perder uniformidad semántica. (Model Context Protocol, 2025).

Flujos de contexto (context flows)

Los context flows permiten mantener información relevante a lo largo de una sesión conversacional. No constituyen memoria permanente, sino un mecanismo de estado transaccional: conservan datos temporales necesarios para la continuidad de un proceso, trámite o interacción compleja.

Mediante este sistema, el agente puede disponer de elementos como:

- Resultados obtenidos previamente por una herramienta.
- Parámetros proporcionados por el usuario.



- Estados intermedios asociados a procesos multietapa.

Esto mejora la coherencia del diálogo y permite que el agente actúe de forma contextualizada en lugar de procesar cada mensaje de manera aislada (Model Context Protocol, 2025).

2.4.4. FUNCIONAMIENTO OPERATIVO DEL MCP

Desde una perspectiva operativa, el MCP define un flujo estructurado que regula cómo un modelo de lenguaje puede interactuar con herramientas y sistemas externos. Este proceso se organiza bajo una arquitectura cliente–servidor, donde el entorno que aloja al modelo de lenguaje incluye un cliente MCP, el cual se comunica con uno o varios servidores MCP que exponen herramientas y recursos (Anthropic, 2024; Google Cloud, 2025).

El funcionamiento puede describirse en las siguientes etapas:

1. Identificación de la necesidad de información externa

A partir de la instrucción del usuario, el modelo de lenguaje detecta que necesita consultar datos externos o ejecutar una acción específica —por ejemplo, acceder a una base de datos, invocar un servicio web o recuperar un archivo.

Esta decisión se expresa mediante una invocación de herramienta (tool call), que el propio LLM produce cuando identifica que la respuesta no puede generarse únicamente con su conocimiento interno (Google Cloud, 2025).

2. Envío de la solicitud al servidor MCP

El cliente MCP, integrado dentro del cliente LLM, toma esa invocación y la convierte en una solicitud formal siguiendo el estándar del protocolo.

La solicitud incluye:

- El nombre de la herramienta a ejecutar.
- Los parámetros necesarios.
- El contexto requerido.



Luego, esta solicitud se envía al servidor MCP correspondiente (Anthropic, 2024; Model Context Protocol, 2025).

3. Validación y ejecución en el servidor

El servidor MCP recibe la solicitud y realiza una serie de validaciones:

- Verifica que la herramienta exista.
- Comprueba que los parámetros cumplan el esquema definido.
- Aplica los mecanismos de autorización y control de acceso, como OAuth 2.1 cuando corresponde.

Si todo es válido, el servidor ejecuta la acción sobre el sistema externo —API, base de datos o servicio empresarial— y genera una respuesta estructurada con los resultados (Model Context Protocol, 2025).

4. Devolución de los datos al LLM

La respuesta del servidor MCP vuelve al cliente MCP, que la traduce a un formato que el modelo de lenguaje pueda incorporar como contexto.

Con esta información, el LLM puede:

- Generar una respuesta final más completa.
- Actualizar el estado de la conversación.
- Iniciar nuevas acciones dentro de un flujo de trabajo más amplio (Anthropic, 2024; Google Cloud, 2025).

5. Registro estructurado de las operaciones

El protocolo MCP también define mecanismos de registro estructurado (logging) que permiten que el servidor envíe mensajes al cliente con detalles de las operaciones realizadas: nivel de severidad, datos asociados, advertencias o errores.

Este registro contribuye a la auditoría, la depuración y el análisis de seguridad, ya que deja trazabilidad completa de todas las interacciones (Model Context Protocol, 2025).



Este flujo garantiza que cualquier interacción entre un LLM y sistemas externos siga reglas explícitas y controladas:

- Las herramientas están definidas con esquemas claros.
- El acceso a recursos sensibles pasa por mecanismos de autorización estándar.
- Cada operación queda registrada para asegurar transparencia y trazabilidad.

En conjunto, MCP proporciona un marco seguro y estructurado que permite ampliar las capacidades de un agente conversacional sin perder control sobre la seguridad, la gobernanza y el uso responsable de los datos externos.

2.4.5. VENTAJAS DEL USO DE MCP

El uso del MCP aporta una serie de beneficios que fortalecen la integración de los modelos de lenguaje en entornos reales y complejos. Según la documentación oficial de la especificación, estos beneficios abarcan dimensiones técnicas, operativas y de gobernanza (Model Context Protocol, 2025; Anthropic, 2024).

Interoperabilidad: El MCP permite que un mismo modelo de lenguaje se conecte a múltiples sistemas, herramientas o servicios externos sin necesidad de crear integraciones específicas para cada caso. Al definir un estándar común para la comunicación entre LLMs y recursos externos, el protocolo simplifica el desarrollo y favorece una interoperabilidad amplia, sostenible y reutilizable (Model Context Protocol, 2025).

Seguridad: El protocolo incorpora mecanismos explícitos de permisos, validaciones y control de parámetros, lo que garantiza que cada solicitud se revise antes de ejecutarse. Este enfoque minimiza riesgos relacionados con el acceso a información sensible y contribuye a un uso más seguro de sistemas corporativos o institucionales (Anthropic, 2024; Model Context Protocol, 2025).

Trazabilidad: El MCP incluye sistemas de registro estructurado (logging) que documentan cada operación realizada. Esta trazabilidad facilita tareas de auditoría, monitoreo y análisis de incidentes, especialmente en contextos donde la transparencia operativa es crítica, como salud, gobierno o servicios empresariales (Model Context Protocol, 2025).



Modularidad: Cada herramienta se define como un componente independiente, lo que permite agregar, modificar o eliminar funcionalidades sin alterar el funcionamiento del modelo de lenguaje. Esta modularidad reduce la necesidad de integraciones ad hoc y facilita que el sistema evolucione de manera ordenada y mantenible (Anthropic, 2024; Model Context Protocol, 2025).

Escalabilidad: Gracias a su diseño estandarizado y extensible, el MCP facilita la incorporación de nuevos servicios, bases de datos o sistemas externos a medida que la organización crece o diversifica sus necesidades. Esto permite escalar el ecosistema del agente conversacional sin rediseños profundos ni desarrollos costosos (Model Context Protocol, 2025).

Estas ventajas consolidan al MCP como un elemento clave para la construcción de agentes conversacionales avanzados, permitiendo que los modelos de lenguaje operen de forma segura, flexible y alineada con las exigencias de entornos reales, especialmente aquellos que requieren control, trazabilidad y gobernanza.

2.4.6. COMPARACIÓN CON ENFOQUES PREVIOS

La introducción del MCP representa un cambio significativo frente a los métodos utilizados anteriormente para permitir que los modelos de lenguaje interactúen con sistemas externos. Las soluciones previas abordaban este problema de manera parcial o fragmentada, mientras que MCP establece un protocolo unificado, formal y estandarizado que redefine la integración entre LLMs y recursos externos, reduciendo la complejidad y aportando coherencia al ecosistema (Model Context Protocol, 2025).

En los primeros enfoques, las integraciones solían implementarse mediante Plugin APIs, es decir, APIs específicas creadas para conectar un modelo de lenguaje con un servicio particular. Cada integración debía desarrollarse de forma independiente, con sus propios formatos, permisos y validaciones. Esta falta de estandarización hacía que el mantenimiento y la escalabilidad resultaran complejos, especialmente cuando se necesitaba integrar múltiples servicios. Frente a este escenario, MCP introduce un conjunto de reglas uniformes que



eliminan los conectores ad hoc y permiten simplificar el desarrollo a largo plazo (Model Context Protocol, 2025).

Otro enfoque previo fue el function calling, utilizado ampliamente para permitir que los modelos invocaran funciones definidas por el desarrollador. Si bien esto supuso un avance importante, estas funciones estaban limitadas a un conjunto local y estático, sin ofrecer un ecosistema interoperable ni un servidor encargado de exponer herramientas de forma estandarizada. MCP amplía este concepto mediante una arquitectura cliente–servidor, esquemas formales y mecanismos de descubrimiento dinámico de herramientas, habilitando integraciones más escalables y consistentes (OpenAI, 2024; Model Context Protocol, 2025).

También se propusieron métodos basados en prompting avanzado, como ReAct, que combina razonamiento y acción para permitir que los LLMs decidan cuándo utilizar herramientas y cómo interpretar sus resultados (Yao et al., 2022). Sin embargo, este enfoque dependía exclusivamente de instrucciones textuales y no proporcionaba un protocolo estructurado, mecanismos de seguridad ni validación formal de parámetros. El MCP supera estas limitaciones al incorporar mensajes tipificados, validaciones explícitas y controles de ejecución que aumentan la confiabilidad del sistema (Model Context Protocol, 2025).

Finalmente, frameworks como LangChain, LlamaIndex o Haystack introdujeron abstracciones útiles para la orquestación de agentes, herramientas y documentos, pero operan a nivel de software y no como un estándar universal. Su alcance está limitado a sus propias implementaciones, mientras que MCP propone un protocolo independiente del framework o proveedor, unificando el modo en que los modelos de lenguaje acceden a recursos externos (LangChain, s.f.; LlamaIndex, s.f.; Haystack, s.f.).

2.5. SEGURIDAD, PRIVACIDAD Y ÉTICA EN SISTEMAS DE SALUD

2.5.1. MARCO LEGAL Y NORMATIVO

En Argentina, el tratamiento de datos personales en el ámbito sanitario está regulado por la Ley de Protección de Datos Personales N.º 25.326, que clasifica la información de salud como una categoría especialmente protegida. La norma exige que cualquier archivo, registro o



banco de datos que contenga este tipo de información garantice la privacidad, la intimidad y el acceso adecuado por parte del titular de los datos (Ley 25.326, 2000).

La ley establece principios rectores fundamentales para asegurar un tratamiento responsable de los datos sensibles. Entre ellos se destacan: el consentimiento informado para su uso, la finalidad específica del tratamiento, la proporcionalidad en la recolección, la calidad y exactitud de los datos, y los principios de seguridad y confidencialidad, que obligan a implementar medidas técnicas y organizativas para resguardar la información (Dirección Nacional de Protección de Datos Personales, 2005).

En el plano internacional existen dos referentes normativos de gran impacto en los sistemas digitales de salud. En Estados Unidos, la Health Insurance Portability and Accountability Act (HIPAA) establece estándares estrictos para la protección de la información médica protegida (Protected Health Information, PHI), regulando la privacidad, la seguridad de los registros electrónicos y los procedimientos asociados a la notificación de incidentes (U.S. Department of Health & Human Services, 1996).

Por su parte, en Europa, el Reglamento General de Protección de Datos (GDPR) refuerza la protección de los datos personales mediante obligaciones como la minimización de datos, la responsabilidad proactiva (*accountability*), los derechos del titular y la transparencia en el tratamiento, principios especialmente relevantes en plataformas digitales que gestionan información sensible (European Commission, 2024).

Tanto las normativas nacionales como los marcos internacionales influyen de manera decisiva en el diseño y la operación de soluciones digitales aplicadas a la salud. En consecuencia, cualquier plataforma que utilice inteligencia artificial en contextos sanitarios debe alinearse con estos requisitos legales para garantizar un uso responsable y seguro de los datos.

2.5.2. RIESGOS ASOCIADOS A LA DIGITALIZACIÓN Y USO DE IA EN SALUD

La incorporación de sistemas digitales e inteligencia artificial en entornos sanitarios ha impulsado mejoras en la gestión clínica y en la eficiencia de numerosos procesos. No obstante, también introduce riesgos relevantes que requieren atención técnica, ética y regulatoria. Entre ellos, uno de los desafíos más críticos es la posible fuga o exposición de



información sensible, especialmente porque los sistemas de salud manejan datos clínicos altamente protegidos que pueden verse comprometidos por brechas de seguridad, accesos no autorizados o prácticas deficientes de almacenamiento y transmisión. Tanto la normativa argentina como los marcos regulatorios internacionales subrayan la obligación de aplicar medidas estrictas de protección y seguridad para prevenir estos incidentes (Ley 25.326, 2000; U.S. Department of Health & Human Services, 1996; European Commission, 2024).

Otro riesgo relevante es la presencia de sesgos algorítmicos. Cuando los modelos se entrenan con datos incompletos o desequilibrados, pueden reproducir inequidades existentes y afectar la precisión de los resultados clínicos, especialmente en poblaciones vulnerables. Este problema es ampliamente reconocido por los marcos regulatorios, que exigen garantizar proporcionalidad, calidad de datos y procesos de evaluación continua para evitar discriminaciones (Dirección Nacional de Protección de Datos Personales, 2005; European Commission, 2024).

La digitalización también puede habilitar decisiones automáticas sin supervisión adecuada, lo que constituye un riesgo crítico. Un sistema basado en IA podría generar recomendaciones incorrectas, ambiguas o potencialmente peligrosas si no interviene un profesional responsable que valide la información. Tanto HIPAA como el GDPR subrayan la importancia de mantener controles humanos en procesos que afecten derechos o la salud del paciente (U.S. Department of Health & Human Services, 1996; European Commission, 2024).

A esto se suma la falta de explicabilidad de muchos modelos avanzados —incluidos los LLMs— cuya complejidad dificulta auditar sus procesos internos y comprender cómo llegan a ciertas conclusiones. Esta falta de transparencia afecta la trazabilidad, limita la capacidad de corregir errores y genera incertidumbre respecto de la responsabilidad jurídica en caso de fallos.(European Commission, 2024).

2.5.3. IMPORTANCIA DE LA TRANSPARENCIA Y LA CONFIANZA EN LA IA APLICADA A SALUD

La transparencia se vuelve un componente fundamental para asegurar la aceptabilidad social de los sistemas de IA. Esto implica ofrecer claridad respecto de las fuentes de datos utilizadas, los mecanismos de razonamiento del modelo, sus limitaciones operativas y el alcance real de



sus recomendaciones. La opacidad algorítmica —especialmente en modelos complejos como los LLMs— puede dificultar la auditoría, limitar la trazabilidad y generar incertidumbre sobre la fiabilidad del sistema, lo que afecta directamente la confianza de los usuarios y dificulta su implementación en entornos clínicos (World Health Organization, 2021a; European Commission, 2024).

La transparencia está estrechamente vinculada con principios éticos fundamentales, como la responsabilidad, la explicabilidad, la justicia y la no maleficencia. La responsabilidad exige que exista un marco claro de rendición de cuentas; la explicabilidad permite comprender y justificar las decisiones algorítmicas; la justicia busca evitar discriminaciones y sesgos que afecten a poblaciones vulnerables; y la no maleficencia obliga a reducir riesgos potenciales y garantizar que los sistemas no generen daño (World Health Organization, 2021a).

Asimismo, la confianza se construye mediante un conjunto de prácticas que trascienden la dimensión técnica: medidas robustas de seguridad, validaciones clínicas rigurosas, evidencia empírica que respalde el desempeño del sistema y mecanismos de supervisión continua que aseguren su funcionamiento correcto y actualizado. Las normativas mencionadas requieren explícitamente medidas de control, auditoría y protección frente a usos indebidos o fallos operativos, elementos clave para consolidar la confiabilidad del ecosistema digital en salud (European Commission, 2024).

2.6. PERSPECTIVAS FUTURAS DE LOS AGENTES CONVERSACIONALES EN SALUD

2.6.1 INTEGRACIÓN MULTIMODAL (VOZ, IMAGEN Y TEXTO)

La evolución de los agentes conversacionales en el ámbito sanitario avanza hacia sistemas capaces de procesar múltiples modalidades de entrada y salida, entre ellas voz, imágenes médicas y texto. Esta capacidad multimodal amplía de manera significativa las posibilidades de la salud digital, permitiendo que la interacción con el sistema sea más natural, accesible y adaptada a las necesidades de cada usuario. Por ejemplo, el uso de entrada por voz facilita la comunicación para personas con dificultades para escribir o con menor alfabetización digital, mientras que la interpretación de imágenes médicas permite enriquecer el diálogo clínico con información visual relevante (Colakoglu et al., 2025; Saab & Freyberg, 2025).



En contextos de salud, poder interpretar imágenes —como estudios diagnósticos, fotografías de lesiones o credenciales médicas— y ofrecer interacción por voz incrementa la inclusión y reduce barreras de acceso para pacientes con distintos perfiles. Asimismo, la combinación de texto e imagen permite que los agentes presenten información visual complementaria, como explicaciones sobre tratamientos, coberturas de seguros o guías paso a paso para completar trámites clínicos. Estas capacidades amplían el alcance y la eficacia del sistema al integrar modalidades que enriquecen la comprensión del contexto y fortalecen la calidad de las respuestas, tal como se observa en los modelos multimodales contemporáneos que combinan visión y lenguaje para mejorar tareas de interpretación y asistencia (Lin et al., 2025).

2.6.2. PERSONALIZACIÓN AVANZADA MEDIANTE PERFILES DE USUARIO

La próxima generación de agentes conversacionales en el ámbito sanitario se orienta hacia sistemas capaces de brindar personalización avanzada a partir de perfiles dinámicos de usuario. Esto implica que los agentes puedan adaptarse progresivamente a la historia de interacción del paciente, a sus preferencias lingüísticas, a consultas recurrentes y a necesidades específicas de salud, lo cual permite ofrecer respuestas más pertinentes y una experiencia comunicativa más intuitiva. Estudios recientes muestran que las intervenciones digitales personalizadas mejoran la adherencia al tratamiento e incrementan la eficacia de la gestión de enfermedades crónicas (Lu et al., 2025).

Esta personalización abre oportunidades para mejorar la adherencia terapéutica, ya que los sistemas pueden generar recordatorios adaptados al comportamiento del paciente, reforzar medidas preventivas o sugerir recomendaciones ajustadas a su perfil clínico y hábitos de salud. De hecho, el análisis sistemático de estrategias de tecnologías digitales para fomentar el autocuidado indica que la personalización, junto con recordatorios, feedback adaptado y gamificación, contribuye significativamente a la mejora de los resultados de salud (Lu et al., 2025).

No obstante, estas capacidades avanzadas demandan estándares elevados de privacidad, transparencia y seguridad. Al manejar datos clínicos sensibles —incluyendo historiales, preferencias de salud y patrones de comportamiento—, es necesario asegurar que se cumplan



los principios normativos de protección de datos presentes en la legislación argentina, estadounidense y europea (Ley N.º 25.326, 2000; U.S. Department of Health & Human Services, 1996; European Commission, 2024). Asimismo, la capacidad de adaptar servicios según el usuario exige que quede claro qué datos se recogen, con qué finalidad, cómo se utilizan y que el usuario esté informado y ejerza control sobre sus datos.

2.6.3. INTEROPERABILIDAD CON SISTEMAS DE HISTORIA CLÍNICA ELECTRÓNICA (HCE)

Una de las líneas de avance más relevantes en el uso de agentes conversacionales en salud es su integración con los sistemas de Historia Clínica Electrónica (HCE). Esta integración permite que el agente acceda a información clínica relevante —como consultas previas, resultados de estudios o antecedentes médicos— y la utilice para generar interacciones más contextualizadas y personalizadas. Cuando el agente se conecta a una HCE interoperable, puede recuperar datos clínicos actualizados del paciente y sostener un diálogo informado que complementa la atención profesional (Li et al., 2022).

La interoperabilidad abre la puerta a una variedad de casos de uso avanzados, entre ellos el seguimiento remoto de pacientes, la revisión automatizada de alertas o valores críticos, el acompañamiento en procesos de autocuidado mediante interacciones periódicas, y la generación de recordatorios personalizados sobre estudios pendientes, medicación o tratamientos vigentes. Todos estos servicios dependen de la integración fluida entre sistemas y de la disponibilidad de información clínica consistente, actualizada y accesible en tiempo real.

Sin embargo, la implementación de estas capacidades implica desafíos importantes. La interoperabilidad médica requiere que datos clínicos sensibles circulen entre múltiples plataformas, lo que demanda medidas estrictas de seguridad, auditoría y cumplimiento normativo. Esto incluye controles de acceso robustos, encriptación, trazabilidad de operaciones y conformidad con los estándares de protección de datos aplicables. La literatura enfatiza que, aunque la interoperabilidad de las HCE tiene un gran potencial para mejorar la coordinación y eficiencia de la atención, también presenta barreras técnicas, organizativas y de seguridad que deben gestionarse cuidadosamente para evitar riesgos y garantizar un uso responsable (Li et al., 2022; Oracle, 2024).



2.6.4. POTENCIAL DEL MCP EN ECOSISTEMAS DE IA MÉDICA DISTRIBUIDA

El MCP posee un potencial significativo para convertirse en un estándar clave dentro de los ecosistemas de inteligencia artificial aplicada a la salud, especialmente en entornos donde distintos agentes, herramientas y fuentes de datos deben interactuar de manera coordinada. El sector sanitario suele operar con sistemas heterogéneos —como módulos administrativos, plataformas de turnos, motores de prestaciones, sistemas de facturación y repositorios de HCE—, lo que hace indispensable contar con un protocolo unificado que estructure estas interacciones (Anthropic, 2024; Model Context Protocol, 2025).

En este escenario, MCP permitiría integrar módulos especializados de forma modular y segura. Por ejemplo, un agente conversacional podría consultar un sistema de turnos, verificar coberturas en un módulo de prestaciones, acceder a información clínica en una HCE o iniciar un trámite administrativo, todo mediante un conjunto de herramientas estandarizadas expuestas por servidores MCP. Este enfoque elimina integraciones ad hoc y asegura que las acciones del modelo se realicen bajo reglas claras, auditables y compatibles con las normativas de protección de datos (Model Context Protocol, 2025).

Además, el protocolo posibilita que un LLM actúe como un orquestador seguro de procesos médico-administrativos. Gracias a su arquitectura cliente-servidor y a su esquema explícito de permisos, validación y registro, el MCP garantiza que cada operación —desde consultar un archivo hasta ejecutar una acción transaccional— se realice bajo un marco de control y trazabilidad. Esto es especialmente relevante en el ámbito sanitario, donde las operaciones deben ser verificables, justificadas y alineadas con criterios éticos y legales.

En ecosistemas distribuidos, MCP facilita la interoperabilidad entre sistemas heterogéneos, permitiendo que múltiples servicios interactúen con un LLM sin necesidad de unificar tecnologías, lenguajes o arquitecturas. Esta capacidad promueve:

- Flexibilidad, al permitir incorporar nuevas herramientas sin rediseñar los sistemas centrales.
- Trazabilidad, mediante registros estructurados de cada acción.



- Modularidad, gracias a herramientas independientes que pueden agregarse o reemplazarse según el caso de uso (Anthropic, 2024; Model Context Protocol, 2025).

CAPÍTULO III

.

Desarrollo de software



3.1 INTRODUCCIÓN

Este capítulo describe el proceso de construcción del prototipo desarrollado para este proyecto, abarcando desde la identificación de los requerimientos y la estructura general del sistema hasta la implementación de sus componentes principales y la integración del agente conversacional.

El desarrollo del prototipo combinó decisiones estructurales tomadas a partir del análisis inicial con ciclos de refinamiento técnico surgidos durante las pruebas de integración y la experimentación con distintas herramientas. Estas iteraciones permitieron ajustar aspectos específicos de la lógica conversacional, de la estructura de datos y de la interacción entre componentes, manteniendo siempre el alcance funcional definido en las primeras etapas del proyecto.

El capítulo se organiza en secciones que presentan:

- La identificación y modelado de los procesos institucionales utilizados como base del prototipo.
- El diseño conceptual y tecnológico del sistema.
- La implementación del backend, el frontend y el agente conversacional.
- La integración de los componentes.
- La validación del prototipo.
- Las iteraciones técnicas que contribuyeron a mejorar la coherencia operativa del prototipo.

De este modo, el capítulo ofrece una visión completa y estructurada del proceso de desarrollo, resaltando tanto los elementos planificados como los ajustes introducidos para optimizar el funcionamiento del sistema en su versión final.

3.2 IDENTIFICACIÓN Y MODELADO DE PROCESOS

Esta sección describe el proceso de identificación, selección y modelado de los trámites institucionales utilizados como base para el prototipo. Para ello se estableció el alcance funcional del sistema, se determinaron los procesos incluidos y excluidos, se relevaron



fuentes documentales y se incorporó retroalimentación experta. Finalmente, se modelaron los procesos seleccionados en términos de pasos, datos y condiciones operativas, definiendo las simplificaciones necesarias para su implementación en un entorno prototípico.

3.2.1 ALCANCE FUNCIONAL Y SELECCIÓN DE PROCESOS

El alcance funcional del prototipo se definió a partir de la identificación de los trámites institucionales más frecuentes y representativos para un afiliado, seleccionando un conjunto de procesos que permitiera evaluar tanto la capacidad del agente para ejecutar acciones mediante herramientas del MCP como su aptitud para brindar información personalizada. A partir del relevamiento documental y del análisis preliminar del dominio, se seleccionaron seis procesos, organizados en dos categorías:

Procesos transaccionales

- Solicitud de reintegro.
- Solicitud de afiliación.
- Comunicaciones institucionales (reclamos, quejas o solicitudes generales).

Consultas informativas personalizadas

- Consultar práctica cubierta.
- Consultar tope por práctica.
- Consultar historial de consumo.

La selección se basó en criterios de frecuencia de uso, relevancia operativa para las obras sociales y viabilidad de implementación dentro de un prototipo individual. Los tres procesos transaccionales permiten evaluar la capacidad del agente conversacional para ejecutar acciones concretas dentro del flujo previsto, mientras que las tres consultas informativas permiten validar su desempeño en la recuperación de datos personalizados y en la gestión de respuestas basadas en la información del afiliado.

Estos seis procesos definieron el alcance funcional estable del prototipo y sirvieron como base para las etapas de modelado, diseño, implementación e integración del sistema.



3.2.2 PROCESOS EXCLUIDOS DEL ALCANCE

Además de los seis procesos seleccionados, se identificaron numerosos trámites y operaciones institucionales que quedaron deliberadamente fuera del alcance del prototipo. Entre ellos se incluyen:

- Gestiones internas administrativas (altas de prestadores, autorizaciones médicas, auditorías, derivaciones).
- Emisión y renovación de credenciales.
- Trámites vinculados a pagos, facturación o recupero financiero.
- Modificaciones sensibles del legajo del afiliado.
- Procesos que requieren integración con sistemas externos o validaciones documentales avanzadas.

Aunque estos procedimientos forman parte de la operatoria real de una obra social, presentan niveles de complejidad, dependencia normativa o interacción con sistemas internos que exceden el objetivo de un prototipo funcional desarrollado en el marco de un trabajo final individual. La exclusión de estos procesos permitió acotar el dominio a un conjunto manejable y representativo, asegurando que el desarrollo y la integración del agente conversacional se concentraran en escenarios viables de implementar con datos simulados y herramientas controladas.

3.2.3 FUENTES DE INFORMACIÓN UTILIZADAS EN EL RELEVAMIENTO

El relevamiento de los procesos se basó en dos fuentes principales de información. La primera fue un análisis documental que incluyó reglamentos, instructivos, normativas vigentes y descripciones operativas de diferentes obras sociales. Estas fuentes permitieron identificar los trámites más frecuentes, sus condiciones formales y los requisitos administrativos asociados a cada uno.

La segunda fuente consistió en consultas informales realizadas a personal técnico con experiencia en sistemas utilizados por obras sociales. Estas consultas no constituyeron entrevistas formales ni técnicas estructuradas de recolección de datos; su propósito fue únicamente aclarar dudas puntuales surgidas del análisis documental y verificar



interpretaciones sobre ciertos aspectos operativos que no siempre se encuentran explicitados en la normativa pública.

Estas dos fuentes permitieron construir un panorama inicial del dominio y seleccionar los procesos que serían representados en el prototipo, sin requerir técnicas adicionales de relevamiento en esta etapa.

3.2.4 MODELADO DE LOS PROCESOS SELECCIONADOS

El modelado de los seis procesos seleccionados tuvo como objetivo traducir los trámites institucionales a flujos operativos estructurados que pudieran ser implementados posteriormente en el agente conversacional y en los servicios del backend. Para cada proceso se describieron los pasos principales, las condiciones de inicio, los datos necesarios para su ejecución, las validaciones correspondientes y los resultados esperados.

En los procesos transaccionales —solicitud de reintegro, solicitud de afiliación y comunicaciones institucionales— el modelado incluyó la identificación de la documentación asociada, los criterios mínimos para la recepción y las verificaciones que el sistema debía realizar antes de considerar una operación como válida. En los procesos informativos —consulta de práctica cubierta, consulta de tope por práctica y consulta de historial de consumo— se definieron los atributos del afiliado y de las prestaciones que condicionan la respuesta personalizada del sistema.

El modelado contempló tanto escenarios típicos como situaciones excepcionales, lo que permitió identificar variaciones frecuentes en los flujos institucionales y preverlas dentro del comportamiento esperado del prototipo. Esta estructuración funcional sirvió como base para definir las herramientas del MCP, la lógica del backend y la organización de la base de datos simulada.

Para representar formalmente los flujos se elaboraron diagramas UML que se incluyen en el Anexo A: Especificación de requerimientos debido a su extensión. En esta sección se presenta únicamente la síntesis conceptual necesaria para comprender la posterior implementación del prototipo.



3.2.5 SIMPLIFICACIONES ADOPTADAS PARA EL PROTOTIPO

Dado que el objetivo del proyecto fue desarrollar un prototipo funcional dentro de un entorno controlado, fue necesario adoptar ciertas simplificaciones operativas respecto de los procesos institucionales reales. Estas simplificaciones permitieron asegurar la viabilidad técnica del sistema sin comprometer la esencia de los flujos administrativos ni los criterios centrales de cada trámite.

3.2.6 VALIDACIÓN DEL MODELADO

El modelado preliminar de los procesos fue revisado mediante una instancia de validación experta con personal técnico vinculado al desarrollo y operación de sistemas utilizados por obras sociales. A través de entrevistas semiestructuradas se confirmaron los pasos principales de cada trámite, se aclararon condiciones particulares que no se encontraban explicitadas en la documentación formal y se identificaron variaciones habituales en la operatoria institucional.

Los ajustes derivados de esta revisión consistieron en precisiones menores en los flujos y en la interpretación de ciertos requisitos administrativos, sin modificar el conjunto de procesos seleccionados ni el alcance funcional del prototipo. El acta correspondiente se incorpora en el Anexo B: Acta De Validación de Requerimientos como respaldo del proceso de validación.

Estas decisiones no alteraron la lógica fundamental de los procesos seleccionados y permitieron construir un modelo operativo claro y consistente, adecuado para evaluar el comportamiento del agente conversacional y las capacidades del sistema en su conjunto.

3.3 DISEÑO CONCEPTUAL DEL PROTOTIPO

El diseño conceptual del prototipo define la estructura general del sistema, los roles de cada componente y la forma en que interactúan para permitir la ejecución de los procesos institucionales seleccionados. Esta etapa estableció las bases técnicas que guiaron la posterior implementación y se orientó a garantizar un modelo modular, comprensible y coherente con los requerimientos funcionales definidos en el análisis previo.



3.3.1 ARQUITECTURA GENERAL DEL SISTEMA

La arquitectura general del prototipo se define a partir de los componentes principales que conforman el sistema y de las relaciones entre ellos. Para apoyar esta descripción se incluye en esta misma sección un diagrama de componentes UML, cuya función es ofrecer una representación visual de la estructura del sistema y de los canales de comunicación entre los módulos.

Los componentes identificados son los siguientes:

- Interfaz de Usuario (React.js): gestiona la interacción directa con el afiliado mediante una interfaz tipo chat. Envía las consultas al backend y muestra las respuestas generadas por el agente conversacional.
- Backend en FastAPI: implementa la lógica operativa del sistema y cumple dos funciones principales:

- exponer endpoints HTTP/REST utilizados por el frontend,
- ejecutar las herramientas que el agente conversacional invoca mediante el MCP.

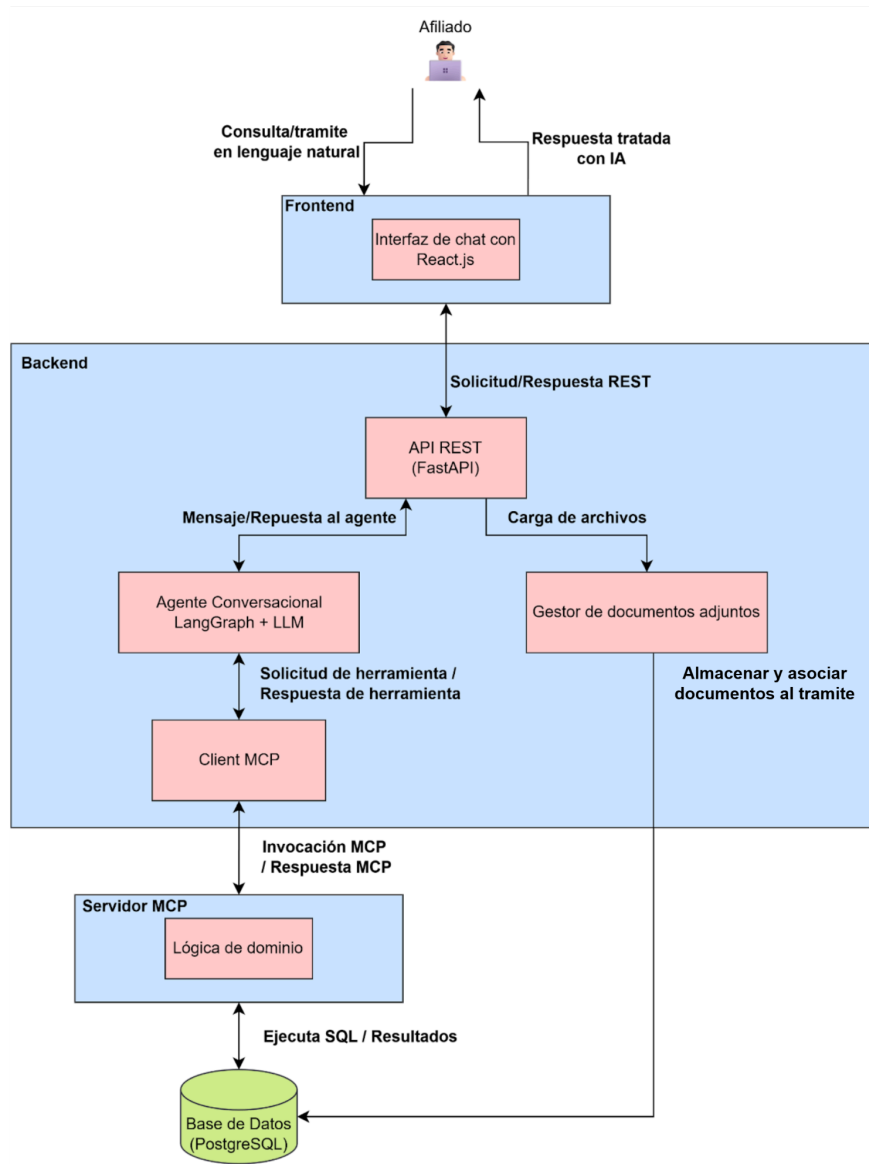
Dentro del backend se integra el cliente MCP, que recibe solicitudes del agente y ejecuta las acciones correspondientes. Asimismo, el backend incorpora el módulo de gestión de documentos, encargado de recibir y procesar archivos PDF enviados por el usuario.

- Agente conversacional (LangGraph + LLM): se ejecuta dentro del backend y utiliza el framework LangGraph. Interpreta las consultas del usuario, razona sobre el contexto y decide cuándo responder directamente o cuándo invocar una herramienta MCP.
- Servidor MCP: componente independiente responsable de exponer las herramientas que representan operaciones del dominio, tales como registrar solicitudes, consultar prestaciones, validar datos o recuperar información. El agente se comunica con este servidor para ejecutar estas acciones.
- Base de datos PostgreSQL: almacena la información simulada del dominio (afiliados, prácticas, topes, consumos y solicitudes). El backend consulta esta base de datos para completar las operaciones requeridas por el agente.

La figura incluida a continuación refleja gráficamente esta arquitectura y muestra la separación de responsabilidades entre los módulos del sistema.

Figura 4

Arquitectura lógica del prototipo



Nota. Elaboración propia.



3.3.2 DISEÑO PRELIMINAR DE LA INTERFAZ

Como parte del diseño conceptual del prototipo se elaboraron mockups preliminares de la interfaz con el propósito de anticipar la experiencia de usuario y definir la estructura básica de interacción. Estos diseños permitieron visualizar la disposición de los elementos principales del sistema antes de avanzar a su implementación en React, facilitando la toma de decisiones sobre el flujo conversacional y la organización general de la pantalla.

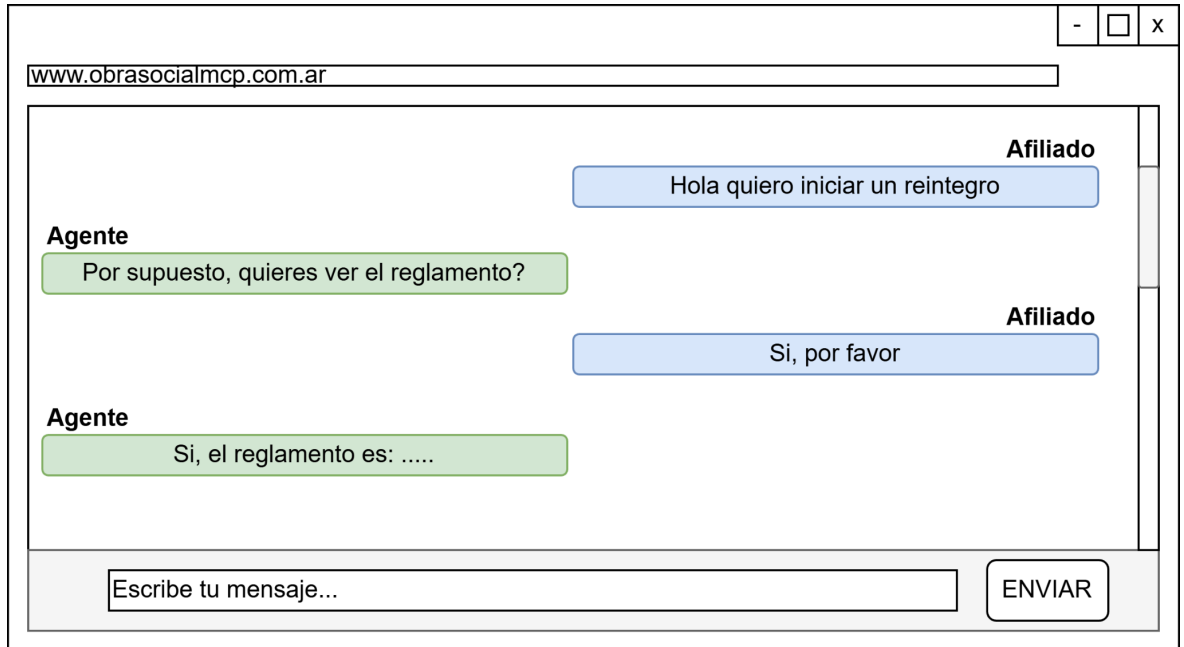
Los mockups contemplan dos vistas fundamentales:

- Vista inicial del chat: representa el estado en el que el afiliado inicia la interacción con el sistema y visualiza la estructura básica del intercambio de mensajes.
- Vista conversacional con adjunto de archivo: ilustra el funcionamiento del diálogo una vez iniciado, incluyendo el historial de mensajes y el espacio destinado a la carga de documentos cuando el proceso lo requiere. Esta vista permitió prever cómo se integrarían elementos interactivos adicionales sin afectar la claridad de la conversación.

Estos modelos cumplen una función exploratoria y orientativa dentro del proceso de diseño: no constituyen la versión final de la interfaz, sino una representación conceptual que guiaba las decisiones posteriores sobre la construcción del frontend.

Figura 5

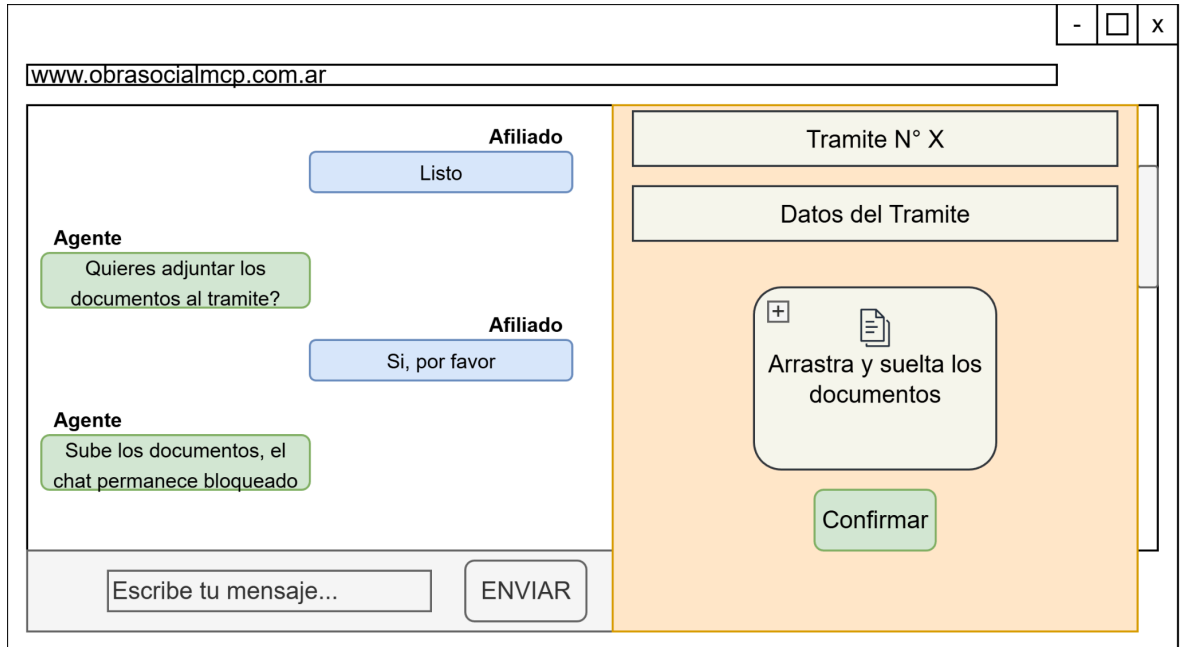
Interfaz conversacional web.



Nota. Elaboración propia.

Figura 6

Interfaz conversacional web durante la carga de documentos.



Nota. Elaboración propia.

3.3.3 FRONTEND

El frontend del prototipo se concibió como una interfaz conversacional orientada a facilitar el intercambio entre el afiliado y el sistema. Desde el punto de vista del diseño conceptual, su función principal es ofrecer un entorno claro y sencillo que permita interactuar con el agente conversacional sin requerir conocimientos técnicos por parte del usuario.

El diseño contempla los siguientes aspectos fundamentales:

- Interfaz tipo chat, que organiza el intercambio en forma de mensajes sucesivos, manteniendo un historial visible que facilita el seguimiento de la conversación.
- Caja de entrada unificada, desde donde el afiliado puede escribir consultas, enviar datos requeridos por el agente e interactuar con los distintos flujos definidos en el sistema.



- Integración con mecanismos de carga de documentos, que permite adjuntar archivos cuando un proceso transaccional lo requiere, manteniendo la coherencia visual y la continuidad de la conversación.
- Separación de responsabilidades, de modo que el frontend se limita a la presentación y captura de datos, delegando en el backend y el agente toda la lógica operativa y decisoria.

Este diseño conceptual sirvió como guía para la implementación posterior de la interfaz en React, permitiendo traducir las decisiones visuales y estructurales en un entorno interactivo que soporte el flujo conversacional previsto en los procesos seleccionados.

3.3.4 BACKEND Y AGENTE CONVERSACIONAL

El backend constituye el punto de articulación entre la interfaz conversacional y los servicios internos del sistema. Desde el diseño conceptual, su función es recibir las solicitudes provenientes del frontend, gestionar el flujo de interacción con el agente conversacional y ejecutar las operaciones auxiliares necesarias durante la conversación.

En este prototipo, el backend se diseña con tres responsabilidades principales:

- Gestionar la comunicación con el agente, enviando los mensajes del usuario y recibiendo las respuestas generadas por el modelo de lenguaje.
- Procesar archivos adjuntos, permitiendo la recepción de documentos PDF enviados por el afiliado y asociándolos al trámite correspondiente mediante un registro controlado en la base de datos. Esta es la única operación directa del backend sobre la base; todas las demás interacciones con los datos se realizan a través del servidor MCP.
- Coordinar la ejecución de acciones del dominio, delegando en el servidor MCP aquellas operaciones que requieren consultar, validar o actualizar información institucional simulada.

El agente conversacional se integra conceptualmente dentro del mismo backend y se estructura con LangGraph. Su comportamiento se basa en un ciclo de interpretación y decisión guiado por el modelo de lenguaje, que le permite:



- Comprender la intención del usuario.
- Contextualizar la interacción según el historial del diálogo.
- Determinar cuándo debe responder directamente y cuándo debe solicitar la ejecución de una herramienta expuesta por el servidor MCP.
- Generar una respuesta clara y coherente para el afiliado.

Este diseño garantiza una separación estricta de responsabilidades:

- El agente interpreta, decide y coordina el flujo conversacional.
- El servidor MCP ejecuta las operaciones del dominio.
- El backend actúa como intermediario y gestiona exclusivamente la recepción y vinculación de documentos.

3.3.5 SERVIDOR MCP

El servidor MCP constituye el componente central encargado de ejecutar las operaciones del dominio institucional dentro del prototipo. Desde el diseño conceptual, su función es proporcionar un conjunto de herramientas estructuradas que el agente conversacional puede invocar para llevar a cabo tareas específicas asociadas a los procesos seleccionados.

Estas herramientas representan acciones concretas del sistema, tales como:

- Consultar información del afiliado.
- Verificar la cobertura de una práctica.
- Recuperar topes o historial de consumo.
- Registrar solicitudes transaccionales.

A diferencia del backend, cuya responsabilidad principal es gestionar la interacción conversacional y el tratamiento de archivos, el servidor MCP es el único punto autorizado de acceso y modificación al modelo de datos. Todas las consultas, validaciones y actualizaciones necesarias para completar un proceso se canalizan exclusivamente a través de este componente, lo que garantiza una separación clara entre la lógica conversacional y la lógica del dominio.



El agente conversacional se comunica con el servidor MCP mediante un canal dedicado, desde el cual solicita la ejecución de las herramientas disponibles. El servidor, por su parte, gestiona la interacción con la base de datos y devuelve al agente los resultados estructurados, que luego se incorporan al flujo de diálogo. Este mecanismo permite mantener un control estricto sobre las operaciones y asegura que el comportamiento del sistema sea consistente con los procesos modelados conceptualmente.

3.3.6 BASE DE DATOS Y MODELO DE INFORMACIÓN

El modelo de datos del prototipo se diseñó con un enfoque simplificado, orientado a representar únicamente la información necesaria para ejecutar los procesos seleccionados. Su función principal es proporcionar una estructura clara y coherente que permita al servidor MCP consultar, validar y registrar la información requerida por el agente conversacional durante el flujo de interacción.

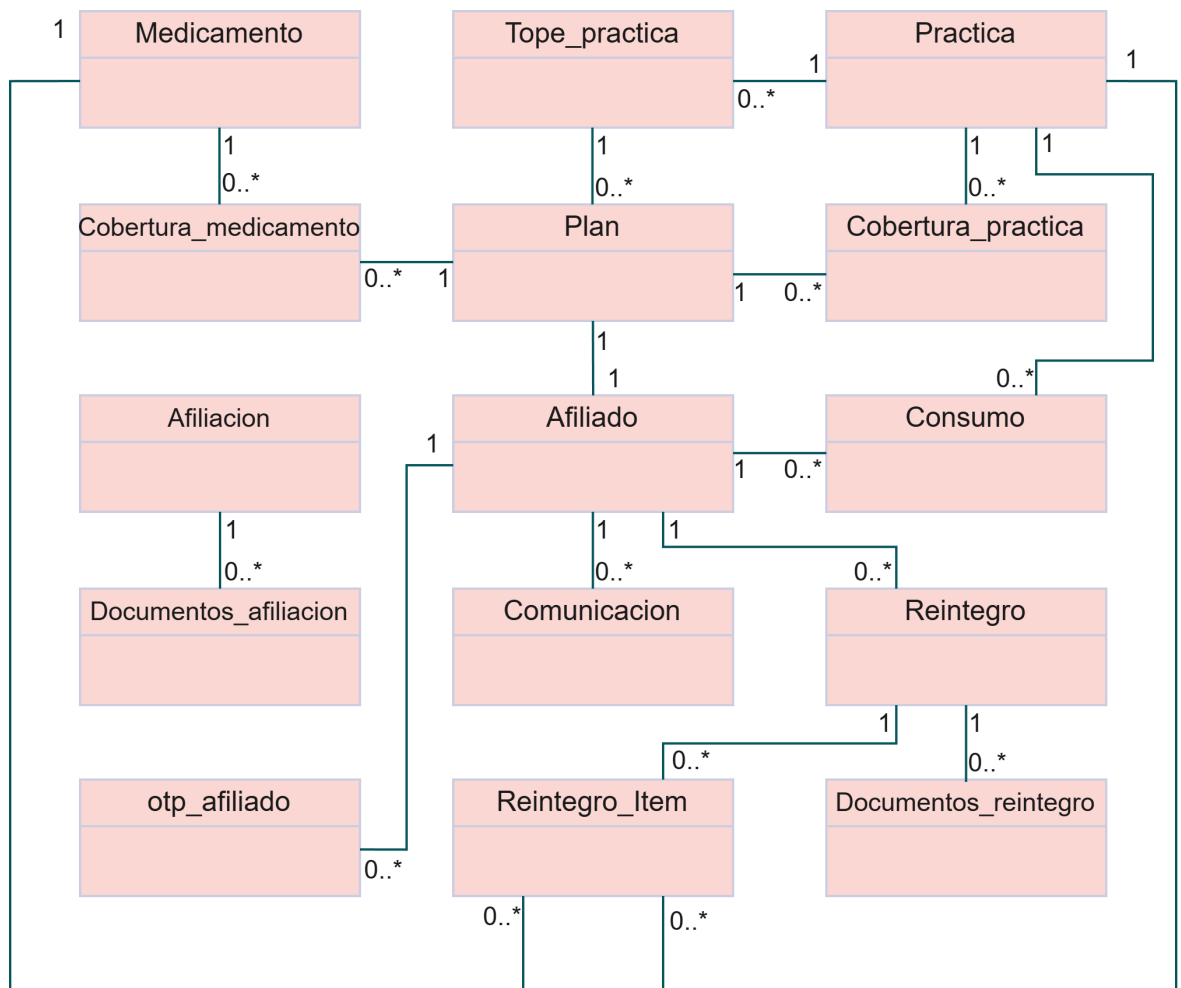
El modelo conceptual incluye algunas de las siguientes entidades principales:

- Afiliado, que agrupa la información básica del usuario del sistema.
- Práctica, que representa las prestaciones médicas disponibles.
- Tope por práctica, que define los límites de cobertura asociados a cada afiliado o grupo.
- Consumo, que registra el uso de prestaciones y permite consultar el historial del afiliado.
- Solicitud, que modela las operaciones transaccionales del sistema, como solicitudes de reintegro, afiliación o comunicaciones institucionales.

Estas entidades se relacionan entre sí de acuerdo con las dependencias lógicas necesarias para soportar los procesos del dominio. El diagrama entidad-relación simplificado presentado en esta sección resume la estructura general del modelo conceptual utilizado durante el diseño.

Figura 7

Modelo entidad-relación general del sistema.



Nota. Elaboración propia.

Este modelo constituye la base conceptual para la ejecución de las herramientas del servidor MCP.



3.4 FUNDAMENTACIÓN DE LA SELECCIÓN DE TECNOLOGÍAS Y HERRAMIENTAS

La construcción del prototipo requirió seleccionar un conjunto de tecnologías que permitiera integrar de manera coherente los componentes del sistema y satisfacer los requerimientos funcionales definidos durante el modelado. La elección del entorno tecnológico no se concibió como una etapa rígida, sino como un proceso exploratorio guiado por pruebas, disponibilidad de herramientas y criterios de simplicidad operativa. En esta sección se describen los fundamentos de dichas elecciones, así como los elementos que conformaron la base técnica utilizada para desarrollar el prototipo.

3.4.1 CRITERIOS GENERALES DE SELECCIÓN

La selección de tecnologías para el prototipo se basó en criterios orientados a garantizar la viabilidad técnica, la integración fluida entre componentes y la simplicidad operativa necesaria en un entorno de desarrollo individual. En particular, se priorizaron:

- Compatibilidad con el MCP: dado que el agente conversacional debía interactuar mediante herramientas estructuradas, resultó esencial elegir un entorno que ofreciera soporte nativo o bien documentado para su integración.
- Simplicidad y claridad en la construcción del prototipo: se privilegiaron frameworks y lenguajes que permitieran desarrollar rápidamente componentes funcionales sin sacrificar estructura ni mantenibilidad.
- Disponibilidad de documentación y soporte comunitario: se valoró que las herramientas contaran con guías oficiales, ejemplos y ecosistemas activos que facilitaran la resolución de problemas durante la implementación.
- Costos operativos y rendimiento: especialmente en relación con la experimentación con modelos de lenguaje, se evaluó la relación entre calidad de respuestas, estabilidad y costo por interacción.
- Experiencia previa del desarrollador: dado el carácter individual del proyecto, se consideró la familiaridad con ciertas tecnologías para reducir tiempos de adaptación y evitar complejidades innecesarias.



Estos criterios permitieron seleccionar un conjunto de herramientas coherente, estable y adecuado para el objetivo del prototipo, manteniendo un equilibrio entre facilidad de implementación y robustez técnica.

3.4.2 LENGUAJE Y FRAMEWORK BASE

Para el desarrollo del prototipo se seleccionó Python como lenguaje principal debido a su amplia adopción en proyectos de inteligencia artificial, la disponibilidad de bibliotecas orientadas a la integración con modelos de lenguaje y, en particular, su compatibilidad con el MCP, cuyo SDK oficial está diseñado para este entorno. Esta elección permitió reducir la complejidad técnica de la integración con el agente conversacional y facilitó la experimentación durante las primeras etapas del proyecto.

Sobre esta base se adoptó FastAPI como framework para la construcción del backend. La decisión estuvo motivada por su enfoque minimalista, su sintaxis clara y su capacidad para definir servicios de manera rápida y estructurada, características especialmente adecuadas para un prototipo. FastAPI ofrece además soporte nativo para manejo de solicitudes HTTP/REST, validación automática de datos y documentación integrada, lo que permitió crear un backend funcional y fácilmente extensible.

La combinación de Python y FastAPI proporcionó un entorno estable, de bajo costo cognitivo y con un ecosistema comunitario amplio, alineado con los criterios generales definidos para la selección tecnológica.

3.4.3 FRAMEWORK CONVERSACIONAL DEL AGENTE

Para el diseño del agente conversacional se seleccionó LangGraph, un framework orientado a la construcción de agentes que combinan razonamiento con la ejecución estructurada de acciones. Su adopción respondió a la necesidad de contar con un entorno flexible que permitiera integrar un modelo de lenguaje con herramientas definidas a través del MCP, manteniendo al mismo tiempo un control explícito sobre el flujo conversacional.

LangGraph ofrece una representación clara del ciclo de interacción del agente, permitiendo modelar de manera declarativa cómo interpreta las instrucciones del usuario, cómo administra



el contexto del diálogo y bajo qué condiciones debe solicitar la ejecución de una herramienta del backend. Esta estructura fue especialmente adecuada para un prototipo donde resultaba fundamental garantizar:

- Una separación limpia entre razonamiento y acciones.
- La capacidad de delegar operaciones al servidor MCP.
- La posibilidad de ajustar el comportamiento del agente durante la integración técnica sin modificar su arquitectura general.

La disponibilidad de documentación reciente, su compatibilidad con modelos de lenguaje accesibles mediante API y su integración natural con Python hicieron de LangGraph una opción apropiada para el desarrollo del agente dentro del entorno propuesto.

3.4.4 MODELO DE LENGUAJE

La selección del modelo de lenguaje se llevó a cabo mediante un proceso exploratorio en el que se evaluaron distintas alternativas disponibles, con el objetivo de identificar una opción que ofreciera estabilidad conversacional, buena interpretación de instrucciones y compatibilidad con el uso de herramientas a través del MCP. Esta exploración constituyó una iteración temprana del proyecto y permitió reducir la incertidumbre técnica asociada al comportamiento del agente dentro del dominio de las obras sociales.

Durante las pruebas se analizaron modelos de diferentes proveedores, comparando su coherencia en diálogos extendidos, capacidad para seguir instrucciones estructuradas, sensibilidad al contexto y costo operativo por interacción. A partir de esta evaluación se seleccionó DeepSeek 3.2, que presentó un equilibrio adecuado entre calidad de respuesta, rendimiento y costos, lo que resultó especialmente relevante en el marco de un prototipo individual.

La integración del modelo elegido no requirió modificaciones en la arquitectura del agente ni en la estructura del sistema, sino únicamente ajustes finos en las instrucciones conversacionales y en la configuración del entorno. Esto permitió mantener la estabilidad del diseño conceptual y avanzar hacia las etapas de integración y validación con un comportamiento conversacional consistente.



3.4.5 INTERFAZ DE USUARIO

Para la construcción de la interfaz de usuario se seleccionó React, una biblioteca ampliamente utilizada para el desarrollo de aplicaciones web interactivas. Su adopción respondió a la necesidad de contar con un entorno flexible y modular que permitiera implementar de manera sencilla una interfaz tipo chat, manteniendo un ciclo de actualización fluido y una organización clara de componentes.

React ofrece ventajas relevantes para un prototipo conversacional, entre ellas:

- Actualización eficiente del estado y del historial de mensajes, fundamental para representar el flujo dinámico de la conversación.
- Componentización, que permite separar elementos visuales y facilitar ajustes posteriores sin afectar la estructura completa de la interfaz.
- Amplia documentación y ecosistema, que acelera el desarrollo y reduce la complejidad de resolver problemas en etapas tempranas.
- Familiaridad del desarrollador, un factor clave para optimizar tiempos y evitar sobrecarga técnica innecesaria dentro del alcance del prototipo.

La elección de React permitió construir rápidamente una interfaz funcional y compatible con el backend, manteniendo la simplicidad requerida en un entorno de desarrollo individual.

3.4.6 BASE DE DATOS

Para la persistencia de datos se seleccionó PostgreSQL, un motor de base de datos relacional ampliamente utilizado por su estabilidad, madurez y capacidad para manejar estructuras consistentes en entornos de desarrollo y producción. En el contexto del prototipo, la elección se fundamentó en los siguientes aspectos:

- Confiabilidad y solidez del modelo relacional, adecuadas para representar de manera clara las entidades del dominio y sus relaciones.
- Simplicidad en el despliegue y configuración, lo que permitió crear rápidamente un entorno controlado para almacenar los datos necesarios sin requerir herramientas adicionales.



- Amplia documentación y soporte comunitario, que facilita la resolución de problemas y asegura compatibilidad con frameworks de Python utilizados en el proyecto.
- Adecuación al alcance del prototipo, dado que las necesidades de persistencia eran moderadas y requerían únicamente una estructura estable para soportar las consultas y operaciones expuestas por el servidor MCP.

PostgreSQL ofreció un equilibrio adecuado entre robustez y facilidad de uso, permitiendo desarrollar una base de datos funcional sin introducir complejidad innecesaria en el entorno del prototipo.

3.5 IMPLEMENTACIÓN DEL PROTOTIPO

3.5.1 IMPLEMENTACIÓN DE LA LÓGICA CONVERSACIONAL

La lógica conversacional del prototipo se implementó sobre un agente basado en un modelo de lenguaje integrado mediante LangGraph. Su funcionamiento se estructura en un ciclo de interpretación y acción que permite procesar las consultas del afiliado, mantener el contexto del diálogo y coordinar la ejecución de herramientas cuando el flujo así lo requiere.

En la implementación, este ciclo se materializa a través de los siguientes mecanismos:

- Interpretación de la solicitud: cada mensaje enviado por el usuario desde el frontend es recibido por el backend, que lo remite al agente junto con el historial necesario para preservar la continuidad de la conversación.
- Razonamiento contextual: el agente analiza la intención del usuario y evalúa si la solicitud puede resolverse mediante una respuesta directa del modelo de lenguaje o si requiere activar una operación del dominio expuesta por el servidor MCP.
- Invocación de herramientas: cuando es necesario realizar una acción —como consultar datos del afiliado, verificar cobertura o registrar una solicitud— el agente genera una instrucción estructurada que indica al servidor MCP qué herramienta debe ejecutarse y con qué parámetros. Esta interacción se implementa mediante mensajes formateados que respetan el protocolo MCP.



- Procesamiento de resultados: una vez ejecutada la herramienta, el servidor MCP devuelve al agente un resultado estructurado. El agente interpreta ese resultado y lo incorpora en una respuesta que combina información operativa con lenguaje natural para presentarla de forma clara al afiliado.
- Gestión del flujo conversacional: el agente mantiene un comportamiento consistente a través de múltiples intercambios, resolviendo solicitudes, pidiendo datos adicionales cuando corresponde y adaptando sus respuestas según el estado actual del proceso en curso.

Este enfoque permite separar la interpretación lingüística —gestionada por el modelo de lenguaje— de la ejecución de acciones concretas —responsabilidad del servidor MCP—, logrando un comportamiento conversacional estable y controlado. La implementación de esta lógica constituyó la base sobre la cual se integraron posteriormente los demás componentes del prototipo.

3.5.2 IMPLEMENTACIÓN DEL SERVIDOR MCP Y HERRAMIENTAS

El servidor MCP se implementó como un componente autónomo encargado de exponer las operaciones del dominio en forma de herramientas invocables por el agente conversacional. Su desarrollo consistió en definir un conjunto de funciones estructuradas, cada una asociada a un proceso o consulta específica, y en establecer los mecanismos de comunicación que permiten recibir solicitudes del agente y devolver resultados formateados.

En la implementación, cada herramienta mantiene responsabilidades bien definidas:

- Validación de datos de entrada, asegurando que las solicitudes cuenten con los parámetros necesarios antes de consultar la base de datos.
- Acceso controlado al modelo de información, ejecutando consultas o actualizaciones según lo requerido por el proceso. Al centralizar todas las operaciones en este módulo, se garantiza que el dominio permanezca encapsulado y que el backend no interactúe directamente con la base de datos, excepto para la gestión de archivos.
- Construcción de respuestas estructuradas, que permiten al agente interpretar de manera consistente los resultados y utilizarlos dentro del flujo conversacional.



- La comunicación entre el agente y el servidor MCP se implementó mediante el protocolo estándar del MCP, basado en mensajes estructurados que contienen la herramienta a ejecutar y sus argumentos. Una vez ejecutada la operación, el servidor MCP formatea la respuesta en un objeto que el agente puede integrar sin ambigüedades en su razonamiento.

Este enfoque permitió modelar de manera clara las acciones operativas asociadas a los procesos seleccionados —como verificar la cobertura de una práctica, consultar topes, recuperar historial de consumo o registrar solicitudes transaccionales— y proporcionó una capa de control que garantiza que todos los accesos al dominio se realicen de forma consistente y centralizada.

3.5.3 IMPLEMENTACIÓN DEL BACKEND

El backend se implementó utilizando FastAPI y actúa como punto de coordinación entre la interfaz de usuario, el agente conversacional y el servidor MCP. Su función principal consiste en gestionar el flujo de mensajes, exponer los endpoints necesarios para el frontend y ejecutar las operaciones auxiliares que no forman parte del dominio institucional.

En la implementación, el backend cumple con las siguientes responsabilidades:

- Recepción de mensajes desde el frontend: cada consulta enviada por el usuario se recibe a través de un endpoint HTTP/REST y se envía al agente junto con el historial conversacional necesario para mantener continuidad en el diálogo.
- Comunicación con el agente: el backend transmite las solicitudes al agente, recibe las respuestas generadas por el modelo de lenguaje y las devuelve al frontend en un formato adecuado para su visualización.
- Delegación de acciones al servidor MCP: cuando el agente determina que debe ejecutar una herramienta, el backend actúa como intermediario enviando la instrucción al servidor MCP y recibiendo el resultado correspondiente. De este modo, el backend no ejecuta lógica del dominio, sino que facilita la interacción entre agente y MCP.
- Gestión de archivos adjuntos: en los procesos que requieren documentación, el backend permite recibir archivos PDF enviados desde la interfaz, almacenarlos



temporalmente y generar los registros necesarios para asociarlos posteriormente a una solicitud. Esta es la única interacción directa del backend con la base de datos, limitada exclusivamente a la carga y vinculación de documentos.

- Formateo y retorno de respuestas: tras completar una acción o recibir un resultado del MCP, el backend prepara la respuesta final para el frontend, asegurando una comunicación clara y consistente.

La implementación del backend proporcionó la infraestructura necesaria para sostener la interacción conversacional de extremo a extremo, manteniendo la separación de responsabilidades y garantizando que todas las operaciones del dominio fueran resueltas exclusivamente por el servidor MCP.

3.5.4 IMPLEMENTACIÓN DEL FRONTEND

El frontend se implementó utilizando React, siguiendo el diseño preliminar establecido en la etapa conceptual. Su desarrollo se centró en construir una interfaz conversacional funcional capaz de enviar consultas, visualizar respuestas y permitir la carga de documentos cuando un proceso lo requiere.

La implementación incluyó los siguientes elementos principales:

- Componente de chat: encargado de mostrar el historial de mensajes entre el afiliado y el agente conversacional. Este componente actualiza dinámicamente la conversación a medida que se reciben nuevas respuestas desde el backend.
- Envío de solicitudes al backend: cada mensaje ingresado por el usuario se transmite a través de un endpoint HTTP/REST. El frontend serializa la consulta, mantiene el estado mínimo necesario y muestra al usuario los mensajes enviados y recibidos.
- Representación de estados conversacionales: el sistema muestra mensajes de carga, errores controlados y confirmaciones de acciones según las respuestas proporcionadas por el backend, lo que aporta claridad al proceso interactivo.
- Integración para carga de documentos: en los procesos que requieren adjuntar archivos, el frontend permite seleccionar y enviar documentos PDF sin interrumpir el



flujo conversacional. El archivo se transmite directamente al backend, que gestiona su procesamiento y asociación.

- Estructura modular de componentes: la interfaz se organizó en componentes reutilizables y aislados, lo que permitió mantener un código claro y facilitar ajustes durante la integración final del sistema.

La implementación del frontend completó el ciclo de interacción del prototipo, proporcionando una interfaz simple, funcional y alineada con el objetivo de evaluar el comportamiento del agente conversacional en un entorno controlado.

3.5.5 IMPLEMENTACIÓN DE LA BASE DE DATOS SIMULADA

La base de datos utilizada por el prototipo se implementó a partir del modelo conceptual definido en la sección 3.3.6. Su propósito es representar únicamente la información necesaria para ejecutar los procesos seleccionados, manteniendo una estructura simple, clara y adecuada para un entorno de prueba controlado.

La implementación consistió en tres actividades principales:

- **Creación del esquema**

Se definieron las tablas, claves primarias, claves foráneas, tipos de datos y restricciones básicas necesarias para reflejar las entidades del modelo conceptual. Este esquema proporciona la estructura mínima requerida para que el servidor MCP pueda consultar, validar y registrar la información utilizada por el agente conversacional.

- **Carga de datos ficticios**

Para posibilitar las pruebas del prototipo se insertaron registros simulados en las tablas principales. Estos valores fueron diseñados para cubrir los escenarios de prueba asociados tanto a los procesos transaccionales como a las consultas informativas. La carga de datos no pretende reproducir un entorno institucional completo, sino proveer insumos suficientes para ejecutar los flujos operativos definidos en el diseño.

- **Configuración de la conexión con el backend y el servidor MCP**



Durante la implementación se configuró la conexión entre el backend y la base de datos utilizando un conjunto reducido de operaciones permitido por el diseño del sistema. El backend interactúa con la base únicamente para la gestión de documentos adjuntos, mientras que todas las consultas y operaciones del dominio son ejecutadas exclusivamente por el servidor MCP, lo que asegura una separación clara de responsabilidades.

La estructura completa del esquema físico implementado —incluyendo tablas, campos, restricciones y relaciones— se documenta en el Anexo C: Esquema de Base de Datos, dado que forma parte de la fase de implementación y complementa al modelo conceptual presentado en la etapa de diseño.

3.6 INTEGRACIÓN DEL PROTOTIPO Y AJUSTE DEL COMPORTAMIENTO CONVERSACIONAL

La integración del prototipo requirió articular los componentes desarrollados en las etapas previas y verificar su funcionamiento conjunto dentro del flujo conversacional. A diferencia de la implementación, centrada en la construcción individual de cada módulo, esta etapa se orientó a evaluar la interacción entre frontend, backend, agente conversacional, servidor MCP y base de datos, asegurando que el sistema operara como un todo coherente. Durante este proceso surgieron ajustes propios de la integración, derivados tanto de pruebas internas como de observaciones del tribunal evaluador. Estos ajustes se incorporaron como iteraciones técnicas acotadas, orientadas a mejorar la seguridad, la consistencia de los datos y el comportamiento del agente, sin modificar el diseño conceptual ni reabrir decisiones estructurales del proyecto. Las subsecciones siguientes describen la integración funcional del sistema y las iteraciones realizadas durante esta fase.

3.6.1 INTEGRACIÓN ENTRE COMPONENTES

La integración del prototipo consistió en articular los distintos módulos desarrollados —frontend, backend, agente conversacional, servidor MCP y base de datos— para conformar un sistema funcional capaz de ejecutar los procesos seleccionados mediante interacción conversacional.



El flujo integrado opera de la siguiente manera:

1. Interacción inicial desde el frontend: El usuario ingresa un mensaje o inicia un trámite desde la interfaz conversacional. El frontend envía esta información al backend mediante un servicio HTTP/REST, junto con el estado mínimo necesario para mantener la continuidad visual del diálogo.
2. Gestión del mensaje en el backend: Al recibir la solicitud, el backend la remite al agente conversacional, incorporando el historial pertinente. En esta etapa el backend actúa únicamente como intermediario, sin interpretar la lógica del trámite.
3. Razonamiento del agente y decisiones de acción: El agente analiza la solicitud y determina si basta con generar una respuesta directa o si es necesario ejecutar una acción del dominio. Para ello evalúa el contexto disponible y decide si solicitar al servidor MCP la ejecución de una herramienta.
4. Ejecución de herramientas en el servidor MCP: Cuando el agente requiere información o debe registrar una operación, el backend transmite la instrucción al servidor MCP. Este módulo ejecuta la herramienta solicitada —consultas, verificaciones o registros— accediendo directamente a la base de datos del prototipo y devolviendo al agente un resultado estructurado.
5. Construcción de la respuesta: Con base en el resultado obtenido del MCP, el agente genera un mensaje final que combina información de dominio con lenguaje natural, el cual se devuelve al backend. Este último prepara la respuesta en el formato esperado por el frontend.
6. Actualización de la interfaz: El frontend recibe la respuesta y actualiza el historial de conversación, permitiendo que el usuario continúe el diálogo de manera fluida.

Este esquema de integración permitió que los componentes operaran de forma desacoplada pero coordinada, manteniendo la separación de responsabilidades definida en el diseño conceptual. La comunicación entre módulos demostró ser estable tanto en consultas simples como en procesos transaccionales que requieren múltiples intercambios de información.



La integración completa del sistema constituyó la base para las iteraciones técnicas posteriores, en las que se realizaron ajustes específicos para mejorar la seguridad, la precisión del modelo de datos y el comportamiento del agente en condiciones reales de uso.

3.6.2 ITERACIONES TÉCNICAS DURANTE LA INTEGRACIÓN

Durante la integración del sistema se realizaron diversas iteraciones técnicas destinadas a mejorar el desempeño conversacional del agente, ajustar la arquitectura a condiciones reales de operación y corregir situaciones detectadas en la validación funcional interna. Estas iteraciones no implicaron reabrir fases previas ni modificar el diseño conceptual general del prototipo; por el contrario, consisten en ajustes acotados y propios de un enfoque metodológico híbrido, en el cual el desarrollo avanza mediante ciclos de implementación, prueba y refinamiento.

A continuación se describen las iteraciones realizadas, agrupadas según el momento en que surgieron y el tipo de ajuste requerido.

3.6.2.1 Iteraciones menores derivadas de la validación funcional interna

Una vez implementado el prototipo en su versión inicial, se efectuó la validación funcional interna mediante la ejecución de casos de uso representativos. Esta validación permitió identificar dos situaciones puntuales que afectan la precisión del agente conversacional. Los ajustes derivados se limitaron a la implementación y no modificaron ni los procesos funcionales ni la arquitectura general del sistema.

Corrección en el manejo de fechas

Durante la ejecución de los casos de prueba se observó que el modelo interpretaba algunos años de forma incorrecta, generando respuestas inconsistentes. Para resolver este comportamiento se incorporó la fecha actual en el prompt del sistema, de modo que el agente dispusiera de un punto temporal de referencia estable. El ajuste fue interno al módulo conversacional y no requirió modificar otros componentes.

Mejora en la identificación de prácticas médicas



Las pruebas mostraron que, en ciertos escenarios, el agente no reconocía prácticas mencionadas con variaciones informales o abreviadas. Como refuerzo operativo se incorporó una herramienta MCP adicional que permite recuperar la lista completa de prácticas cubiertas por el afiliado. Esto habilitó una verificación secundaria cuando no se encuentra coincidencia directa. La modificación no alteró los flujos funcionales establecidos ni la arquitectura del prototipo.

Estas iteraciones surgieron directamente de la validación interna y representan ajustes finales orientados a mejorar la precisión del diálogo sin afectar elementos estructurales del sistema.

3.6.2.2 Ajustes técnicos durante el desarrollo e integración progresiva del prototipo

Antes de la validación funcional interna, y mientras se integraban los distintos componentes del sistema, se realizaron ajustes técnicos derivados de la operación real de módulos parcialmente implementados. Estas iteraciones fueron parte del proceso natural de consolidación del prototipo y respondieron a necesidades detectadas durante la construcción e integración progresiva.

Ajuste en la gestión de documentos adjuntos

El diseño preliminar preveía que los documentos adjuntos fueran gestionados por el servidor MCP. Sin embargo, durante las primeras pruebas de integración se verificó que esta aproximación no era adecuada para las operaciones de recepción, validación y almacenamiento temporal requeridas. En consecuencia, se aplicó una modificación arquitectónica puntual que reasignó la gestión de documentos al backend.

El ajuste implicó:

- Incorporar un módulo específico de gestión de archivos en el backend.
- Retirar del MCP la responsabilidad sobre documentos.
- Actualizar la arquitectura técnica para reflejar esta redistribución.



La modificación fue acotada y no alteró los procesos funcionales del sistema, pero permitió adecuar la arquitectura a las necesidades reales de operación.

Selección definitiva del modelo de lenguaje

Durante la construcción del agente conversacional se evaluaron distintos modelos con el fin de asegurar estabilidad y coherencia contextual. Estas pruebas llevaron a seleccionar DeepSeek 3.2, que mostró un desempeño superior en los escenarios del dominio. La sustitución solo requirió ajustar la configuración conversacional, sin afectar los procesos ni la arquitectura del prototipo.

Estos ajustes técnicos forman parte de las iteraciones naturales del desarrollo bajo una metodología híbrida, donde la integración progresiva de componentes revela oportunidades para mejorar la solidez técnica del sistema sin modificar su diseño conceptual general.

3.6.2.3 Ajustes derivados de observaciones del tribunal evaluador

Durante la presentación preliminar del prototipo, el tribunal evaluador realizó observaciones orientadas a reforzar la seguridad y la coherencia operativa del sistema. Estas observaciones dieron lugar a ajustes acotados que se incorporaron durante la etapa de integración, sin reabrir fases previas ni modificar el diseño conceptual general. Los cambios se limitaron a componentes específicos del prototipo, manteniendo intactos los procesos funcionales definidos.

Incorporación del mecanismo de doble factor de autenticación (OTP)

El tribunal señaló la necesidad de reforzar la verificación de identidad en operaciones sensibles. Para atender esta recomendación se agregó un mecanismo de doble factor de autenticación mediante un código temporal enviado al correo del afiliado.

Este ajuste implicó:

- Añadir una nueva entidad en la base de datos para almacenar los códigos OTP emitidos.



- Incorporar un paso adicional en el flujo conversacional, donde el agente solicita y valida el código.
- Extender el módulo MCP con una herramienta encargada de verificar la validez del código.

Si bien esto requirió modificar el modelo de datos, se trató de un cambio puntual que no afectó la estructura general del sistema ni los procesos definidos. El ajuste reforzó la seguridad del prototipo y su alineación con prácticas institucionales reales.

Corrección en la gestión del CBU del afiliado

El tribunal también observó que el prototipo no debía solicitar al usuario ingresar su CBU, sino utilizar únicamente el valor ya registrado institucionalmente. Atendiendo esta recomendación, se ajustó el comportamiento del agente para que dejara de solicitar dicho dato y, en su lugar, lo recupera desde la base de datos cuando fuera necesario.

La modificación consistió en:

- Actualizar la estructura interna de datos para reflejar que el CBU es un dato institucional, no un valor ingresado por el usuario.
- Ajustar la herramienta MCP responsable de recuperar esta información.
- Modificar la lógica conversacional para eliminar cualquier solicitud de ingreso del CBU.

Este ajuste mejoró la coherencia del prototipo y fortaleció su adecuación a lineamientos de seguridad comunes en entornos reales.

3.6.2.4 Ajustes derivados de la validación experta

A partir de la validación experta del prototipo, se incorporaron ajustes puntuales orientados a mejorar la claridad de la interacción y la experiencia del afiliado, sin introducir cambios en la arquitectura general ni en los procesos funcionales definidos. Las observaciones realizadas por los expertos se centraron principalmente en aspectos de usabilidad conversacional y



adecuación del lenguaje al usuario final, y dieron lugar a pequeñas iteraciones para incorporar los ajustes.

Los ajustes implementados se detallan a continuación.

Mejora en la confirmación de ítems durante la carga de reintegros

Durante la validación se observó que, al momento de cargar un ítem en el proceso de solicitud de reintegro, el agente confirmaba la acción mediante un mensaje genérico, sin mostrar información concreta sobre el ítem agregado. A partir de esta observación, se ajustó el comportamiento del agente para que, luego de cada carga, muestre explícitamente los datos del ítem registrado, tales como tipo de prestación y monto informado.

Este ajuste implicó:

- Modificar el mensaje de confirmación del agente para incluir un resumen del ítem cargado.
- Mantener el flujo conversacional existente, agregando únicamente información contextual relevante para el afiliado.

La mejora permite al usuario verificar de manera inmediata que la información ingresada fue interpretada correctamente, reforzando la transparencia del proceso sin alterar su lógica funcional.

Ajuste en la comunicación inicial de las capacidades del agente

Los expertos señalaron la conveniencia de reforzar, al inicio de la conversación, la comunicación de los trámites y consultas que el agente es capaz de gestionar. En respuesta a esta observación, se ajustó el mensaje inicial del agente para que explicita de forma clara y resumida los procesos disponibles, estableciendo expectativas adecuadas desde el primer intercambio.

Este ajuste consistió en:



- Incorporar un mensaje introductorio que enumera los principales trámites soportados por el agente,
- Mantener un lenguaje orientado al afiliado, evitando descripciones técnicas o internas.

El cambio contribuye a una interacción más guiada y reduce la posibilidad de consultas fuera del alcance del agente.

Eliminación de referencias a componentes técnicos internos en la interfaz de interacción

Durante la validación experta se observó que la interfaz de chat del prototipo incluía referencias visibles a componentes técnicos internos del sistema, en particular vinculadas al uso del módulo MCP. Si bien estas referencias resultaban útiles durante el desarrollo y la depuración del prototipo, los expertos señalaron que no aportan valor al afiliado y podrían generar confusión en un contexto de uso real.

A partir de esta observación, se ajustó la interfaz de interacción para eliminar toda mención a componentes técnicos internos, manteniendo dichas referencias únicamente a nivel de implementación y fuera del alcance del usuario final.

Este ajuste implicó:

- Revisar los elementos visibles de la interfaz de chat para identificar referencias técnicas expuestas.
- Eliminar o reemplazar dichas referencias por etiquetas neutras orientadas al usuario.
- Preservar el funcionamiento interno del sistema sin modificaciones en la lógica conversacional ni en la arquitectura del prototipo.

La modificación permitió mejorar la presentación del prototipo y reforzar la separación entre los aspectos técnicos internos del sistema y la experiencia de uso del afiliado, alineando la interfaz con prácticas habituales de sistemas de atención al usuario.

3.6.3 VALIDACIÓN FUNCIONAL INTERNA DEL PROTOTIPO

La validación funcional interna tuvo como objetivo verificar que el prototipo implementado ejecutara correctamente los procesos seleccionados y respondiera de manera coherente a los



distintos escenarios planteados. Esta validación se realizó una vez completada la implementación inicial del sistema, aplicando técnicas de prueba de caja negra y considerando exclusivamente las entradas y salidas observables en la interacción con el agente conversacional.

Para ello se definieron casos de uso representativos que abarcaron tanto los procesos transaccionales como los informativos, incluyendo además escenarios alternativos y de error. Cada caso fue especificado mediante una descripción del escenario, las entradas esperadas y los criterios de aceptación asociados. La ejecución de estos casos se llevó a cabo sobre el prototipo integrado —frontend, backend, agente conversacional, servidor MCP y base de datos ficticia— en un entorno controlado. La descripción completa de los casos de prueba y sus resultados se presenta en el Anexo D.

En total se evaluaron 24 casos de prueba: 22 cumplieron correctamente con los criterios de aceptación, confirmando la validez de los flujos funcionales implementados. En dos casos se detectaron comportamientos menores que no afectaban la estructura del sistema, pero sí requerían mejoras en la precisión del agente conversacional. Específicamente, se observaron inconsistencias en la interpretación de ciertos valores temporales y dificultades puntuales en el reconocimiento de determinadas prácticas mencionadas por el usuario.

Estos hallazgos dieron lugar a las iteraciones descritas en la sección 3.6.2.1, donde se realizaron ajustes acotados sobre la configuración del agente y se incorporó una herramienta MCP adicional para reforzar la identificación de prácticas. Una vez aplicados estos ajustes, se reejecutaron los casos de prueba afectados, verificando la corrección del comportamiento sin necesidad de repetir la validación completa, dado que los procesos funcionales y la arquitectura general del sistema permanecieron inalterados.

La validación funcional interna permitió consolidar el comportamiento del prototipo, confirmando que los componentes integrados operan de manera consistente y que el sistema cumple con los objetivos funcionales establecidos para los procesos seleccionados.



3.6.4 VALIDACIÓN EXPERTA DEL PROTOTIPO

La validación experta tuvo como propósito evaluar el prototipo en función del dominio institucional, con el objetivo de verificar la adecuación de los procesos modelados, el flujo conversacional y el enfoque técnico en relación con las prácticas operativas habituales de las obras sociales. A diferencia de la validación funcional interna, orientada a comprobar la corrección de la lógica del sistema, esta instancia se centró en analizar la coherencia conceptual y operativa del agente dentro del contexto real de uso.

La validación se realizó con la participación de profesionales con experiencia en el análisis funcional, la gestión operativa y el desarrollo de sistemas core para obras sociales. Su conocimiento del dominio permitió contrastar el comportamiento del agente conversacional con los procedimientos administrativos vigentes, evaluando si los flujos implementados representan de manera adecuada los trámites reales y si el diseño conversacional resulta comprensible y consistente para el afiliado.

Durante esta instancia se revisaron de manera guiada los procesos implementados en el prototipo, analizando la correspondencia entre los pasos conversacionales y los procedimientos administrativos reales, la claridad de los intercambios con el usuario, el tratamiento de la información sensible y la viabilidad del enfoque técnico propuesto. Las observaciones y comentarios de los expertos fueron documentados de forma estructurada y se presentan en detalle en el Anexo E: Validación Experta.

Los resultados de la validación permitieron confirmar que el prototipo se ajusta al dominio institucional de las obras sociales, tanto en términos funcionales como conversacionales y técnicos. Asimismo, se identificaron oportunidades de mejora de carácter incremental, principalmente orientadas a la usabilidad y a la evolución futura del sistema, sin requerir modificaciones en la arquitectura ni en los procesos definidos. En conjunto, esta validación experta complementa la evaluación técnica previa y refuerza la consistencia del prototipo propuesto en el marco de este trabajo.



3.7. CIERRE DEL DESARROLLO

El presente capítulo describió el proceso de construcción del prototipo, desde la identificación y modelado de los procesos institucionales hasta la implementación e integración de sus componentes principales. A lo largo de este desarrollo se abordaron de manera progresiva las decisiones técnicas necesarias para materializar un agente conversacional capaz de interactuar con sistemas institucionales simulados y ejecutar acciones del dominio bajo un esquema controlado.

El resultado de este proceso es un prototipo funcional que integra una interfaz conversacional web, un backend de coordinación, un agente conversacional basado en un modelo de lenguaje y un servidor MCP responsable de ejecutar las operaciones del dominio sobre una base de datos simulada. La integración de estos componentes permitió validar, en un entorno controlado, el flujo completo de interacción entre el afiliado y el sistema, desde la formulación de una consulta en lenguaje natural hasta la ejecución de acciones asociadas a trámites administrativos específicos.

3.7.1. FUNCIONAMIENTO DEL PROTOTIPO

El prototipo desarrollado permite llevar adelante seis procesos representativos de la interacción habitual entre afiliados y obras sociales, combinando consultas informativas personalizadas y trámites transaccionales. A través de la interfaz conversacional, el usuario puede iniciar un diálogo en lenguaje natural, proporcionar la información requerida y recibir respuestas contextualizadas en función de su situación particular.

Desde el punto de vista técnico, el agente conversacional interpreta las intenciones del usuario, mantiene el contexto del diálogo y decide, en función de dicho contexto, si debe responder directamente o solicitar la ejecución de una herramienta expuesta por el servidor MCP. Las operaciones del dominio se ejecutan exclusivamente a través de estas herramientas, lo que garantiza un acceso controlado y trazable a la información simulada.

El flujo completo de interacción se desarrolla de manera secuencial y coherente, permitiendo que un mismo diálogo abarque tanto la recopilación de datos como la ejecución de acciones,



sin fragmentar la experiencia del usuario ni delegar la coordinación del trámite fuera del entorno conversacional.

3.7.2. RESULTADO DE LAS PRUEBAS

El prototipo integrado fue sometido a una instancia de validación funcional interna, orientada a verificar la correcta ejecución de los procesos seleccionados y la coherencia del flujo conversacional. Las pruebas permitieron confirmar que los distintos componentes del sistema operan de forma consistente y coordinada, habilitando la ejecución completa de los casos de uso definidos.

Complementariamente, se realizó una validación experta con profesionales con experiencia en el desarrollo y operación de sistemas utilizados por obras sociales. Esta instancia permitió contrastar los flujos implementados con los procedimientos administrativos habituales, así como evaluar la claridad de la interacción conversacional y la adecuación del enfoque técnico al dominio. Los resultados de esta validación confirmaron que el comportamiento del prototipo resulta comprensible y coherente con la operatoria institucional, y que el enfoque propuesto es viable dentro del contexto analizado.

Las observaciones surgidas de ambas instancias dieron lugar a ajustes técnicos puntuales, orientados a mejorar la precisión del diálogo y a reforzar la estabilidad del sistema, sin modificar la arquitectura general ni el alcance funcional definido.

3.7.3. ALCANCES Y RESTRICCIONES DEL DESARROLLO

El prototipo desarrollado se diseñó como una prueba de concepto orientada a evaluar la viabilidad técnica y conceptual del enfoque propuesto. En este sentido, los procesos implementados y la información gestionada se apoyan en una base de datos simulada y en flujos administrativos simplificados, que representan de manera acotada la operatoria real de una obra social.



Entre los principales alcances del desarrollo se encuentra la posibilidad de integrar conversación en lenguaje natural con la ejecución controlada de acciones del dominio, manteniendo trazabilidad y coherencia en el flujo de interacción. Al mismo tiempo, el prototipo no aborda aspectos propios de un entorno productivo, tales como la integración con sistemas reales, la gestión de identidades a gran escala, el cumplimiento normativo exhaustivo o la auditoría operativa continua.

Estas restricciones responden al alcance definido para el proyecto y no invalidan los resultados obtenidos, sino que delimitan el contexto en el cual deben interpretarse. El prototipo cumple así su función como base experimental para analizar el comportamiento de un agente conversacional orientado a la gestión administrativa en salud y sienta las bases técnicas para desarrollos posteriores.

Conclusiones

.



4.1 SÍNTESIS DEL TRABAJO REALIZADO

El presente trabajo abordó la problemática de la gestión administrativa y el acceso a la información en el ámbito de las obras sociales, en un contexto caracterizado por la fragmentación de los sistemas, la multiplicidad de canales de atención y las dificultades de los afiliados para realizar trámites de manera integrada y previsible. Frente a este escenario, el objetivo general del proyecto fue explorar la viabilidad de un agente conversacional inteligente capaz de articular interacción en lenguaje natural con la ejecución controlada de gestiones administrativas.

Para ello, se diseñó e implementó un prototipo de agente conversacional basado en un modelo de lenguaje de gran escala, integrado a una arquitectura desacoplada que incluye un backend de coordinación, un servidor MCP y una base de datos institucional simulada. Este enfoque permitió separar explícitamente el razonamiento conversacional del acceso a los datos y de la ejecución de operaciones del dominio, garantizando un control explícito sobre las acciones realizadas por el sistema.

A lo largo del trabajo se definieron y modelaron procesos representativos de la interacción entre afiliados y obras sociales, se diseñó una arquitectura orientada a la trazabilidad y a la coherencia operativa, y se implementó un prototipo funcional que integra conversación, validación de información y ejecución de acciones dentro de un mismo flujo de diálogo. La validación realizada permitió comprobar que el sistema es capaz de interpretar intenciones en lenguaje natural y de conducir al usuario a lo largo de trámites administrativos concretos, manteniendo consistencia con los procedimientos institucionales modelados.

El proyecto permitió demostrar que la combinación de agentes conversacionales basados en modelos de lenguaje con mecanismos de control como el MCP constituye un enfoque viable para avanzar hacia soluciones de autogestión más integradas en el ámbito de la salud, estableciendo una base técnica y conceptual para el análisis de este tipo de sistemas en contextos institucionales.



4.2. CUMPLIMIENTOS DE LOS OBJETIVOS

El trabajo desarrollado permitió materializar un agente conversacional inteligente orientado a la autogestión de afiliados de obras sociales, basado en un modelo de lenguaje de gran escala integrado a una base de datos institucional mediante un backend controlado por un MCP. La solución implementada habilita la consulta de información personalizada y la iniciación estructurada de trámites administrativos a través del diálogo en lenguaje natural, garantizando mecanismos de validación, trazabilidad y control sobre las acciones ejecutadas. El prototipo resultante demostró la capacidad de interpretar consultas en lenguaje natural, recuperar información contextualizada y ejecutar operaciones asociadas a trámites administrativos reales dentro de un entorno conversacional controlado, manteniendo una separación explícita entre el razonamiento del agente y el acceso a los datos del dominio. La arquitectura adoptada y las pruebas realizadas evidencian la viabilidad del enfoque propuesto para el contexto analizado.

En este marco, se abordó la identificación y modelado de procesos de autogestión representativos de la interacción habitual entre afiliados y obras sociales, seleccionándose seis procesos institucionales —tres de carácter transaccional y tres consultas informativas personalizadas— en función de criterios de frecuencia de uso, relevancia operativa y viabilidad de implementación. Dichos procesos fueron formalizados y validados conceptualmente con especialistas del dominio, constituyendo la base para el diseño del sistema. Sobre esta base se diseñó e implementó una interfaz web conversacional accesible, que permite la interacción fluida con el sistema mediante un esquema de diálogo continuo, integrando tanto el intercambio de mensajes como la carga de documentación cuando los procesos lo requieren.

El agente conversacional desarrollado es capaz de interpretar las intenciones expresadas por el usuario y, en función del contexto del diálogo, decidir entre la generación de respuestas informativas o la invocación de herramientas específicas a través del backend. Este backend opera bajo el control de un MCP configurado como capa de validación lógica y operativa, actuando como único punto autorizado para el acceso y modificación de la información del dominio. De este modo, se garantiza la coherencia semántica de las interacciones, la



integridad de las gestiones administrativas y la trazabilidad de las operaciones realizadas. Asimismo, el prototipo permite registrar solicitudes asociadas a trámites transaccionales y consultar información histórica del afiliado, utilizando una base de datos simulada diseñada específicamente para soportar los procesos considerados.

La evaluación del sistema se llevó a cabo mediante pruebas funcionales con escenarios simulados representativos de situaciones reales, complementadas por una validación experta. Estas instancias permitieron confirmar la coherencia del comportamiento del sistema con los procesos institucionales modelados, así como su utilidad, usabilidad y viabilidad dentro del contexto de uso considerado. En conjunto, el desarrollo realizado permitió evaluar de manera integral la propuesta técnica y su aplicabilidad al dominio de las obras sociales, dentro del alcance definido para el proyecto.

4.3. APORTES DEL PROYECTO

El proyecto desarrollado aporta valor tanto en el plano técnico como en su aplicación al dominio de las obras sociales, al proponer y materializar un enfoque de interacción conversacional que trasciende el uso informativo tradicional de los asistentes virtuales. A diferencia de soluciones centradas exclusivamente en la respuesta automática de consultas, el trabajo integra de manera explícita el diálogo en lenguaje natural con la ejecución controlada de acciones administrativas, estableciendo un marco coherente entre interacción, validación y operación.

Desde el punto de vista técnico, el principal aporte radica en la arquitectura adoptada para la integración de un agente conversacional basado en un modelo de lenguaje con sistemas institucionales. La utilización de un MCP como capa intermedia permite desacoplar el razonamiento del agente del acceso directo a los datos y de la ejecución de operaciones del dominio. Este esquema habilita un control explícito sobre las acciones que el sistema puede realizar, limita el alcance operativo del modelo de lenguaje y garantiza la trazabilidad de cada interacción que deriva en una gestión administrativa. La experiencia obtenida a partir del prototipo demuestra que este enfoque resulta adecuado para dominios sensibles, donde la



integridad de la información y la previsibilidad del comportamiento del sistema son requisitos centrales.

Asimismo, el proyecto aporta una forma concreta de estructurar flujos conversacionales orientados a procesos administrativos, combinando la interpretación flexible del lenguaje natural con lógicas transaccionales bien definidas. El agente no se limita a responder consultas, sino que conduce al usuario a lo largo de un trámite completo, solicitando información, validando datos y ejecutando acciones en función del contexto del diálogo. Este modelo de interacción contribuye a reducir la fragmentación de la experiencia del usuario y a evitar la delegación de la coordinación del trámite fuera del entorno conversacional.

En el plano aplicado, el trabajo ofrece una propuesta alineada con las necesidades actuales de las obras sociales, donde persisten dificultades para integrar canales de atención y brindar experiencias de autogestión consistentes. El prototipo desarrollado demuestra la posibilidad de implementar un asistente conversacional capaz de operar como punto de acceso unificado a información personalizada y a gestiones administrativas básicas, manteniendo coherencia con los procedimientos institucionales. En este sentido, el aporte del proyecto no se limita a la construcción de un artefacto técnico, sino que establece una base conceptual para el diseño de soluciones conversacionales orientadas a la gestión administrativa en salud, especialmente en contextos donde la interoperabilidad y la integración de sistemas aún se encuentran en proceso de maduración.

En conjunto, los aportes del proyecto se inscriben en la intersección entre la ingeniería de software, la informática en salud y las tecnologías conversacionales, ofreciendo un enfoque que combina flexibilidad en la interacción con control en la ejecución. Este equilibrio constituye uno de los principales valores del trabajo y lo diferencia de implementaciones centradas únicamente en la capacidad lingüística del modelo o en la digitalización aislada de trámites.

4.4. JUSTIFICACIÓN Y VALOR DEL USO DE MCP EN LA ARQUITECTURA

La adopción del MCP como componente central de la arquitectura constituye uno de los ejes distintivos del proyecto y un aporte relevante en el contexto de la aplicación de agentes



conversacionales en dominios institucionales sensibles. Más allá de su implementación técnica, el uso de MCP permitió evaluar un enfoque de control conversacional orientado a garantizar coherencia, trazabilidad y gobernanza sobre las acciones ejecutadas por el agente.

En el marco del prototipo desarrollado, el MCP funcionó como una capa intermedia que separa de manera explícita el razonamiento del agente conversacional del acceso directo a los datos y de la ejecución de operaciones del dominio. Este diseño evitó que el modelo de lenguaje interactúa de forma libre con la base de datos o con funciones críticas del sistema, restringiendo su accionar a un conjunto de herramientas definidas, auditables y alineadas con los procesos institucionales modelados. Como resultado, cada acción derivada de una interacción conversacional quedó sujeta a reglas explícitas de validación y a un flujo operativo previamente establecido.

El valor de este enfoque se manifiesta especialmente en el dominio de las obras sociales, donde las gestiones administrativas involucran información sensible, normativas específicas y la necesidad de mantener registros confiables de las operaciones realizadas. La arquitectura basada en MCP permitió demostrar que es posible integrar modelos de lenguaje en este contexto sin delegar en ellos el control operativo del sistema, reduciendo riesgos asociados a comportamientos no deseados, inconsistencias semánticas o ejecuciones fuera de alcance.

Asimismo, la separación de responsabilidades promovida por el MCP favorece la mantenibilidad y la evolución del sistema. Al desacoplar el agente conversacional, el backend de coordinación y el servidor responsable de las operaciones del dominio, la arquitectura resultante habilita la modificación o sustitución de componentes de manera localizada, sin comprometer la coherencia global del sistema. Esta característica resulta especialmente relevante en escenarios donde los modelos de lenguaje, las regulaciones o los procesos institucionales pueden evolucionar de forma independiente.

En términos generales, la experiencia obtenida a partir del prototipo permite concluir que el MCP constituye una estrategia adecuada para estructurar agentes conversacionales orientados a la acción en entornos institucionales. Su uso no solo aporta un mecanismo técnico de control, sino que introduce un marco conceptual que facilita la integración responsable de



inteligencia artificial generativa en sistemas administrativos, equilibrando flexibilidad en la interacción con control en la ejecución.

4.5. LIMITACIONES DEL TRABAJO

El presente trabajo se desarrolló con un alcance deliberadamente acotado, orientado a la construcción y evaluación de un prototipo funcional que permitiera analizar la viabilidad del enfoque propuesto. En este sentido, las conclusiones obtenidas deben interpretarse considerando una serie de limitaciones inherentes al contexto y a las decisiones adoptadas durante el desarrollo.

En primer lugar, el prototipo opera sobre una base de datos simulada y representa una versión simplificada de los procesos administrativos reales de una obra social. Si bien los flujos modelados fueron definidos y validados conceptualmente con expertos del dominio, no se abordó la integración con sistemas productivos existentes ni con infraestructuras reales de interoperabilidad. Aspectos como la gestión completa de identidades, el manejo de grandes volúmenes de datos, la concurrencia de usuarios o la auditoría operativa continua quedaron fuera del alcance del trabajo.

En segundo lugar, el agente conversacional se basa en un modelo de lenguaje preentrenado accesible mediante API. En este contexto, el rendimiento del sistema —en términos de calidad de las respuestas y robustez frente a consultas complejas— está vinculado a la elección del modelo provisto por el servicio utilizado. Si bien existen distintas alternativas con diferentes niveles de desempeño, el acceso a modelos de mayor capacidad implica un costo económico superior. Por este motivo, el trabajo no profundiza en un análisis comparativo exhaustivo del impacto que la selección de modelos más avanzados podría tener sobre el rendimiento global del sistema en escenarios de uso intensivo.

Finalmente, el conjunto de procesos implementados representa sólo una fracción de la operatoria administrativa de las obras sociales. La selección de seis procesos respondió a criterios de relevancia y viabilidad dentro del marco de un trabajo final individual, por lo que no se contemplan trámites de mayor complejidad normativa, integraciones con prestadores externos ni escenarios que requieran validaciones documentales avanzadas. Estas limitaciones



no invalidan los resultados obtenidos, pero delimitan el contexto en el cual el enfoque propuesto fue evaluado.

4.6 LÍNEAS DE TRABAJO FUTURO

A partir de los resultados obtenidos y de las limitaciones identificadas, se abren diversas líneas de trabajo futuro orientadas a extender y profundizar el enfoque propuesto. Estas posibles evoluciones pueden organizarse en función de su impacto técnico, funcional y contextual.

Una primera línea de trabajo se vincula con el fortalecimiento del prototipo desde el punto de vista técnico y operativo. Esto incluye la incorporación de mecanismos de seguridad más avanzados, una gestión de identidades más robusta y la adaptación del sistema a escenarios de mayor concurrencia de usuarios. Asimismo, sería pertinente profundizar el análisis de rendimiento y estabilidad del agente conversacional en contextos de uso prolongado, así como evaluar estrategias adicionales de monitoreo y auditoría de las acciones ejecutadas a través del MCP.

Una segunda línea de evolución se orienta a la ampliación del alcance funcional del sistema. La integración de nuevos procesos administrativos —particularmente aquellos de mayor complejidad normativa o que requieren interacción con prestadores externos— permitiría evaluar la escalabilidad del enfoque conversacional y del modelo de control propuesto. En este sentido, la incorporación de flujos más heterogéneos constituirá una oportunidad para analizar el comportamiento del agente en escenarios con mayores restricciones y dependencias institucionales.

Finalmente, una línea de trabajo relevante consiste en la integración del enfoque desarrollado con infraestructuras reales del sistema de salud. Esto incluye la interoperabilidad con sistemas existentes, el cumplimiento de marcos regulatorios específicos y la adaptación del agente conversacional a contextos organizacionales concretos. La validación del sistema en entornos productivos o en estudios piloto controlados permitiría profundizar el análisis de su impacto real en la experiencia de los afiliados y en la eficiencia de los procesos administrativos.



4.7 CONCLUSIONES FINALES

El desarrollo del presente trabajo permitió analizar de manera integral el potencial de los agentes conversacionales basados en modelos de lenguaje para apoyar la gestión administrativa en el ámbito de las obras sociales. A partir de la construcción y validación de un prototipo funcional, se exploró un enfoque que combina interacción en lenguaje natural con la ejecución controlada de acciones institucionales, atendiendo a las particularidades de un dominio caracterizado por la sensibilidad de la información y la necesidad de coherencia operativa.

Los resultados obtenidos evidencian que es posible integrar modelos de lenguaje en sistemas administrativos sin delegar en ellos el control directo de las operaciones del dominio. La incorporación de un mecanismo de orquestación como el MCP permitió estructurar un marco de interacción en el cual el agente conversacional actúa como intermediario inteligente, pero bajo reglas explícitas que garantizan trazabilidad, validación y previsibilidad en cada gestión realizada. Este enfoque contribuye a reducir los riesgos asociados al uso de inteligencia artificial generativa en entornos institucionales y ofrece una alternativa viable para su adopción responsable.

Desde una perspectiva aplicada, el trabajo demuestra que los asistentes conversacionales pueden evolucionar desde herramientas meramente informativas hacia soluciones de autogestión más integradas, capaces de acompañar al usuario a lo largo de trámites administrativos completos dentro de un mismo flujo de diálogo. Esta capacidad resulta especialmente relevante en el contexto de las obras sociales, donde la fragmentación de canales y sistemas continúa siendo una fuente de ineficiencias y de insatisfacción para los afiliados.

En términos académicos y técnicos, el proyecto aporta una experiencia concreta de diseño, implementación y evaluación de una arquitectura conversacional orientada a la acción, que puede servir como referencia para futuros desarrollos en el campo de la informática en salud y de los sistemas administrativos basados en inteligencia artificial. El enfoque adoptado pone de manifiesto la importancia de combinar flexibilidad en la interacción con mecanismos de



control y gobernanza, especialmente en dominios donde la confiabilidad del sistema es un requisito central.

El trabajo confirma que la integración responsable de agentes conversacionales en la gestión administrativa en salud no solo es técnicamente viable, sino que representa una oportunidad concreta para repensar los modelos de interacción entre las instituciones y sus usuarios. La propuesta desarrollada sienta bases conceptuales y técnicas que pueden ser extendidas y adaptadas en futuros proyectos, contribuyendo al avance de soluciones digitales más accesibles, coherentes y centradas en las persona



BIBLIOGRAFÍA

Amazon Web Services. (s.f.). *¿Qué es el procesamiento del lenguaje natural (NLP)?*. Recuperado el 22 de octubre de 2025, de: <https://aws.amazon.com/what-is/nlp/>

Anthem Insurance Companies, Inc. (s.f.). *Cómo usar la aplicación móvil Sydney Health de Anthem*. Recuperado el 28 de octubre de 2025, de: <https://www.anthem.com/es/member-resources/sydney-app>

Anthropic. (2024). *Introducing the Model Context Protocol*. Recuperado el 5 de noviembre de 2025, de: <https://www.anthropic.com/news/model-context-protocol>

APAC News Network. (2024). *Government launches Ayushman Bharat Digital Mission WhatsApp chatbot for instant healthcare support*. Recuperado el 11 de noviembre de 2025, de: <https://apacnewsnetwork.com/2024/11/government-launches-ayushman-bharat-digital-mission-whatsapp-chatbot-for-instant-healthcare-support/>

Australian Digital Health Agency. (s.f.). *My Health Record*. Recuperado el 19 de octubre de 2025, de: <https://www.digitalhealth.gov.au/initiatives-and-programs/my-health-record>

AXA. (s.f.). *Emma by AXA*. Recuperado el 25 de octubre de 2025, de: <https://www.axa.com.hk/en/emma-by-axa>

Babylon Health. (s.f.). *Babylon Health*. *Wikipedia*. Recuperado el 7 de noviembre de 2025, de: https://en.wikipedia.org/wiki/Babylon_Health

Bastias-Butler, E., & Ulrich, A. (2019). *Transformación digital del sector salud en América Latina y el Caribe: La historia clínica electrónica*. Banco Interamericano de Desarrollo. <http://dx.doi.org/10.18235/0001659>



Díaz, N. (2023). *How Kaiser Permanente is using AI to meet patients' needs faster*. *Becker's Hospital Review*. Recuperado el 13 de noviembre de 2025, de: <https://www.beckershospitalreview.com/innovation/how-kaiser-permanente-is-using-ai-to-meet-patients-needs-faster.html>

Brown, D., Barrera, A., Ibañez, L., Budassi, I., Murphy, B., Shrestha, P., Salomón-Ballada, S., Kriscovich, J., & Torrente, F. (2024). A behaviourally informed chatbot increases vaccination rates in Argentina more than a one-way reminder. *Nature Human Behaviour*, 8(12), 2314–2321. <https://doi.org/10.1038/s41562-024-01985-7>

Burgess, M. (2022). *Babylon disrupted the UK's health system. Then it left*. *Wired*. Recuperado el 2 de noviembre de 2025, de: <https://www.wired.com/story/babylon-disrupted-uk-health-system-then-left/>

Breck, E., Cai, S., Nielsen, E., Salib, M., & Sculley, D. (2017). *The ML test score: A rubric for ML production readiness and technical debt reduction*. IEEE Big Data Conference. doi: 10.1109/BigData.2017.8258038

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... Amodei, D. (2020). *Language models are few-shot learners*. arXiv. <https://doi.org/10.48550/arXiv.2005.14165>

Buenos Aires Ciudad. (s.f.). *Boti: el WhatsApp de la Ciudad*. Recuperado el 17 de octubre de 2025, de: <https://buenosaires.gob.ar/boti>

Caldarini, G., Jaf, S., & McGarry, K. (2022). *A literature survey of recent advances in chatbots*. arXiv. <https://arxiv.org/pdf/2201.06657>

Colakoglu, S., Durmus, M., Polat, Z. P., Yildiz, A., & Sezgin, E. (2025). *User engagement with a multimodal conversational agent for self-care and chronic disease management: A retrospective analysis*. *Journal of Medical Systems*, 49, 76. <https://doi.org/10.1007/s10916-025-02202-2>



Nsaif, W. S., Salih, H. M., Saleh, H. H., & Al-Nuaimi, B. T. (2024). *Conversational agents: An exploration into chatbot evolution, architecture, and important techniques*. The Eurasia Proceedings of Science, Technology, Engineering and Mathematics, 27, 246–262. <https://doi.org/10.55549/epstem.1518795>

CVS Health. (2025). *Aetna launches new AI and digital tools to improve access and care*. Recuperado el 8 de noviembre de 2025, de: <https://www.cvshealth.com/news/innovation/aetna-launches-new-ai-and-digital-tools-to-improve-access-and-care.html>

Damingo, L., Igulu, K., Saturday, N., Asunogie, T., & Elliot, K. (2025). *Virtual health assistants: AI in patient engagement*. In *Lecture Notes* (pp. 1–15). https://doi.org/10.1007/978-3-031-80813-5_16

Denecke, K. (2024). *How do conversational agents in healthcare impact on patient agency*. Health Informatics. <https://aclanthology.org/2024.teicai-1.1.pdf>

Dev, S. (2020). *AI powered tech of Apollo 24I7 serves millions since Feb this year*. *Express Computer*. Recuperado el 27 de octubre de 2025, de: <https://www.expresscomputer.in/interviews/ai-powered-tech-of-apollo-24i7-serves-millions-since-feb-this-year/62218/>

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). *BERT: Pre-training of deep bidirectional transformers for language understanding*. NAACL 2019. <https://doi.org/10.48550/arXiv.1810.04805>

Dirección Nacional de Protección de Datos Personales. (2005). *Disposición N.º 2/2005: Guía de buenas prácticas en protección de datos personales*. Recuperado el 21 de octubre de 2025, de: <https://www.argentina.gob.ar/normativa/nacional/disposici%C3%B3n-2-2005-103921/texto>

DKV Seguros. (s.f.). *App Salud Digital: Quiero cuidarme Más*. Recuperado el 23 de octubre de 2025, de: https://dkvclubdesalud.dkvseguros.com/app_salud_digital



- Eysenbach, G. (2001). *What is e-health?* *Journal of Medical Internet Research*, 3(2), e20.
<https://doi.org/10.2196/jmir.3.2.e20>
- European Commission. (2024). *Legal framework of EU data protection*. Recuperado el 14 de noviembre de 2025, de:
https://commission.europa.eu/law/law-topic/data-protection/legal-framework-eu-data-protection_en
- Garcia, P., Ma, S. P., Shah, S., Smith, M., Jeong, Y., Devon-Sand, A., et al. (2024). *Artificial intelligence-generated draft replies to patient messages*. *JAMA Network Open*.
<https://jamanetwork.com/journals/jamanetworkopen/fullarticle/2816494>
- GeeksforGeeks. (2025). *Natural Language Processing (NLP) Pipeline*. Recuperado el 20 de octubre de 2025, de:
<https://www.geeksforgeeks.org/nlp/natural-language-processing-nlp-pipeline/>
- Gil Moreira, J. (2023). *Un sistema que cruje: La atención sanitaria, principal motivo de reclamo en un ranking bonaerense*. *La Nación*. Recuperado el 10 de noviembre de 2025, de:
<https://www.lanacion.com.ar/sociedad/un-sistema-que-cruje-la-atencion-sanitaria-principal-motivo-de-reclamo-en-un-ranking-bonaerense-nid19072023/>
- Gobierno de Mendoza. (2021). *Así funciona el chat virtual MendoBot*. Recuperado el 26 de octubre de 2025, de:
<https://www.mendoza.gov.ar/prensa/asi-funciona-el-chat-virtual-mendobot-para-realizar-todo-tipo-de-consultas/>
- Gobierno de Neuquén. (2017). *Salud: presentaron la plataforma digital Andes*. Recuperado el 19 de octubre de 2025, de:
<https://www.neuqueninforma.gob.ar/noticias/2017/07/28/104424-salud-presentaron-la-plataforma-digital-andes>



Gobierno de Santa Fe. (2024). *La Provincia actualiza el sistema de registro de información de atenciones en la salud pública*. Recuperado el 20 de octubre de 2025, de: <https://www.santafe.gob.ar/noticias/noticia/280141/>

Google Cloud. (2025). *What is the Model Context Protocol (MCP)?*. Recuperado el 9 de noviembre de 2025, de: <https://cloud.google.com/discover/what-is-model-context-protocol>

Government of Dubai Media Office. (2021). *Dubai now provides an option to book the COVID-19 vaccine via WhatsApp*. Recuperado el 4 de noviembre de 2025, de: <https://www.mediaoffice.ae/en/news/2021/may/31-05/dubai-now-provides-an-option-to-book-the-covid-19-vaccine-via-whatsapp>

Haystack. (s.f.). *Haystack: Neural Search Framework*. Recuperado el 18 de octubre de 2025, de: <https://haystack.deepset.ai>

Hegde, R. (2024). *MLOps: Continuous Delivery and Automation Pipelines in Machine Learning*. Medium. Recuperado el 4 de noviembre de 2025, de: <https://medium.com/%40rajuhegde2006/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning-093cd6e09fb3>

Hospital Italiano de Buenos Aires. (2024). *Desafíos en la integración de los LLMs en la atención médica*. Recuperado el 16 de noviembre de 2025, de: <https://www.hospitalitaliano.org.ar/hiba/es/news/desafios-en-la-integracion-de-los-llms-en-la-atencion-medica>

IBM. (2024). *What is natural language processing?*. Recuperado el 12 de noviembre de 2025, de: <https://www.ibm.com/think/topics/natural-language-processing>

IMSS. (2020). *Plataforma Interactiva de Atención VIHrtual*. Recuperado el 29 de octubre de 2025, de: <https://www.imss.gob.mx/pivihimss>

IMSS. (2021). *IMSS dispone de aplicativo Mi Chat ONCOIMSS para mejorar atención a pacientes pediátricos oncológicos*. Recuperado el 3 de noviembre de 2025, de: <https://www.imss.gob.mx/prensa/archivo/202109/390>



- IndiaAI. (2020). *India launches AI-powered “MyGov Corona Helpdesk”*. Recuperado el 29 de octubre de 2025, de: <https://indiaai.gov.in/news/india-launches-ai-powered-mygov-corona-helpdesk>
- INECO. (2024). *Cómo una herramienta tecnológica logró aumentar la tasa de vacunación contra el COVID en Argentina*. Recuperado el 15 de octubre de 2025, de: <https://www.ineco.org.ar/novedades/como-una-herramienta-tecnologica-logro-aumentar-la-tasa-de-vacunacion-contra-el-covid-en-argentina/>
- IOMA. (s.f.). *Centro de atención personalizado – chatbots 24/7*. Recuperado el 22 de octubre de 2025, de: <https://www.ioma.gba.gov.ar>
- Jiang, A. Q., Sablayrolles, A., Mensch, A., Bamford, C., Chaplot, D., ... Scialom, M. (2023). *Mistral 7B*. arXiv. <https://doi.org/10.48550/arXiv.2310.06825>
- Jurafsky, D., & Martin, J. H. (2025). *Speech and language processing* (3rd ed.). Stanford. <https://web.stanford.edu/~jurafsky/slp3/>
- Kasneji, E., Sessler, K., Küchemann, S., Bannert, M., Dementieva, D., Fischer, F., ... Kasneji, G. (2023). *ChatGPT for good? Opportunities and challenges for education. Learning and Individual Differences, 103, 102274*. <https://doi.org/10.1016/j.lindif.2023.102274>
- Kobie, N. (2022). *Babylon disrupted the UK’s health system. Then it left. Wired*. Recuperado el 7 de noviembre de 2025, de: <https://www.wired.com/story/babylon-disrupted-uk-health-system-then-it-left/>
- Kot, S., & Nguyen, P. T. (2020). *The concept of intelligent agent in business interactions. Journal of Business & Industrial Marketing, 35(7), 1155–1168*. <https://doi.org/10.1108/JBIM-10-2018-0291>
- LangChain. (s.f.). *LangChain documentation*. Recuperado el 23 de octubre de 2025, de: <https://docs.langchain.com>



Bibliografía

La Mañana Neuquén. (2023). *El WhatsApp de PAMI: uno a uno, todos los trámites que se resuelven a distancia*. Recuperado el 9 de noviembre de 2025, de: <https://www.lmneuquen.com/pais/el-whatsapp-pami-uno-uno-todos-los-tramites-que-se-resuelven-distancia-n1068819>

Ley N.º 25.326. (2000). *Protección de los Datos Personales*. <https://servicios.infoleg.gob.ar>

Li, E., Clarke, J., Ashrafian, H., Darzi, A., & Neves, A. L. (2022). Impact of EHR interoperability on patient safety. *JMIR*, 24(9), e38144. doi: 10.2196/38144

Lin, Z., Basu, S., Beigi, M., Manjunatha, V., Rossi, R. A., ... Huang, L. (2025). *A survey on mechanistic interpretability for multimodal models*. arXiv:2502.17516

LlamaIndex. (s.f.). *LlamaIndex documentation*. Recuperado el 24 de octubre de 2025, de: <https://developers.llamaindex.ai/python/framework/>

Lu, T., Lin, Q., Yu, B., & Hu, J. (2025). *A systematic review of strategies in digital technologies for motivating adherence to chronic illness self-care*. **npj Health Systems**, 2, Article 13. <https://doi.org/10.1038/s44401-025-00017-4>

Matheny, M., Thadaney Israni, S., Ahmed, M., & Whicher, D. (2019). *Artificial intelligence in health care: The hope, the hype, the promise, the peril*. National Academy of Medicine. <https://doi.org/10.17226/27111>

Khan, S. A. (2025). *UNDERSTANDING THE NLP PIPELINE*. *Medium*. Recuperado el 12 de noviembre de 2025, de: <https://medium.com/@salehakramkhan/>

Medifé. (s.f.). *Te presentamos Medi*. Recuperado el 21 de octubre de 2025, de: <https://www.medife.com.ar/noticias/asociados/te-presentamos-medi>

MeekaC. (2024). *Patient Experience and AI*. *Mayo Clinic Connect*. Recuperado el 8 de noviembre de 2025, de: <https://connect.mayoclinic.org/blog/ehlers-danlos-syndrome/newsfeed-post/patient-experience-and-ai/>



Miller, R. A., Pople, H. E., & Myers, J. D. (1982). *Internist-I*. *New England Journal of Medicine*. <https://doi.org/10.1056/NEJM198208193070803>

Ministerio de Salud de la Nación. (s.f.). *Red Nacional de Salud Digital*. Recuperado el 18 de octubre de 2025, de: <https://www.argentina.gob.ar/salud/digital/red>

Ministério da Saúde. (2023). *Ministério da Saúde lança assistente virtual no WhatsApp com informações oficiais sobre vacinação*. Recuperado el 12 de noviembre de 2025, de: <https://blognossavoz.com.br/ministerio-da-saude-lanca-assistente-virtual-no-whatsapp-para-tir-ar-duvidas-sobre-vacinas-e-desmentir-fake-news/>

Minaee, S., Mikolov, T., Nikzad, N., Chenaghlu, M., Socher, R., Amatriain, X., & Gao, J. (2024). *Large language models: A survey*. arXiv:2402.06196

Model Context Protocol. (2025). *MCP Specification*. Recuperado el 1 de noviembre de 2025, de: <https://modelcontextprotocol.io>

MyChart. (s.f.). *MyChart*. Recuperado el 28 de octubre de 2025, de: <https://www.mychart.org>

NHS. (2024). *About the NHS App*. el 1 de noviembre de 2025, de: <https://www.nhs.uk>

NIB Health. (2024). *nib launches AI-powered Symptom Checker*. Recuperado el 31 de octubre de 2025, de: <https://www.nib.com.au/media/health/nib-launches-ai-symptom-checker>

ObsBA. (2024). *Te presentamos a ObsBI: chatbot y asistente virtual de ObsBA*. Recuperado el 25 de octubre de 2025, de: <https://www.obsba.org.ar/te-presentamos-a-obsbi-chatbot-y-asistente-virtual-de-obsba/>

OMINT. (s.f.). *Mint: chatbot para socios*. Recuperado el 24 de octubre de 2025, de: <https://www.omint.com.ar/website2/default.aspx?tabid=5314>

OpenAI. (2024). *Function calling*. Recuperado el 11 de noviembre de 2025, de: <https://platform.openai.com>



OPS. (s.f.). *Sistemas de información para la salud*. Recuperado el 20 de octubre de 2025, de: <https://www.paho.org/es/temas/sistemas-informacion-para-salud>

OPS. (2021). *La inteligencia artificial en la salud pública*. (PAHO/EIH/IS/21-011). <https://iris.paho.org/items/e13b6904-bfc3-4cc8-9e61-0ef350433dd9>

OPS. (2025). *Transformación digital del sector salud en la Región de las Américas*. <https://iris.paho.org/handle/10665.2/64574>

Oracle. (2024). *Interoperability in healthcare explained*. Recuperado el 8 de noviembre de 2025, de: <https://www.oracle.com/health/interoperability-healthcare/>

OSDEPYM. (2022). *OSDEPYM: consultas médicas por videollamada a través de la app móvil*. Recuperado el 2 de noviembre de 2025, de: [https://www.osdepym.com.ar/PortalCMS/novedadHome.htm?idNovedad=343#:~:text=Si%20ya%20descargaste%20nuestra%20App,videoconsulta\)%20y%20motivo%20de%20consulta.](https://www.osdepym.com.ar/PortalCMS/novedadHome.htm?idNovedad=343#:~:text=Si%20ya%20descargaste%20nuestra%20App,videoconsulta)%20y%20motivo%20de%20consulta.)

PAMI. (2022). *Pame: asistente virtual de PAMI en WhatsApp*. Recuperado el 5 de noviembre de 2025, de: <https://www.pami.org.ar/bot-pami>

Ping An Health. (2025). *Ping An Xin Yi AI doctor assistant service to enhance accessibility and satisfaction in medical services*. PR Newswire. Recuperado el 5 de noviembre de 2025, de: <https://www.prnewswire.com/news-releases/ping-an-health-launches-ping-an-xin-yi-ai-doctor-assistant-service-to-enhance-accessibility-and-satisfaction-in-medical-services-302385892.html>

Radford, A., Narasimhan, K., Salimans, T., & Sutskever, I. (2018). *Improving Language Understanding by Generative Pre-Training*. OpenAI. https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

Radziwill, N. M. & Benton, M. C. (2017). Evaluating Quality of Chatbots and Intelligent Conversational Agents. arXiv preprint arXiv:1704.04579



Bibliografía

Rastogi, A., Zang, X., Sunkara, S., Gupta, R., & Khaitan, P. (2019). *Schema-guided dialogue dataset*. arXiv. <https://arxiv.org/pdf/1909.05855>

Saab, K., & Freyberg, J. (2025). *AMIE gains vision*. Google Research Blog. <https://research.google/blog/amie-gains-vision-a-research-ai-agent-for-multi-modal-diagnostic-dialogue/>

Sancor Salud. (s.f.). *Inteligencia artificial en Sancor Salud: asistente virtual Sandy*. Recuperado el 20 de octubre de 2025, de: <https://osuti.sancorsalud.com.ar/noticias/inteligencia-artificial-en-sancor-salud>

Secretaría de Salud Provincia de Buenos Aires. (2022). *Programa Salud Digital Bonaerense*. Recuperado de: <https://www.ms.gba.gov.ar/sitios/saluddigitalbonaerense/wp-content/uploads/sites/245/2022/1/1/Resumen-del-Plan-Estrategico-de-Implementacion-del-Programa-Salud-Digital-Bonaerense-2022-2027.pdf>

Shortliffe, E. H. (1976). *MYCIN*. <https://doi.org/10.1097/00004669-197610000-00011>

Shawar, B. A., & Atwell, E. (2007). *Chatbots: Are they really useful?* LDV Forum. DOI: <https://doi.org/10.21248/jlcl.22.2007.88>

Sommerville, I. (2016). *Software engineering* (10th ed.). Pearson Education.

UC San Diego Health. (2024). *Study reveals AI enhances physician-patient communication*. Recuperado el 10 de noviembre de 2025, de: <https://health.ucsd.edu/news/press-releases/2024-04-15-study-reveals-ai-enhances-physician-patient-communication/>

Unidad de Auditoría Interna. (2023). *Informe de Auditoría N° 06/2023*. https://www.argentina.gob.ar/sites/default/files/2024/06/06.-_informe_final_no_06-23_-_auditoria_y_sindicatura_social.pdf



- Swiss Medical. (2024). *Swity: tu asistente virtual*. Recuperado el 14 de noviembre de 2025, de: <https://blog.swissmedical.com.ar/swity-tu-asistente-virtual>
- Tapia, J. de, Encina, R., Piangatelli, M. del C., Pirola, J., González, G. P., & Moscoso, N. S. (2023). *Barreras al acceso según las etapas del proceso de atención de la salud de los adultos mayores*. **Gerokomos**, **34**(3), 183–187. https://scielo.isciii.es/scielo.php?script=sci_arttext&pid=S1134-928X2023000300006
- TechTarget. (2024). *Natural language processing (NLP)*. Recuperado el 13 de noviembre de 2025, de: <https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP>
- Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., ... Lample, G. (2023). *LLaMA: Open and efficient models*. <https://arxiv.org/abs/2302.13971>
- UCHealth Today. (2025). *Can AI translations of medical test results reduce confusion and anxiety for patients?*. *UCHealth Today*. Recuperado el 6 de noviembre de 2025, de: <https://www.uchealth.org/today/can-ai-help-patients-understand-their-complex-test-results/>
- U.S. Department of Health & Human Services. (1996). *HIPAA*. <https://www.hhs.gov/hipaa>
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gómez, A. N., Kaiser, Ł., & Polosukhin, I. (2017). *Attention is all you need*. arXiv. <https://arxiv.org/abs/1706.03762>
- Weizenbaum, J. (1966). *ELIZA*. *Communications of the ACM*, *9*(1), 36–45. <https://doi.org/10.1145/365153.365168>
- Wen, B., Norel, R., Liu, J., Stappenbeck, T., Zulkernine, F., & Chen, H. (2024). *Leveraging Large Language Models for Patient Engagement: The Power of Conversational AI in Digital Health*. arXiv. <https://arxiv.org/abs/2406.13659>
- Wikipedia contributors. (s.f.). *Babylon Health*. *Wikipedia*. Recuperado el 11 de noviembre de 2025, de: https://en.wikipedia.org/wiki/Babylon_Health



- World Health Organization (2021a). *Ethics and governance of artificial intelligence for health*. <https://www.who.int/publications/i/item/9789240029200>
- World Health Organization. (2021b). *Global strategy on digital health 2020–2025*. <https://iris.who.int/server/api/core/bitstreams/1f4d4a08-b20d-4c36-9148-a59429ac3477/content>
- World Health Organization. (2020). *WHO Health Alert brings COVID-19 facts to billions via WhatsApp*. Recuperado el 8 de noviembre de 2025, de: <https://www.who.int/news-room/feature-stories/detail/who-health-alert-brings-covid-19-facts-to-billions-via-whatsapp>
- Xie, Q., et al. (2024). Me-LLaMA: Foundation Large Language Models for Medical Applications. <https://doi.org/10.48550/arXiv.2402.12749>
- Yao, S., Yu, D., Zhao, S., Shafiullah, N., & Narasimhan, K. (2022). *ReAct: Synergizing reasoning and acting*. <https://arxiv.org/abs/2210.03629>
- Z.N. & W.P. (2024). Large language models in healthcare and medical domain. arXiv. <https://arxiv.org/html/2401.06775>
- Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., ... Wen, J.-R. (2025). *A survey of large language models (v16)*. <https://doi.org/10.48550/arXiv.2303.1>

ANEXOS

ANEXO A: ESPECIFICACIÓN DE REQUERIMIENTOS

Introducción

El presente anexo reúne la especificación formal y completa de los requerimientos del sistema conversacional desarrollado en el marco de esta tesis. Su finalidad es complementar el análisis presentado en el Capítulo III mediante la presentación detallada de:

- los requerimientos funcionales transversales y específicos,
- los requerimientos no funcionales,
- las precondiciones y postcondiciones asociadas a cada proceso,
- y el manejo de excepciones y errores.

El contenido aquí documentado constituye la base técnica que sustenta el diseño y la implementación del agente conversacional basado en Model Context Protocol (MCP), asegurando la trazabilidad y completitud del sistema modelado.

Actores del Sistema

El sistema conversacional desarrollado se estructura alrededor de dos actores principales, cuya interacción define el conjunto de procesos funcionales incluidos en el alcance del prototipo.

Afiliado

El afiliado es el usuario final del sistema y el actor central en todos los procesos documentados. Interactúa mediante lenguaje natural a través de la interfaz web y puede:

- realizar consultas informativas sobre coberturas, topes y consumos,
- iniciar trámites administrativos como reintegros o solicitudes de afiliación,
- cargar documentación requerida en formato digital,
- registrar reclamos o comunicaciones formales,
- consultar el estado de trámites ya realizados.

El afiliado no necesita conocimientos técnicos ni manipular formularios externos; el agente guía el proceso mediante diálogo interactivo.

Agente Conversacional

El agente conversacional es el sistema inteligente encargado de interpretar las solicitudes del afiliado y coordinar las herramientas necesarias para responder o ejecutar la acción requerida. Sus responsabilidades principales incluyen:

- interpretar la intención del usuario mediante un modelo de lenguaje,
- mantener el contexto conversacional durante la sesión activa,
- invocar herramientas MCP para consultar o registrar información,
- validar datos y solicitar correcciones cuando sea necesario,
- presentar resultados de forma comprensible y guiada.

Aunque actúa como mediador entre el afiliado y la base de datos institucional, el agente no tiene acceso directo a los datos: todas las operaciones se ejecutan utilizando MCP

Requerimientos Funcionales

Los requerimientos funcionales describen el conjunto de capacidades que el sistema debe ofrecer para satisfacer las necesidades operativas identificadas durante el análisis del dominio. En el contexto del agente conversacional basado en MCP, estos requerimientos se dividen en dos grupos:

- Requerimientos funcionales transversales, aplicables a todos los procesos y necesarios para garantizar el funcionamiento coherente del sistema.
- Requerimientos funcionales específicos, asociados directamente a los procesos seleccionados para el prototipo: solicitudes de reintegros, solicitudes de afiliación, comunicaciones, consultas de topes, consultas de consumos y verificación de cobertura.

Cada requerimiento se documenta siguiendo la estructura formal adoptada para este anexo, con el fin de asegurar trazabilidad y claridad en la definición:

- Propósito
- Actores involucrados
- Precondiciones
- Flujo principal
- Flujos alternativos

- Excepciones
- Postcondiciones

Esta sección constituye la base normativa y operativa del comportamiento esperado del agente y cómo interactúa con la base de datos.

Requerimientos Funcionales Transversales

RF-001 — Identificación del Usuario

Código	RF-001
Nombre	Identificación del Usuario
Propósito	Garantizar que cualquier consulta o trámite realizado por el afiliado esté asociado a su identidad real, permitiendo acceder únicamente a la información correspondiente y evitar operaciones no autorizadas.
Actores involucrados	- Afiliado - Agente conversacional
Precondiciones	El afiliado inicia una interacción que requiere acceso a información personal o trámites administrativos.
Flujo principal	1. El agente solicita el DNI y el número de afiliado. 2. El afiliado ingresa los datos solicitados 3. El agente valida el ingreso del afiliado 4. El agente confirma la identidad y continúa con el proceso solicitado.
Flujos alternativos	- Si el DNI o número de afiliado no coinciden, el agente solicita reingreso de datos. - Si el afiliado no existe, el agente informa la situación y detiene el proceso.

Excepciones	-Error en el intercambio de datos entre el backend y el sistema externo. -El servidor externo no devolvió una respuesta válida o no respondió.
Postcondiciones	- El afiliado queda identificado dentro del contexto de sesión.- Sus datos pueden ser utilizados el agente sin volver a solicitar credenciales.

RF-002: Interpretación de Intención

Código	RF-002
Nombre	Interpretación de Intención
Propósito	Determinar qué acción solicita el afiliado a partir de su mensaje en lenguaje natural.
Actores involucrados	- Afiliado - Agente conversacional
Precondiciones	El afiliado ingresa un mensaje a través de la interfaz.
Flujo principal	1. El agente analiza el texto mediante el modelo de lenguaje. 2. Clasifica la intención: consulta, reclamo, reintegro, afiliación, cobertura, topes o consumos. 3. Selecciona el flujo correspondiente y continúa la conversación.
Flujos alternativos	- Mensaje ambiguo → el agente solicita aclaración. - Mensaje incompleto → el agente pide información adicional.
Excepciones	- Incapacidad del modelo de lenguaje para determinar intención.
Postcondiciones	- La conversación avanza por el flujo correcto.

RF-003 — Orquestación de Herramientas MCP

Código	RF-003
Nombre	Orquestación de Herramientas MCP
Propósito	Garantizar que toda operación sobre los datos (consultas o transacciones) sea realizada de manera controlada, evitando acceso directo a la base de datos desde el agente conversacional.
Actores involucrados	- Agente conversacional
Precondiciones	El agente ha identificado una intención que requiere información o ejecución de un proceso.
Flujo principal	<ol style="list-style-type: none"> 1. El agente identifica que se requiere una acción específica. 2. El agente invoca la herramienta correspondiente. 3. La herramienta consulta o modifica la base de datos según sea necesario. 4. El resultado estructurado se devuelve al agente. 5. El agente presenta la respuesta al afiliado.
Flujos alternativos	- Error en la herramienta → el agente comunica la situación y puede solicitar repetir o reiniciar el proceso.
Excepciones	<ul style="list-style-type: none"> - Timeouts. - Fallas de conexión.
Postcondiciones	- La respuesta que recibe el afiliado viene de una herramienta confiable y validada.

Requerimientos Funcionales Específicos por Proceso

RF-004 — Gestión de Solicitud de Reintegro

Código	RF-004
---------------	---------------

Nombre	Gestión de Solicitud de Reintegro
Propósito	Permitir que un afiliado inicie y complete una solicitud de reintegro, incluyendo la carga de documentación respaldatoria y la validación de los datos ingresados.
Actores involucrados	- Afiliado - Agente conversacional
Precondiciones	- El afiliado debe estar identificado (RF-001). - Deben existir prácticas registradas en la base para asociar al reintegro.
Flujo principal	<ol style="list-style-type: none"> 1. El afiliado expresa su intención de iniciar un reintegro. 2. El agente presenta el reglamento o las condiciones necesarias para 3. realizar el trámite. 4. El afiliado confirma que desea continuar. 5. El agente crea un reintegro inicial asociado al afiliado. 6. El agente solicita agregar uno o más ítems (prácticas o medicamentos). 7. El afiliado ingresa los datos correspondientes a cada ítem. 8. El agente valida los datos ingresados (montos, fechas, tipo de práctica). 9. El agente solicita la carga de comprobantes en formato digital. 10. El afiliado carga la documentación requerida. 11. El agente guarda la documentación y envía la solicitud de reintegro
Flujos alternativos	- El afiliado decide no continuar → se cancela la operación sin cambios. - No se agregan ítems → el agente informa que no es posible generar un reintegro vacío.
Excepciones	- Error en la creación del reintegro. - Documentos en formato incorrecto. - Ítems duplicados o inválidos.

Postcondiciones	<ul style="list-style-type: none"> - Se registra un reintegro válido con ítems asociados y documentos adjuntos. - El afiliado puede consultar posteriormente su estado.
------------------------	---

RF-005: Gestión de Solicitud de Afiliación

Código	RF-005
Nombre	Gestión de Solicitud de Afiliación
Propósito	Permitir que un usuario inicie una solicitud de afiliación mediante un diálogo guiado, proporcionando datos personales y adjuntando la documentación requerida.
Actores involucrados	<ul style="list-style-type: none"> - Afiliado - Agente conversacional
Precondiciones	<ul style="list-style-type: none"> - El usuario debe proporcionar datos personales básicos. - El usuario no debe estar previamente registrado como afiliado.
Flujo principal	<ol style="list-style-type: none"> 1. El usuario manifiesta su intención de afiliarse. 2. El agente presenta los requisitos o reglamento del proceso. 3. El agente solicita los datos personales necesarios para completar la solicitud. 4. El agente verifica que el usuario no esté registrado previamente. 5. El agente solicita la carga de la documentación obligatoria. 6. El agente carga los archivos requeridos. 7. El agente valida la integridad y formato de la documentación. 8. El agente registra la solicitud. 9. El agente informa al usuario que la afiliación se encuentra en evaluación.

Flujos alternativos	<ul style="list-style-type: none"> - Usuario ya registrado → el sistema informa la situación y detiene el proceso. - Datos personales insuficientes → el sistema solicita completar información.
Excepciones	<ul style="list-style-type: none"> - Errores inesperados durante el registro.
Postcondiciones	<ul style="list-style-type: none"> - La solicitud queda registrada con toda la información personal y documental. - El usuario puede consultar posteriormente el estado del trámite.

RF-006 — Registro de Comunicaciones

Código	RF-006
Nombre	Registro de Comunicaciones
Propósito	Permitir que el afiliado registre una comunicación formal hacia la obra social (reclamo, sugerencia o agradecimiento) mediante una interacción conversacional guiada.
Actores involucrados	<ul style="list-style-type: none"> - Afiliado - Agente conversacional
Precondiciones	<ul style="list-style-type: none"> - El afiliado debe estar identificado (RF-001). - Debe poseer un estado activo en la base de información institucional.
Flujo principal	<ol style="list-style-type: none"> 1. El afiliado solicita registrar un reclamo, sugerencia o agradecimiento. 2. El agente conversacional solicita una descripción del motivo. 3. El afiliado ingresa el texto correspondiente. 4. El agente valida que la descripción no esté vacía. 5. El agente registra la comunicación en el sistema. 6. El agente confirma que la comunicación fue registrada exitosamente. 7. El agente ofrece la posibilidad de consultar comunicaciones previas.

Flujos alternativos	<ul style="list-style-type: none"> - Información insuficiente → el agente solicita más información. - El afiliado decide cancelar el proceso → el agente lo confirma y finaliza el flujo.
Excepciones	<ul style="list-style-type: none"> - Error en el registro de la comunicación.
Postcondiciones	<ul style="list-style-type: none"> - La comunicación queda almacenada asociada al afiliado. - El afiliado puede consultar posteriormente sus comunicaciones registradas.

RF-007: Consulta de Disponibilidad de Topes

Código	RF-007
Nombre	Consulta de Disponibilidad de Topes
Propósito	Permitir que el afiliado consulte cuántas unidades disponibles le restan para una práctica médica con tope mensual, anual o por evento.
Actores involucrados	<ul style="list-style-type: none"> - Afiliado - Agente conversacional
Precondiciones	<ul style="list-style-type: none"> - El afiliado debe estar identificado (RF-001). - Debe conocer o describir la práctica a consultar. - Deben existir topes definidos para esa práctica según el plan del afiliado.
Flujo principal	<ol style="list-style-type: none"> 1. El afiliado solicita conocer la disponibilidad de una práctica. 2. El agente solicita el nombre o descripción de la práctica. 3. El afiliado ingresa la práctica. 4. El agente identifica la práctica dentro del catálogo institucional. 5. El agente obtiene el tope asociado al plan del afiliado. 6. El agente obtiene el consumo registrado hasta el momento. 7. El agente calcula el saldo disponible.

	8. El agente comunica al afiliado cuántas unidades le restan.
Flujos alternativos	<ul style="list-style-type: none"> - Práctica mal escrita → el agente solicita reformulación o propone coincidencias. - La práctica no tiene tope → el agente informa que no aplica ninguna limitación.
Excepciones	<ul style="list-style-type: none"> - Práctica inexistente. - Datos de consumo incompletos.
Postcondiciones	- El afiliado recibe información clara y actualizada sobre su disponibilidad.

RF-008: Consulta Histórica de Consumos

Código	RF-008
Nombre	Consulta Histórica de Consumos
Propósito	Permitir que el afiliado consulte las prácticas médicas registradas a su nombre durante un período reciente, incluyendo fechas, prestadores y códigos de práctica.
Actores involucrados	<ul style="list-style-type: none"> - Afiliado - Agente conversacional
Precondiciones	- El afiliado debe estar identificado (RF-001).
Flujo principal	<ol style="list-style-type: none"> 1. El afiliado solicita consultar sus consumos recientes. 2. El agente confirma el período de interés o utiliza un período por defecto. 3. El agente accede al historial de consumos del afiliado. 4. El agente organiza la información en una lista clara y ordenada. 5. El agente presenta el resultado al afiliado.

Flujos alternativos	<ul style="list-style-type: none"> - No existen consumos en el período → el agente informa que no hay registros. - El afiliado solicita un período diferente → el agente genera una nueva consulta.
Excepciones	<ul style="list-style-type: none"> - Fechas inválidas. - Datos inconsistentes en el historial.
Postcondiciones	<ul style="list-style-type: none"> - El afiliado recibe un listado legible y detallado de sus consumos recientes.

Requerimientos No Funcionales (RNF)

Los requerimientos no funcionales establecen las características de calidad que el sistema debe satisfacer para garantizar un funcionamiento adecuado, robusto y consistente. Estos requerimientos complementan a los funcionales y condicionan la arquitectura, la experiencia del usuario y la operación general del agente conversacional.

RNF-01 — Modularidad y Desacoplamiento

Código	RNF-01
Nombre	Modularidad y Desacoplamiento
Descripción	El sistema debe estar estructurado de forma modular, de modo que los componentes encargados de la interacción conversacional, la lógica de negocio y el acceso a datos permanezcan desacoplados entre sí.
Objetivo	Facilitar el mantenimiento, la escalabilidad y la posibilidad de reemplazar o actualizar componentes sin afectar al resto del sistema.
Criterios de cumplimiento	<ul style="list-style-type: none"> - Cada módulo debe ser independiente y cumplir una única responsabilidad.

	- La comunicación entre módulos debe realizarse mediante interfaces bien definidas.
--	---

RNF-02 — Persistencia de Contexto durante la conversación

Código	RNF-02
Nombre	Persistencia de Contexto durante la conversación
Descripción	El agente conversacional debe mantener el contexto del diálogo mientras la conversación del afiliado permanezca activa, recordando información como DNI o número de afiliado cuando sea pertinente.
Objetivo	Evitar solicitar repetidamente la misma información y asegurar fluidez en la interacción.
Criterios de cumplimiento	- El sistema recuerda datos relevantes sólo durante la conversación en curso. - Al finalizar la sesión, el contexto se limpia completamente.

RNF-03: Tiempo de Respuesta Adecuado

Código	RNF-03
Nombre	Tiempo de Respuesta Adecuado
Descripción	El sistema debe responder a las consultas del afiliado en tiempos razonables, garantizando una interacción conversacional fluida y sin demoras perceptibles.
Objetivo	Mantener una experiencia natural y continua, evitando interrupciones o sensación de bloqueo.
Criterios de cumplimiento	- El tiempo de respuesta promedio debe ser lo suficientemente corto para mantener el flujo conversacional.

	- Las respuestas que requieran múltiples pasos deben incluir mensajes intermedios que informen el estado del proceso.
--	---

RNF-05: Fiabilidad y Manejo Correcto de Errores

Código	RNF-05
Nombre	Fiabilidad y Manejo Correcto de Errores
Descripción	El sistema debe manejar fallas y errores de forma controlada, informando al afiliado mediante mensajes claros y evitando que el flujo conversacional entre en estados inconsistentes.
Objetivo	Mantener la integridad del proceso y la confianza del usuario en el sistema.
Criterios de cumplimiento	- Ante entradas inválidas, el sistema debe solicitar correcciones.- Cuando una operación no pueda completarse, el sistema debe informar claramente y ofrecer alternativas.

Diagramas

Los diagramas incluidos en esta sección complementan la especificación de requerimientos mediante representaciones visuales que permiten comprender el funcionamiento del sistema desde diferentes niveles de abstracción.

Diagrama de Contexto

El diagrama de contexto presentado a continuación representa la interacción global entre los elementos principales que conforman el sistema en su nivel más abstracto. Su objetivo es mostrar, desde un enfoque funcional, cómo se relacionan el afiliado, el agente conversacional y la base de datos que contiene la información necesaria para responder consultas o registrar solicitudes.

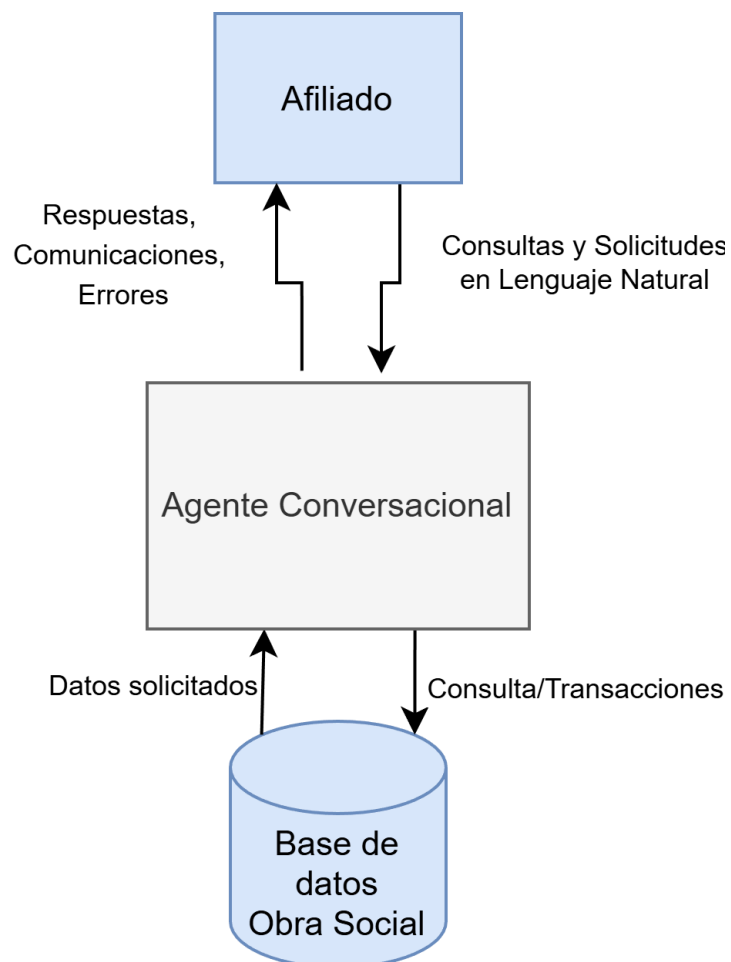


Diagrama de Casos de Uso

El diagrama de casos de uso presenta al Afiliado como el único actor externo que interactúa con el Sistema Conversacional. Desde la interfaz de chat, el usuario puede iniciar diferentes operaciones, que incluyen solicitar reintegros y solicitudes de afiliación, registrar comunicaciones formales, consultar prácticas cubiertas, verificar topes disponibles y revisar consumos realizados.

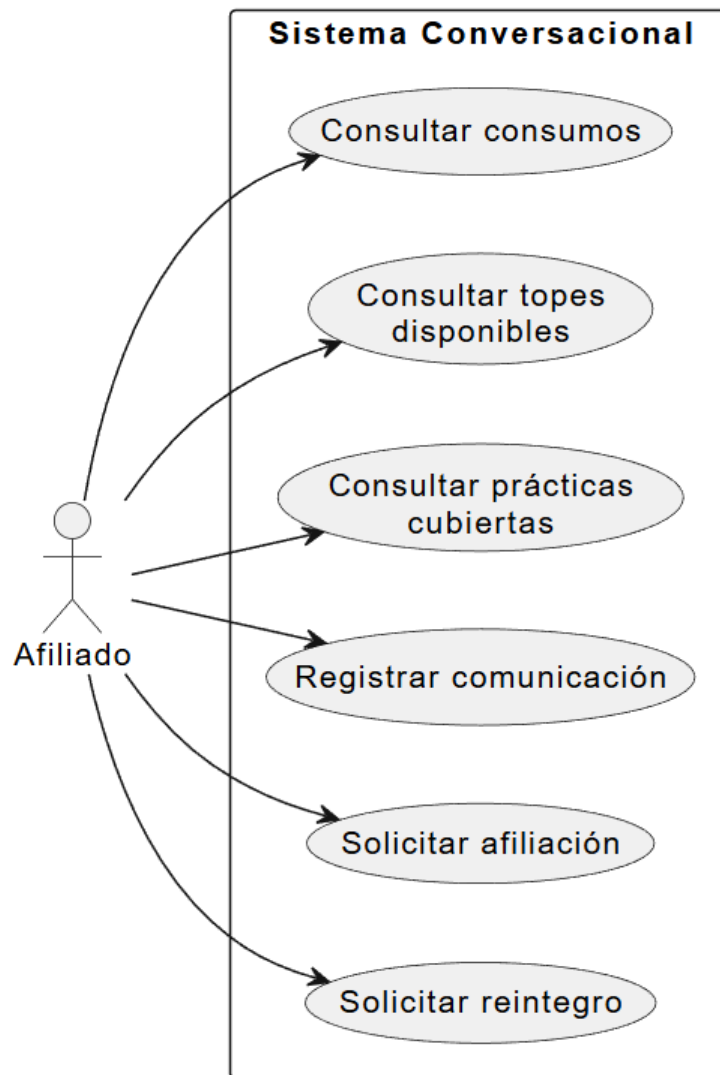


Diagrama de Actividad - Flujo Transaccional

El diagrama de actividad ilustra el flujo general de un trámite transaccional dentro del sistema, abarcando desde la expresión inicial de la intención del afiliado hasta el envío final de la solicitud.

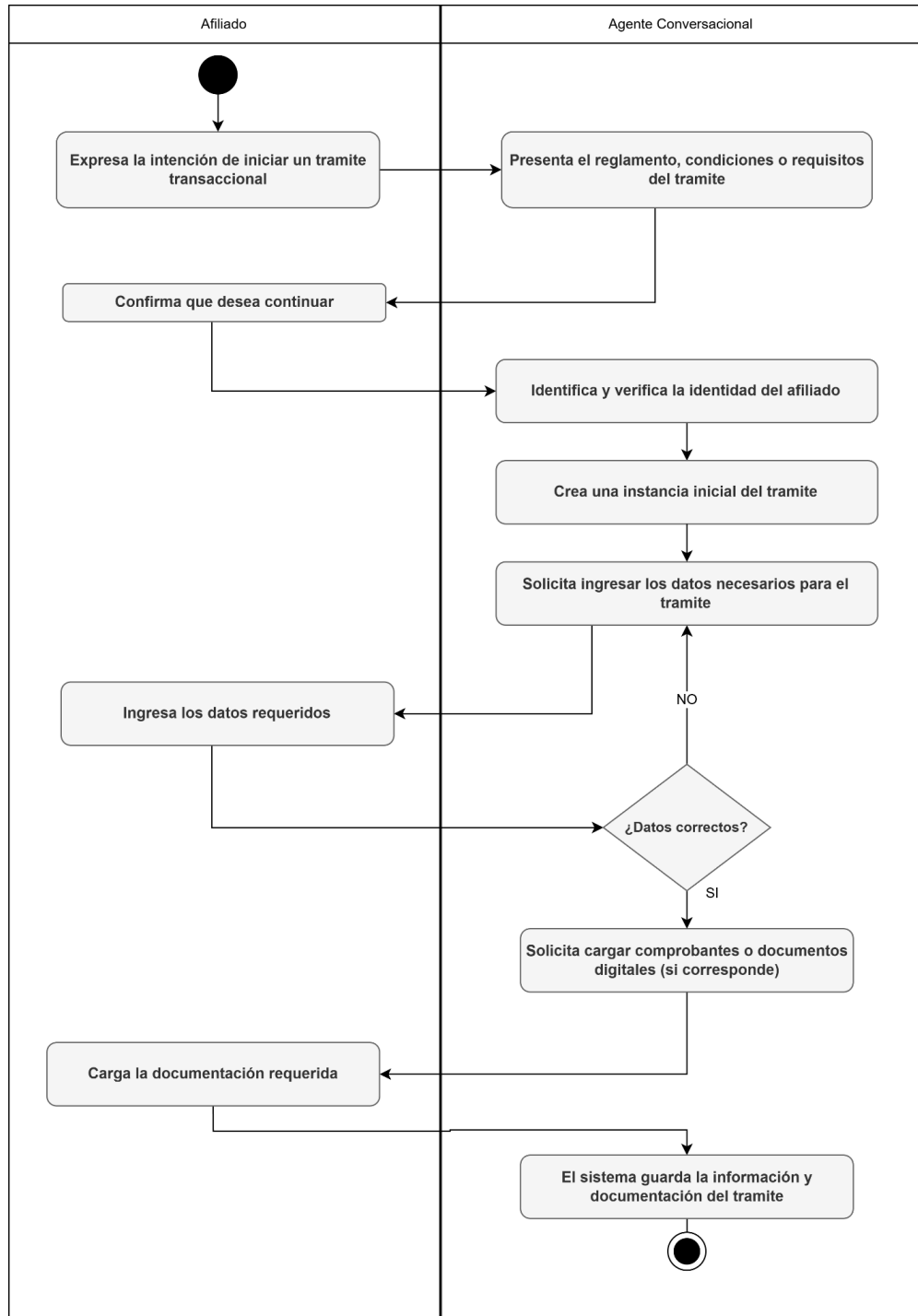


Diagrama de Actividad - Flujo de Consulta

Describe el flujo general de una operación de consulta dentro del Sistema Conversacional. El proceso comienza cuando el afiliado solicita acceder a determinada información.

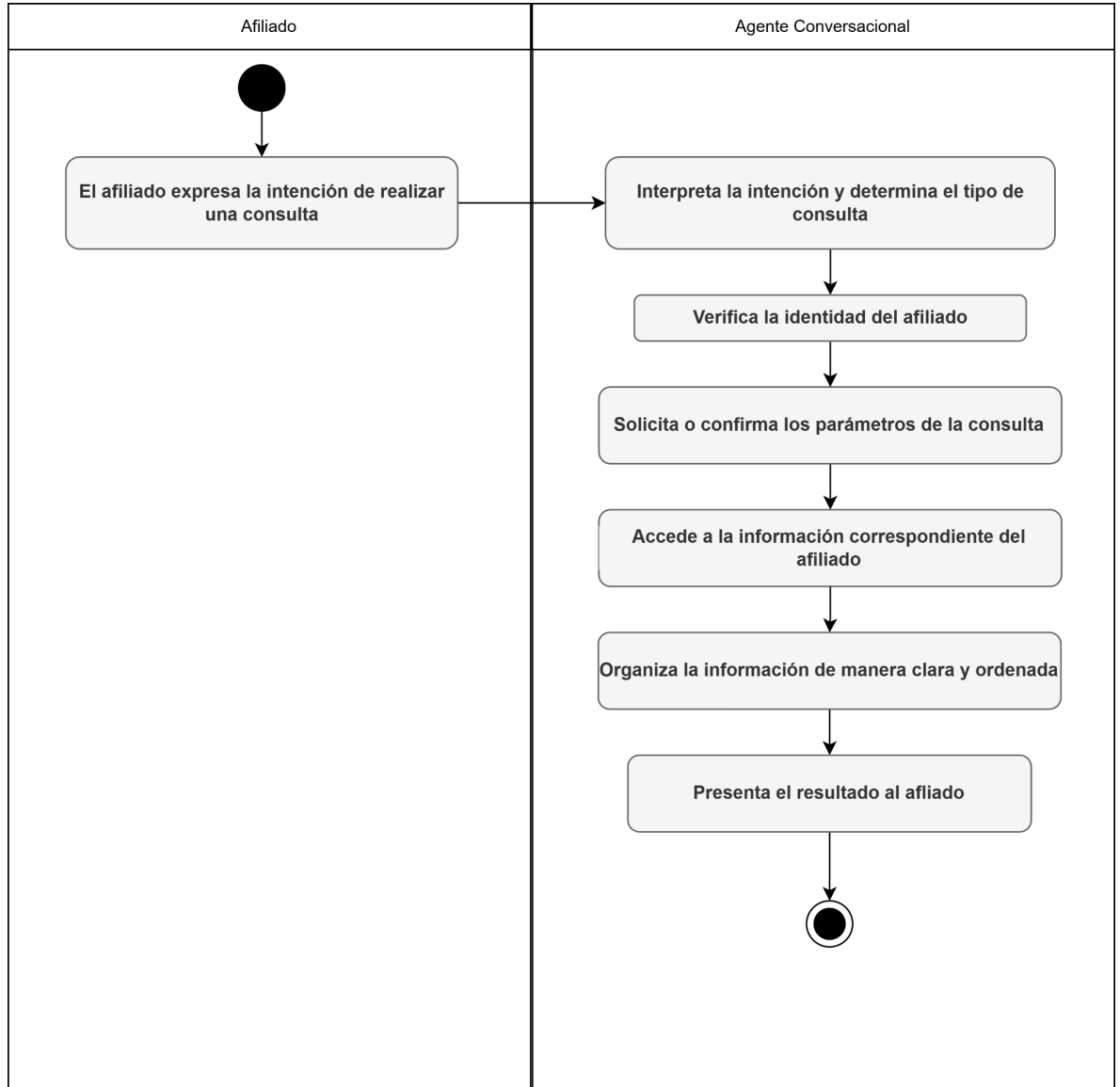


Diagrama de Secuencia - Transacción General

El diagrama de secuencia para el flujo transaccional muestra la interacción entre el Afiliado, el Agente Conversacional y la base de datos de la obra social durante la ejecución de un trámite.

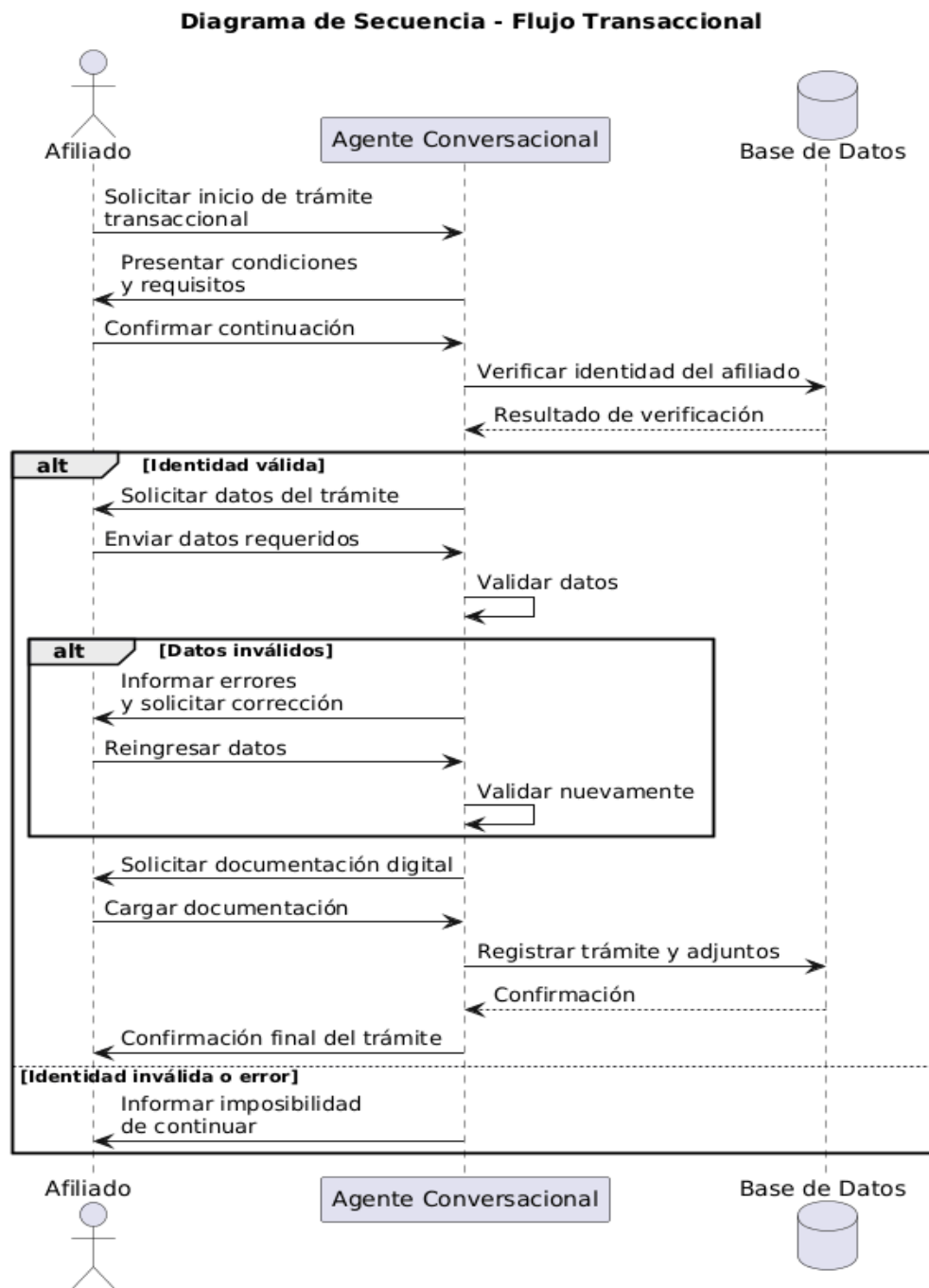
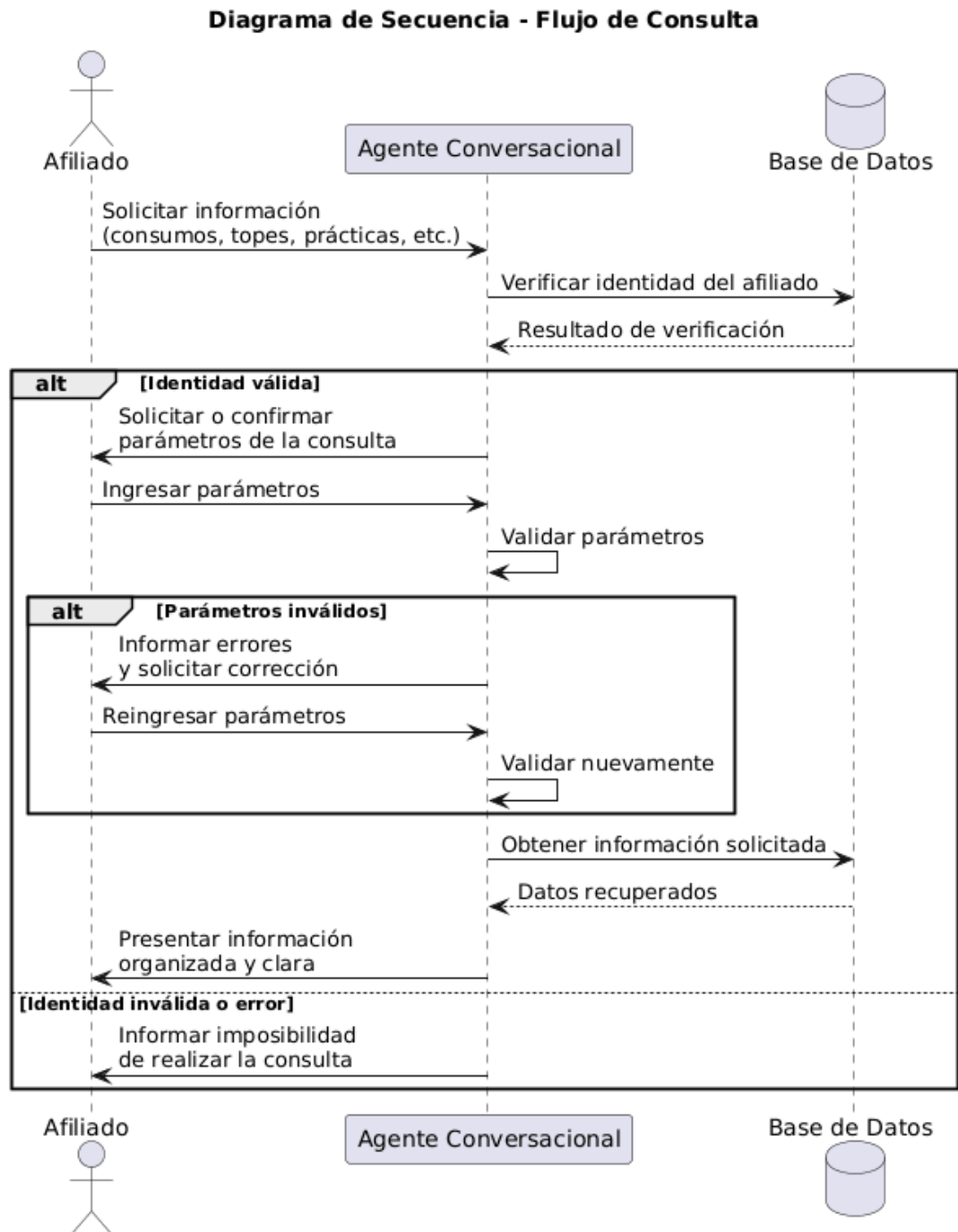


Diagrama de secuencia consulta general

El diagrama de secuencia del flujo de consulta muestra cómo el agente recibe la solicitud del afiliado, recupera la información necesaria y la presenta de forma clara, completando una operación de consulta sin modificar los datos existentes.



Especificación de Datos por Proceso

La presente sección define los datos conceptuales necesarios para la ejecución de cada proceso incluido dentro del alcance del sistema. Estos datos no representan estructuras físicas de base de datos, sino elementos lógicos indispensables para garantizar que un proceso pueda iniciarse, validarse y completarse correctamente.

Criterio de Elaboración

Para determinar los datos requeridos por cada proceso transaccional se realizó un análisis comparativo de la normativa publicada por cuatro obras sociales argentinas: OSEP, APOS, IOSFA e IPROSS.

Dado que cada institución posee reglamentos propios con particularidades en plazos, documentación y validaciones, se construyó un modelo conceptual unificado, basado en:

- los requisitos comunes presentes en la mayoría de los reglamentos analizados,
- los documentos obligatorios recurrentes,
- los datos administrativos necesarios para validar identidad, práctica y contexto,
- criterios de consistencia temporal y documental típicos del dominio.

Los demás procesos —consultas informativas y comunicaciones— no cuentan con reglamentación específica en las distintas obras sociales, por lo que fueron modelados en base a prácticas administrativas comunes y el funcionamiento operativo esperado en un sistema de consultas.

Simplificaciones adoptadas para el prototipo

Con el fin de mantener un desarrollo ágil y enfocado en los aspectos tecnológicos del agente conversacional, se aplicaron las siguientes simplificaciones sobre los procesos transaccionales:

- **Reintegros:** Se excluyeron datos que varían significativamente entre obras sociales, tales como información del prestador, diagnósticos clínicos, autorizaciones previas e informes complementarios. Se conservó únicamente el conjunto mínimo necesario para representar el proceso administrativo básico: tipo de práctica, fecha, monto y comprobantes esenciales.

- **Afiliación:** Se modeló únicamente la afiliación de titulares adultos, omitiendo las modalidades de hijos, cónyuges o adherentes, debido a la complejidad documental que presentan y a la inexistencia de un estándar común entre instituciones. Asimismo, se reemplazó el “estado civil” por un tipo de afiliado simplificado (empleado, monotributista, jubilado o particular), suficiente para clasificar la solicitud sin replicar estructuras administrativas propias de cada obra social.

Estas decisiones metodológicas permiten mantener un equilibrio adecuado entre fidelidad del dominio y viabilidad de implementación, acorde al alcance del prototipo y a los objetivos tecnológicos de la tesis.

Datos necesarios para el Proceso de Solicitud de Reintegro

El proceso de reintegro fue simplificado respecto del modelo normativo real, manteniendo únicamente la información indispensable para representar el flujo básico sin reproducir requisitos específicos de cada obra social.

Datos de identificación

- DNI
- Número de afiliado
- Datos del reintegro
- Tipo de práctica
- Fecha de realización
- Monto declarado

Documentación obligatoria (simplificada)

- Factura o comprobante en formato PDF
- Orden o constancia que respalde la práctica (si aplica)

Observaciones del modelo simplificado

- No se solicita diagnóstico médico.
- No se solicita información del prestador.
- No se validan autorizaciones previas.
- No se requiere informe complementario ni resultados de estudios.

Se mantiene únicamente el requisito de carga de comprobantes básicos para validar la solicitud.

Datos necesarios para el Proceso de Solicitud de Afiliación

Este proceso se modeló exclusivamente para titulares adultos, evitando ramificaciones propias de adherentes, hijos, cónyuges u otras categorías presentes en la normativa real.

Datos personales obligatorios

- DNI
- Nombre y apellido
- Fecha de nacimiento
- Domicilio
- Datos de contacto (teléfono o correo electrónico)

Datos administrativos

- Tipo de afiliado:
 - Empleado
 - Monotributista
 - Jubilado
 - Particular

(Esta clasificación reemplaza al estado civil y a la documentación vinculada a grupos familiares.)

Documentación obligatoria

- Fotografía del DNI (frente y dorso)
- Comprobante de domicilio o servicio (opcional según el caso)

Observaciones del modelo simplificado

- No se requiere constancia de vínculo familiar.
- No se solicita documentación laboral específica.
- No se contemplan adherentes ni cargas familiares.
- No se gestionan categorías especiales ni planes derivados.

Datos para Registro de Comunicaciones

- Identificación del afiliado
- Tipo de comunicación (reclamo, sugerencia, agradecimiento)
- Descripción del mensaje

Datos para Consulta de Topes

- Identificación del afiliado
- Práctica a consultar
- Período asociado al tope (mensual/anual/evento)

Datos para Consulta de Consumos

- Identificación del afiliado
- (Opcional) Período solicitado

Datos para Verificación de Cobertura

- Identificación del afiliado
- Nombre o descripción de la práctica

Glosario de Términos

El glosario reúne los conceptos esenciales utilizados en el dominio de una obra social y en el funcionamiento del agente conversacional. Su propósito es asegurar que todos los términos empleados en los requerimientos sean comprendidos de forma consistente y no ambigua.

Afiliado

Persona registrada en la obra social que posee un plan activo y puede realizar consultas y trámites mediante el agente conversacional.

Agente Conversacional

Sistema de interacción en lenguaje natural que interpreta consultas, guía trámites y accede a la información necesaria para responder o registrar solicitudes.

Práctica Médica

Procedimiento, prestación o servicio de salud reconocido por la obra social y registrado en el catálogo institucional.

Cobertura

Condiciones bajo las cuales la obra social reconoce o financia una práctica médica, incluyendo copagos, límites o requisitos adicionales.

Tope

Cantidad máxima permitida para una práctica médica dentro de un período determinado (mensual, anual o por evento).

Consumo

Registro de una práctica realizada por el afiliado, contabilizada para efectos de límites y cálculos de tope.

Reintegro

Devolución de un gasto en salud realizada por la obra social al afiliado, previa presentación de comprobantes y validación de la información.

Solicitud de Afiliación

Proceso mediante el cual una persona inicia su ingreso como afiliado a la obra social, aportando información personal y documentación.

Comunicación

Mensaje enviado por el afiliado a la obra social para registrar un reclamo, sugerencia o agradecimiento.

Documentación Obligatoria

Archivos requeridos para completar un trámite (por ejemplo, DNI, comprobantes, facturas o constancias).

Catálogo de Prácticas

Listado oficial de prácticas médicas reconocidas por la obra social, junto con sus códigos y descripciones.

Plan de Salud

Conjunto de prestaciones, coberturas, límites y beneficios asignados al afiliado según su tipo de plan.

Este glosario constituye la referencia terminológica oficial del anexo, asegurando un marco conceptual claro para interpretar los requerimientos.

Referencias

La presente sección reúne las fuentes documentales consultadas para el análisis comparativo de los procesos administrativos de afiliación y reintegros utilizado en la elaboración del modelo conceptual unificado. Si bien los procedimientos reales de cada obra social poseen particularidades propias, la información contenida en estos documentos permitió identificar elementos comunes y estructurar un conjunto de datos mínimo, adecuado para el diseño del prototipo desarrollado en este trabajo.

Las referencias aquí incluidas corresponden a documentación pública disponible en los portales institucionales de cada organismo.

OSEP — Obra Social de Empleados Públicos

Requisitos para Afiliación de Titulares de Administración Pública.

Disponible en:

https://www.osep.gob.ar/osep/archivos/requisitostramites/afil/Afil_TitAdmPublica.pdf

Reintegros — Gastos Médicos y Cirugías e Internaciones.

Disponible en:

https://www.osep.gob.ar/osep/archivos/requisitostramites/reintegro/Reintegro_GastosMedicos_CxInternaciones.pdf

APOS — Administración Provincial de Obra Social

Información general sobre afiliaciones.

Disponible en:

<https://aposlr.gob.ar/afiliaciones-2/>

IOSFA — Instituto de Obra Social de las Fuerzas Armadas y de Seguridad

Procedimiento de Afiliación.

Disponible en:

<https://iosfa.gob.ar/formularios/afiliaciones/95/afiliarse>

Procedimiento de Reintegros.

Disponible en:

<https://iosfa.gob.ar/formularios/tramites/139/reintegros>

IPROSS — Instituto Provincial del Seguro de Salud (Río Negro)

Información para Nuevos Afiliados.

Disponible en:

<https://ipross.rionegro.gov.ar/info/57/ipross-nuevos-afiliados?n=MzQ1>

Procedimiento y requisitos para Reintegros.

Disponible en:

<https://ipross.rionegro.gov.ar/info/136/reintegros-informacion>

Nota Final del Anexo

Las fuentes citadas en esta sección fueron utilizadas exclusivamente con fines de análisis comparativo, para identificar patrones comunes entre obras sociales y derivar un conjunto de datos conceptual aplicable al prototipo desarrollado.

El sistema implementado no replica normativas particulares, ni tiene por objetivo sustituir procedimientos administrativos reales, sino formar un modelo académico representativo y simplificado, adecuado al alcance de una tesis de ingeniería orientada a agentes conversacionales.

ANEXO B: ACTA DE VALIDACIÓN DE REQUERIMIENTOS

ACTA GENERAL DE VALIDACIÓN DE PROCESOS – ENTREVISTAS TÉCNICAS

Fecha: 15/10/2025

Lugar: San Fernando Del Valle de Catamarca

Duración total aproximada: 45 minutos

Tesista: Miguel Bernardo Cejas Romero – Ingeniería en Informática

Propósito del acta

El objetivo de esta acta es dejar constancia de la validación técnica realizada por tres responsables de módulos del sistema informático de obras sociales, quienes revisaron los procesos modelados en el prototipo desarrollado para la tesis. La finalidad de dicha validación es confirmar que la representación simplificada utilizada en el prototipo mantiene coherencia con los procedimientos reales, aun cuando el modelo fue reducido en alcance y complejidad para adecuarse al tiempo de desarrollo y a los objetivos académicos del trabajo.

Documentación presentada a los entrevistados

El tesista presentó a los participantes el **Documento de Requerimientos**, que contiene:

- La descripción de los procesos incluidos en el alcance del prototipo.
- Diagramas UML (actividades, secuencia y contexto) elaborados a partir del análisis de los flujos institucionales.
- Las reglas de negocio esenciales, validaciones mínimas y condiciones previas.
- El modelo de datos simplificado basado en entidades centrales del dominio.
- La descripción de las herramientas MCP que implementan cada operación en el prototipo.

Durante la revisión, el tesista aclaró que tanto los procesos como el modelo de datos fueron deliberadamente simplificados, preservando el funcionamiento conceptual, pero omitiendo variaciones, excepciones y detalles operativos presentes en los sistemas reales.

Metodología aplicada

La validación se realizó mediante entrevistas semiestructuradas. Esta metodología permitió seguir una guía común de preguntas orientada a evaluar la coherencia del modelado, dejando al mismo tiempo espacio para comentarios técnicos y observaciones sobre las simplificaciones introducidas en el prototipo.

Guía de preguntas utilizada

Las entrevistas se llevaron a cabo utilizando la siguiente guía:

- ¿El proceso representado conserva la lógica central del procedimiento real?
- ¿Las validaciones obligatorias (identidad, afiliación, documentación, topes, consumos) están correctamente planteadas?
- ¿Las reglas de negocio esenciales están adecuadamente reflejadas en esta versión simplificada?
- ¿Los diagramas UML representan la estructura general del flujo, aun omitiendo particularidades operativas?
- ¿El modelo de datos simplificado es consistente con la lógica del dominio, aunque reduzca detalles?
- ¿Los estados finales y transiciones son coherentes con los resultados habituales del proceso real?

Procesos revisados

Los tres participantes revisaron y evaluaron los siguientes procesos:

1. Afiliación de nuevos beneficiarios (incluyendo carga de documentación).
2. Solicitud de reintegros (incluyendo carga de documentación e ítems).
3. Consultas de cobertura.
4. Consultas de topes y consumos.
5. Registro de comunicaciones.
6. Revisión del modelo de datos.

Conclusiones generales de los entrevistados

Los tres participantes coinciden en que:

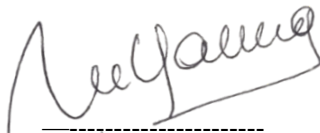
- Los procesos presentados mantienen la estructura lógica esencial de la operatoria real, aun cuando fueron adaptados y simplificados.
- El nivel de simplificación aplicado es adecuado para el propósito académico de la tesis y para las limitaciones del prototipo desarrollado.
- Las reglas de negocio fundamentales están correctamente representadas.
- Los diagramas UML permiten comprender la lógica general del proceso, incluso sin contemplar todas las variantes reales.
- El modelo de datos simplificado respeta la organización funcional del dominio y resulta suficiente para los casos incluidos en el alcance del prototipo.
- Las diferencias entre el sistema real y el modelo académico fueron aclaradas y no afectan la validez conceptual de los procesos.

Constancia de validación


Los abajo firmantes dejan constancia de que revisaron la documentación presentada por el tesista, comprendiendo que se trata de un modelo reducido en complejidad y diseñado exclusivamente para fines académicos. Aun así, validan que la descripción de los procesos, el modelado y el flujo general reflejan de manera razonable y coherente la operatoria real

del sistema informático de obras sociales. Autorizan su incorporación en el documento de tesis como material correctamente fundamentado.

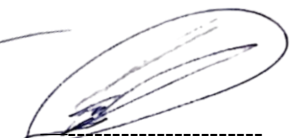
Firmas de los participantes:



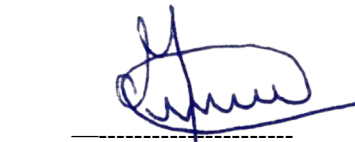
Lic. Maria Virginia Garcia
Delivery Manager
Tekhne



Lic. Maria Nelly Rodriguez
**Responsable de Módulo de
Gestión de Trámites**
Tekhne



Ing. Patricio Pinetta
**Responsable de Módulo
Prestacional**
Tekhne



Miguel Bernardo Cejas Romero
Tesista

ANEXO C: ESQUEMA DE BASE DE DATOS

Introducción

La base de datos implementada para el proyecto de agente conversacional (integrado con el servidor MCP) modela la información principal de una obra social/prepaga y los procesos que el agente asiste: solicitudes de afiliación, coberturas de prácticas, consumos, solicitudes de reintegros y comunicaciones de los afiliados.

El objetivo del esquema es:

- Centralizar los datos maestros de afiliados, planes, prácticas y medicamentos.
- Registrar las solicitudes de **afiliación** y los **documentos** asociados.
- Soportar la gestión de **reintegros** (cabecera e ítems), así como sus documentos respaldatorios.
- Registrar **consumos** de prestaciones y **comunicaciones** (reclamos, quejas etc.) de los afiliados.
- Definir reglas de **cobertura** (porcentaje, copago y topes) por plan y por práctica/medicamento.
- Brindar mecanismos de **verificación de identidad** vía códigos OTP para acceso de afiliados.

El agente conversacional utiliza esta base como fuente de verdad para responder consultas, validar datos y registrar operaciones a través de herramientas expuestas por el servidor MCP.

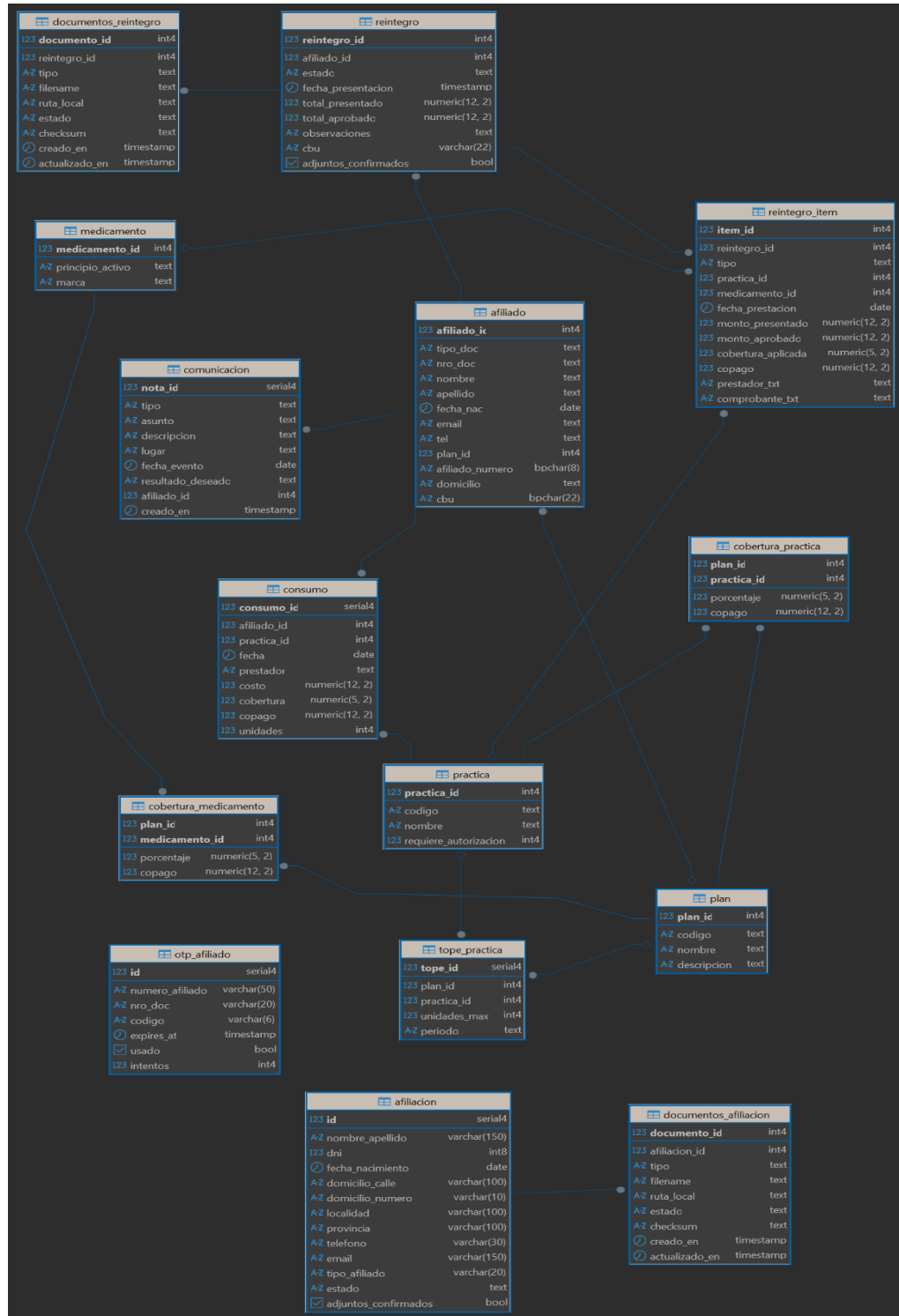
Entidades principales

Las entidades centrales del modelo son:

- **afiliado**: persona ya incorporada a la obra social, con un plan asignado.
- **plan**: conjunto de coberturas contratadas por el afiliado.
- **practica** y **medicamento**: catálogos de prestaciones y fármacos.
- **cobertura_practica** y **cobertura_medicamento**: reglas de cobertura por plan.
- **consumo**: registro de prestaciones efectivamente utilizadas.
- **reintegro** y **reintegro_item**: solicitudes de reintegro y sus ítems.
- **afiliacion** y **documentos_afiliacion**: proceso de alta de afiliado y sus adjuntos.
- **documentos_reintegro**: documentos adjuntos a las solicitudes de reintegro.
- **comunicacion**: reclamos, sugerencias y agradecimientos de afiliados.
- **tope_practica**: límites de unidades por periodo para ciertas prácticas.
- **otp_afiliado**: códigos de un solo uso para validar identidad del afiliado.

Diagrama Entidad-Relación

A continuación se muestra el diagrama generado desde DBeaver, el cual incluye todas las entidades, sus campos, tipos de datos y las relaciones existentes entre ellas:



Descripción de tablas y campos

Tabla **afiliacion**

Registra solicitudes de afiliación (pre–alta).

Campos principales

- **id** (serial, PK): Identificador único de la solicitud.
- **nombre_apellido** (varchar(150), NOT NULL): Nombre completo del solicitante.
- **dni** (int8, NOT NULL, único, > 0): Documento nacional de identidad.
- **fecha_nacimiento** (date, NOT NULL, check \geq 18 años): Fecha de nacimiento.
- **domicilio_calle** (varchar(100), NOT NULL): Calle del domicilio.
- **domicilio_numero** (varchar(10), NOT NULL): Altura del domicilio.
- **localidad** (varchar(100), NOT NULL): Localidad de residencia.
- **provincia** (varchar(100), NOT NULL): Provincia de residencia.
- **telefono** (varchar(30), NULL): Teléfono de contacto.
- **email** (varchar(150), NULL): Correo electrónico de contacto.
- **tipo_afiliado** (varchar(20), NOT NULL, check): Tipo de afiliado:
 - EMPLEADO, MONOTRIBUTISTA, JUBILADO, PARTICULAR.
- **estado** (text, NOT NULL, default 'PENDIENTE'): Estado del trámite.
- **adjuntos_confirmados** (bool, NOT NULL, default **false**): Indica si la documentación adjunta fue verificada.

Tabla **documentos_afiliacion**

Documentos adjuntos a una solicitud de afiliación.

- **documento_id** (int4, PK, identity): Identificador del documento.
- **afiliacion_id** (int4, NOT NULL, FK \rightarrow **afiliacion.id**): Solicitud asociada.
- **tipo** (text, NOT NULL): Tipo de documento (DNI, recibo de sueldo, etc., según la lógica de negocio).
- **filename** (text, NOT NULL): Nombre del archivo subido.
- **ruta_local** (text, NOT NULL): Ruta o ubicación en el sistema de archivos.
- **estado** (text, NOT NULL, default 'recibido', check):

- Valores válidos: **recibido, vinculado, reemplazado**.
- **checksum** (text, NULL): Hash del archivo para verificación de integridad.
- **creado_en** (timestamp, NOT NULL, default **now()**): Fecha de creación.
- **actualizado_en** (timestamp, NOT NULL, default **now()**): Última actualización.

Tabla **afiliado**

Afiliados activos de la obra social.

- **afiliado_id** (int4, PK, identity): Identificador interno del afiliado.
- **tipo_doc** (text, NOT NULL): Tipo de documento (DNI, LE, LC, etc.).
- **nro_doc** (text, NOT NULL): Número de documento.
- **nombre** (text, NOT NULL): Nombre/s del afiliado.
- **apellido** (text, NOT NULL): Apellido/s del afiliado.
- **fecha_nac** (date, NULL): Fecha de nacimiento.
- **email** (text, NULL): Correo electrónico de contacto.
- **tel** (text, NULL): Teléfono de contacto.
- **plan_id** (int4, NULL, FK → **plan.plan_id**): Plan de cobertura vigente.
- **afiliado_numero** (bpchar(8), NOT NULL, único, check formato): Número de afiliado, 8 dígitos numéricos.
- **domicilio** (text, NULL): Domicilio (texto libre).
- **cbu** (bpchar(22), NULL): CBU para pagos/reintegros.

Tabla **plan**

Catálogo de planes de cobertura.

- **plan_id** (int4, PK, identity): Identificador del plan.
- **codigo** (text, NOT NULL, único): Código interno/comercial del plan.
- **nombre** (text, NOT NULL): Nombre comercial del plan.
- **descripcion** (text, NULL): Descripción general.

Tabla **práctica**

Catálogo de prácticas médicas.

- **practica_id** (int4, PK, identity): Identificador de la práctica.
- **codigo** (text, NOT NULL, único): Código arancelario o nomenclador.
- **nombre** (text, NOT NULL): Descripción de la práctica.
- **requiere_autorizacion** (int4, NULL, default 0): Indicador (0/1) de si requiere autorización previa.

Tabla **medicamento**

Catálogo de medicamentos.

- **medicamento_id** (int4, PK, identity): Identificador del medicamento.
- **principio_activo** (text, NOT NULL): Nombre del principio activo.
- **marca** (text, NULL): Marca o nombre comercial.

Tabla **cobertura_practica**

Define la cobertura para prácticas según plan (tabla de relación N:M entre **plan** y **practica**).

- **plan_id** (int4, NOT NULL, PK, FK → **plan.plan_id**).
- **practica_id** (int4, NOT NULL, PK, FK → **practica.practica_id**).
- **porcentaje** (numeric(5,2), NOT NULL): Porcentaje de cobertura de la obra social.
- **copago** (numeric(12,2), NULL, default 0): Monto de copago a cargo del afiliado.

Tabla **cobertura_medicamento**

Cobertura de medicamentos por plan (también N:M).

- **plan_id** (int4, NOT NULL, PK, FK → **plan.plan_id**).
- **medicamento_id** (int4, NOT NULL, PK, FK → **medicamento.medicamento_id**).
- **porcentaje** (numeric(5,2), NOT NULL): Porcentaje cubierto.
- **copago** (numeric(12,2), NULL, default 0): Copago a cargo del afiliado.

Tabla **consumo**

Registra consumos de prácticas por parte de los afiliados.

- **consumo_id** (serial4, PK): Identificador del consumo.
- **afiliado_id** (int4, NOT NULL, FK → **afiliado.afiliado_id**): Afiliado que realizó la práctica.
- **practica_id** (int4, NOT NULL, FK → **practica.practica_id**): Práctica realizada.
- **fecha** (date, NOT NULL): Fecha de la prestación.
- **prestador** (text, NULL): Nombre del prestador o institución.
- **costo** (numeric(12,2), NULL): Costo total de la prestación.
- **cobertura** (numeric(5,2), NULL): Porcentaje de cobertura aplicado.
- **copago** (numeric(12,2), NULL): Copago efectivamente pagado por el afiliado.
- **unidades** (int4, NULL, default 1): Cantidad de unidades (sesiones, etc.).

Tabla **reintegro**

Cabecera de solicitudes de reintegro.

- **reintegro_id** (int4, PK, identity): Identificador del reintegro.
- **afiliado_id** (int4, NOT NULL, FK → **afiliado.afiliado_id**): Afiliado solicitante.
- **estado** (text, NOT NULL, default 'PENDIENTE'): Estado de la solicitud (pendiente, aprobado, rechazado, etc., según la lógica de negocio).
- **fecha_presentacion** (timestamp, NOT NULL, default **CURRENT_TIMESTAMP**): Fecha/hora de presentación.
- **total_presentado** (numeric(12,2), NULL, default 0): Importe total presentado por el afiliado.
- **total_aprobado** (numeric(12,2), NULL, default 0): Importe aprobado para reintegro.
- **observaciones** (text, NULL): Notas internas o comentario de auditoría.
- **cbu** (varchar(22), NOT NULL): CBU donde se acreditará el reintegro.
- **adjuntos_confirmados** (bool, NOT NULL, default **false**): Indica si la documentación se validó.

Tabla **reintegro_item**

Ítems individuales de una solicitud de reintegro (pueden ser prácticas o medicamentos).

- **item_id** (int4, PK, identity): Identificador del ítem.

- **reintegro_id** (int4, NOT NULL, FK → **reintegro.reintegro_id**): Reintegro al que pertenece.
- **tipo** (text, NOT NULL, check): Tipo de ítem:
 - **practica** o **medicamento**.
- **practica_id** (int4, NULL, FK → **practica.practica_id**): Práctica asociada (si **tipo** = 'practica').
- **medicamento_id** (int4, NULL, FK → **medicamento.medicamento_id**): Medicamento asociado (si **tipo** = 'medicamento').
- **fecha_prestacion** (date, NOT NULL): Fecha de la prestación/compra.
- **monto_presentado** (numeric(12,2), NOT NULL): Importe presentado en el comprobante.
- **monto_aprobado** (numeric(12,2), NULL): Importe aprobado para reintegro.
- **cobertura_aplicada** (numeric(5,2), NULL): Porcentaje aplicado.
- **copago** (numeric(12,2), NULL): Copago resultante (si aplica).
- **prestador_txt** (text, NULL): Descripción textual del prestador.
- **comprobante_txt** (text, NULL): Descripción del comprobante (número, tipo, etc.).

Tabla **documentos_reintegro**

Documentación respaldatoria de un reintegro (facturas, órdenes, etc.).

- **documento_id** (int4, PK, identity): Identificador del documento.
- **reintegro_id** (int4, NOT NULL): Reintegro asociado (relación lógica con **reintegro.reintegro_id**).
- **tipo** (text, NOT NULL): Tipo de documento (factura, receta, orden médica, etc.).
- **filename** (text, NOT NULL): Nombre del archivo.
- **ruta_local** (text, NOT NULL): Ruta de almacenamiento.
- **estado** (text, NOT NULL, default 'recibido', check):
 - **recibido**, **vinculado**, **reemplazado**.
- **checksum** (text, NULL): Hash del archivo.
- **creado_en** (timestamp, NOT NULL, default **now()**): Fecha de creación.
- **actualizado_en** (timestamp, NOT NULL, default **now()**): Última actualización.

Tabla **comunicacion**

Registra comunicaciones de los afiliados (reclamos, sugerencias, agradecimientos).

- **nota_id** (serial4, PK): Identificador de la comunicación.
- **tipo** (text, NOT NULL, check): Tipo de comunicación:
 - **AGRADECIMIENTO, SUGERENCIA, RECLAMO.**
- **asunto** (text, NULL): Asunto o título.
- **descripcion** (text, NOT NULL): Detalle de la comunicación.
- **lugar** (text, NULL): Lugar relacionado (sucursal, clínica, etc.).
- **fecha_evento** (date, NULL): Fecha del hecho referido.
- **resultado_deseado** (text, NULL): Resultado esperado por el afiliado.
- **afiliado_id** (int4, NOT NULL, FK → **afiliado.afiliado_id**): Afiliado que realiza la comunicación.
- **creado_en** (timestamp, NULL, default **CURRENT_TIMESTAMP**): Fecha de registro.

Tabla **tope_practica**

Define topes de uso de prácticas por plan y periodo.

- **tope_id** (serial4, PK): Identificador del tope.
- **plan_id** (int4, NULL, FK → **plan.plan_id**): Plan al que aplica el tope. Puede ser nulo si el tope es global.
- **practica_id** (int4, NULL, FK → **practica.practica_id**): Práctica a la que aplica.
- **unidades_max** (int4, NOT NULL): Cantidad máxima de unidades.
- **periodo** (text, NOT NULL, check): Periodo del tope:
 - **mensual** o **anual**.

Tabla **otp_afiliado**

Códigos de un solo uso para validación/autenticación de afiliados.

- **id** (serial4, PK): Identificador interno.
- **numero_afiliado** (varchar(50), NOT NULL): Número de afiliado al que se envía el OTP.
- **nro_doc** (varchar(20), NOT NULL): Documento del afiliado (para validación cruzada).
- **codigo** (varchar(6), NOT NULL): Código OTP (habitualmente 6 dígitos).

- `expires_at` (timestamp, NOT NULL): Fecha/hora de expiración.
- `usado` (bool, NOT NULL, default `false`): Indica si ya fue utilizado.
- `intentos` (int4, NOT NULL, default 0): Cantidad de intentos de validación.

Relaciones entre tablas

A continuación se resumen las relaciones clave del modelo:

- **afiliado – plan**

- Relación: muchos afiliados pertenecen a un plan.
- Implementación: `afiliado.plan_id` → `plan.plan_id` (N:1).

- **afiliado – comunicacion**

- Relación: un afiliado puede realizar muchas comunicaciones.
- Implementación: `comunicacion.afiliado_id` → `afiliado.afiliado_id` (1:N).

- **afiliado – consumo**

- Relación: un afiliado puede registrar muchos consumos.
- Implementación: `consumo.afiliado_id` → `afiliado.afiliado_id` (1:N).

- **afiliado – reintegro**

- Relación: un afiliado puede tener múltiples solicitudes de reintegro.
- Implementación: `reintegro.afiliado_id` → `afiliado.afiliado_id` (1:N).

- **plan – cobertura_practica – practica**

- Relación: N:M entre planes y prácticas, con atributos propios (porcentaje, copago).
- Implementación:
 - `cobertura_practica.plan_id` → `plan.plan_id`.

- `cobertura_practica.practica_id` → `practica.practica_id`.
- **plan – cobertura_med medicamento – medicamento**
 - Relación: N:M entre planes y medicamentos, con atributos de cobertura.
 - Implementación:
 - `cobertura_med medicamento.plan_id` → `plan.plan_id`.
 - `cobertura_med medicamento.med medicamento_id` → `medicamento.med medicamento_id`.
- **plan – tope_practica – practica**
 - Relación: para ciertos planes y prácticas se definen topes de unidades por periodo.
 - Implementación:
 - `tope_practica.plan_id` → `plan.plan_id`.
 - `tope_practica.practica_id` → `practica.practica_id`.
- **practica – consumo**
 - Relación: cada consumo refiere a una práctica.
 - Implementación: `consumo.practica_id` → `practica.practica_id` (N:1).
- **reintegro – reintegro_item**
 - Relación: un reintegro tiene uno o varios ítems.
 - Implementación: `reintegro_item.reintegro_id` → `reintegro.reintegro_id` (1:N).
- **reintegro_item – practica / medicamento**
 - Relación: cada ítem se vincula a una práctica o a un medicamento, según el tipo.

- Implementación:

- `reintegro_item.practica_id` → `practica.practica_id`.
- `reintegro_item.medicamento_id` → `medicamento.medicamento_id`.

- **reintegro – documentos_reintegro**

- Relación lógica: un reintegro puede tener múltiples documentos adjuntos.
- Implementación: campo `documentos_reintegro.reintegro_id` asociado a `reintegro.reintegro_id` (1:N).

- **afiliacion – documentos_afiliacion**

- Relación: una solicitud de afiliación puede tener múltiples documentos.
- Implementación: `documentos_afiliacion.afiliacion_id` → `afiliacion.id` (1:N).

- **otp_afiliado**

- No tiene claves foráneas explícitas, pero se relaciona lógicamente con la entidad `afiliado` a través de `numero_afiliado` y `nro_doc`, para validar la identidad durante procesos asistidos por el agente conversacional.

ANEXO D: VALIDACIÓN FUNCIONAL DEL PROTOTIPO

Introducción

La validación funcional tuvo como objetivo comprobar que el prototipo desarrollado cumpliera con los requerimientos definidos y que los procesos seleccionados pudieran ejecutarse de forma coherente a través del agente conversacional. Dado que se trata de una prueba de concepto, el foco de esta instancia no fue medir rendimiento en condiciones de producción, sino verificar que los flujos principales se comportaran según lo previsto en el diseño.

Las pruebas se realizaron en un entorno controlado, utilizando datos ficticios cargados en la base de datos simulada. Esto permitió reproducir los escenarios necesarios sin comprometer información real y garantizando que los resultados fueran comparables entre diferentes ejecuciones.

Para organizar la validación, se definieron escenarios de prueba basados en los procesos relevados en las fases iniciales del proyecto (consultas, trámites y manejo de errores). A partir de esos escenarios se construyeron casos de prueba que describen, para cada situación, las entradas a utilizar, el comportamiento esperado del sistema y los criterios de aceptación correspondientes.

En las secciones siguientes se detallan los escenarios evaluados y los resultados obtenidos.

Metodología de validación funcional

La validación funcional se realizó mediante técnicas de prueba de caja negra, enfocadas exclusivamente en las entradas proporcionadas por el usuario y las respuestas observables del prototipo. Este enfoque resulta adecuado para un sistema conversacional, ya que permite evaluar si el agente comprende correctamente las solicitudes, ejecuta los procesos definidos y retorna información coherente sin necesidad de inspeccionar la lógica interna del modelo o del backend.

Para estructurar la validación se definieron escenarios representativos basados en los procesos incluidos en el alcance del prototipo: identificación del afiliado, consultas

informativas, operaciones transaccionales simples, manejo de documentación y tratamiento de errores. Cada escenario describe un tipo de interacción relevante dentro del funcionamiento esperado del agente.

A partir de estos escenarios se elaboraron casos de prueba específicos, los cuales establecieron:

- las entradas que se utilizarían (mensajes del usuario o datos requeridos por el flujo),
- el comportamiento esperado del prototipo según lo definido en el diseño,
- los criterios de aceptación, orientados a verificar la coherencia conversacional, la invocación correcta de las operaciones implementadas y la consistencia de los datos recuperados.

Las pruebas se ejecutaron manualmente sobre la interfaz del chat, reproduciendo la interacción típica de un usuario final. Esta modalidad permitió observar de manera directa la construcción del diálogo, la recuperación de información y la ejecución de los distintos procesos.

Escenarios de prueba

Los escenarios de prueba representan situaciones típicas dentro del alcance del prototipo y permiten organizar los casos de validación de acuerdo con los procesos conversacionales definidos.

A continuación se presentan los seis escenarios utilizados en la validación funcional.

Escenario 1 — Identificación del afiliado

Este escenario evalúa el proceso de identificación del afiliado, el cual fue ajustado durante la fase de integración a partir de observaciones realizadas por el tribunal académico. En la versión inicial del prototipo, la identificación se realizaba únicamente mediante la combinación de DNI y número de afiliado. Sin embargo, se señaló la necesidad de

incorporar un mecanismo adicional de verificación para reforzar la seguridad del acceso a la información personal.

Como resultado, se incorporó un doble factor de validación, consistente en una confirmación adicional que debe proporcionar el usuario una vez completada la identificación básica. Este ajuste se integró dentro de la fase de implementación sin modificar el alcance funcional ni reabrir fases previas del proceso metodológico.

El escenario 1 permite validar este mecanismo actualizado, comprobando:

- que el sistema solicita el segundo factor tras recibir DNI y número de afiliado,
- que la afiliación solo se confirma si ambos factores son válidos,
- que el acceso a consultas y trámites posteriores se habilita únicamente después de completar el doble factor,
- que ante un dato incorrecto, el agente indica el error y permite corrección sin perder continuidad.

Resultado esperado:

El prototipo debe completar la identificación únicamente cuando ambos factores hayan sido verificados correctamente y, en caso contrario, denegar el acceso o solicitar la corrección correspondiente.

Escenario 2 — Solicitud de reintegro

Este escenario evalúa un flujo transaccional completo, incluyendo:

- inicio del trámite de reintegro,
- registro de ítems asociados,
- solicitud y carga de documentación requerida,
- Confirmación del trámite finalizado.

Resultado esperado:

El sistema debe crear un reintegro válido, vincular los ítems y registrar los documentos asociados conforme a los datos ficticios cargados.

Escenario 3 — Consulta de cobertura

Este escenario examina la capacidad del agente para responder consultas informativas relacionadas con prácticas médicas, incluyendo:

- reconocimiento de la práctica ingresada,
- verificación de su cobertura, copago o restricciones,
- recuperación de condiciones desde la base de datos simulada.

Resultado esperado:

El prototipo debe brindar una respuesta clara, precisa y coherente con los datos almacenados.

Escenario 4 — Consulta de topes y consumos

Este escenario permite validar:

- la recuperación de topes vigentes,
- el cálculo del saldo disponible según consumos registrados,
- la presentación clara de la información al usuario.

Resultado esperado:

El agente debe mostrar los topes correspondientes y el consumo actual, calculando de forma correcta el saldo disponible para el afiliado.

Escenario 5 — Registro de comunicación

Este escenario evalúa la creación de comunicaciones o reclamos por parte del afiliado, incluyendo:

- ayuda en la redacción de la comunicación
- registro de la comunicación,
- almacenamiento en el sistema,

Resultado esperado:

El prototipo debe generar correctamente la comunicación e informar al usuario sobre su registro.

Escenario 6 — Manejo de errores y datos incompletos

Este escenario evalúa la robustez del prototipo frente a entradas inválidas o información incompleta, tales como:

- datos mal escritos,
- formatos incorrectos,
- ausencia de información obligatoria,
- reintentos fallidos de identificación.

Resultado esperado:

El agente debe detectar el error, solicitar correcciones y recuperar el flujo sin perder el contexto de la conversación.

Distribución de casos de prueba

La validación funcional incluyó 24 casos de prueba, distribuidos entre los escenarios anteriores de la siguiente manera:

Escenario	Casos asignados	Objetivo validado
Identificación del afiliado	1–5	Autenticación y verificación de identidad
Solicitud de reintegro	6–10	Registro del trámite, ítems y documentación
Consulta de cobertura	11–13	Recuperación de condiciones de cobertura
Consulta de topes y consumos	14–16	Cálculo de consumos y saldo disponible
Registro de comunicación	17–18	Alta de comunicaciones
Manejo de errores	19–24	Detección y corrección de entradas inválidas

Esta distribución permitió cubrir tanto los flujos principales como los alternativos definidos en el alcance funcional del prototipo.

Resultados de la validación funcional

La ejecución de los 24 casos de prueba permitió evaluar el comportamiento del prototipo en los distintos procesos incluidos dentro de su alcance funcional. De manera general, el sistema mostró estabilidad, coherencia en el intercambio conversacional y correcta articulación entre frontend, backend y base de datos simulada.

Respecto del proceso de identificación del afiliado, se validó el mecanismo de doble factor incorporado durante la fase de desarrollo. En todos los casos evaluados, el prototipo solicitó dicha verificación adicional tras el ingreso de DNI y número de afiliado, habilitando el acceso únicamente cuando ambos factores coincidieron con los registros disponibles. En situaciones de datos incompletos o erróneos, el agente indicó la corrección necesaria y mantuvo el contexto del flujo.

En los procesos transaccionales, como la creación de solicitudes de reintegro o el registro de comunicaciones, el prototipo completó de forma correcta la secuencia de pasos, registró la información de acuerdo con el modelo de datos simulado y devolvió confirmaciones claras al usuario.

Las consultas informativas —cobertura, topes y consumos— devolvieron valores coherentes con los datos ficticios almacenados en la base de datos, cumpliendo los criterios de precisión establecidos para la validación.

En cuanto al manejo de errores, el agente identificó adecuadamente entradas inválidas, formatos incorrectos o información insuficiente. En la mayoría de los casos, el sistema permitió recuperar el flujo sin inconsistencias. Las fallas observadas fueron puntuales: una interpretación incorrecta de una práctica médica escrita con errores tipográficos significativos y un reinicio del proceso de identificación debido a datos incompletos. Ninguna de estas situaciones requirió modificaciones estructurales.

Síntesis de resultados

La ejecución de los 24 casos de prueba permitió obtener una visión integral del comportamiento del prototipo en los distintos procesos incluidos en su alcance funcional. En general, el sistema mostró estabilidad, coherencia conversacional e integración correcta entre sus componentes.

Resultados globales

Del total de 24 casos:

- 22 casos fueron exitosos, cumpliendo los criterios de aceptación.
- 2 casos presentaron fallas menores, que no afectaron la estructura del prototipo ni requirieron redefiniciones metodológicas.

Ambas fallas se consideran esperables dentro del desarrollo de un prototipo, y su impacto fue acotado.

Síntesis por escenario

Escenario 1 — Identificación del afiliado

El mecanismo de doble factor funcionó correctamente.

No se registraron fallas en este escenario.

Escenario 2 — Solicitud de reintegro

El prototipo completó el flujo de reintegro en casi todos los casos evaluados.

La única falla registrada se produjo cuando el usuario ingresó una práctica médica con un nombre severamente erróneo, lo que impidió que el sistema identificara el código correspondiente y completara el ítem del reintegro.

La falla no comprometió el funcionamiento general del flujo, pero señala un límite razonable para un prototipo cuya base contiene datos ficticios.

Escenario 3 — Consulta de cobertura

Las respuestas coincidieron con los datos definidos en la base simulada.

Sin fallas en este escenario.

Escenario 4 — Consulta de topes y consumos

El cálculo del saldo disponible se realizó correctamente en todos los casos.

Sin fallas en este escenario.

Escenario 5 — Registro de comunicación

El prototipo registró y recuperó comunicaciones sin inconsistencias.
Sin fallas en este escenario.

Escenario 6 — Manejo de errores

El sistema detectó la mayoría de las entradas inválidas y orientó al usuario hacia su corrección.

La única falla observada se presentó cuando el texto de entrada no incluía el año explícitamente, lo que llevó al agente a inferir y asignar un año incorrecto.

Tabla de resultados por escenario

Escenario	Casos	Resultados
Identificación del afiliado	1–5	5 exitosos / 0 fallas
Solicitud de reintegro	6–10	4 exitosos / 1 falla menor
Consulta de cobertura	11–13	3 exitosos / 0 fallas
Consulta de topes y consumos	14–16	3 exitosos / 0 fallas
Registro de comunicación	17–18	2 exitosos / 0 fallas
Manejo de errores	19–24	5 exitosos / 1 falla menor

Observaciones finales

Los resultados obtenidos confirman que el prototipo es funcional y consistente con los requerimientos definidos. La interacción conversacional, la articulación entre backend y base de datos y la ejecución de procesos se comportaron de manera estable.

Las dos fallas detectadas —una por identificación incorrecta de una práctica médica y otra por imposibilidad de recuperar un flujo tras un error consecutivo— se consideran propias del contexto de prototipo y no representan problemas arquitectónicos ni de diseño.

ANEXO E: VALIDACIÓN EXPERTA

Contexto y enfoque de la validación experta

La validación experta se realizó sobre un agente conversacional desarrollado como prototipo académico, cuyo objetivo es explorar la viabilidad de utilizar interfaces conversacionales para la atención administrativa de afiliados de obras sociales. El prototipo se diseñó como una prueba de concepto, orientada a evaluar decisiones de modelado de procesos, diseño conversacional y arquitectura técnica, más que a su despliegue en un entorno productivo.

La validación tuvo como finalidad analizar si el agente propuesto representa de manera adecuada los trámites administrativos habituales del dominio, si el flujo de interacción conversacional resulta comprensible y consistente para el afiliado, y si el enfoque técnico adoptado es compatible con los sistemas de información utilizados por las obras sociales. En este sentido, la validación no se orientó a medir desempeño cuantitativo ni eficiencia operativa, sino a evaluar la corrección conceptual y la adecuación del diseño desde la perspectiva de expertos del dominio.

El proceso de validación se llevó a cabo en el ámbito de una empresa de software especializada en soluciones para obras sociales (Tekhne S.A.), lo que permitió contrastar el prototipo con prácticas reales del sector. La evaluación se realizó mediante una demostración guiada del agente, acompañada de instancias de intercambio y discusión con profesionales con experiencia directa en el diseño, implementación y operación de sistemas administrativos para afiliados.

El alcance de la validación se delimitó explícitamente al prototipo presentado. Si bien el diseño del agente contempla criterios alineados con la normativa vigente en materia de protección de datos y tratamiento de información sensible —aspecto que fue considerado durante la evaluación con los expertos—, quedaron fuera del análisis las implementaciones formales propias de un entorno productivo, tales como mecanismos de auditoría, monitoreo continuo, controles operativos avanzados o certificaciones regulatorias. Estas dimensiones, si bien relevantes para un despliegue real, no afectan el objetivo de esta validación, centrado en evaluar la viabilidad conceptual, funcional y técnica de la propuesta dentro del marco de este trabajo académico.

Expertos participantes

La validación experta se realizó con la participación de profesionales con experiencia directa en el análisis, implementación y operación de sistemas informáticos utilizados por obras sociales. La selección de los participantes buscó cubrir dos perspectivas centrales

para este prototipo: (i) la visión funcional y operativa de los trámites administrativos, y (ii) la visión técnica asociada a la integración con sistemas core y a la viabilidad del enfoque.

Participaron los siguientes perfiles:

- 1 Jefe de módulo de prestaciones, con conocimiento operativo y de gestión sobre los procesos administrativos del área (coberturas, consumos, reintegros y reglas del dominio).
- 2 Analistas funcionales, con experiencia en relevamiento, especificación y validación de procesos en sistemas de gestión para obras sociales, aportando criterio sobre completitud, secuencia lógica de pasos y adecuación del flujo conversacional.
- 1 Desarrollador de sistemas core, con experiencia en implementación e integración sobre plataformas existentes, aportando evaluación sobre factibilidad técnica, separación de responsabilidades y posibles requerimientos de evolución hacia un entorno real.

La combinación de estos perfiles permitió evaluar el prototipo desde dimensiones complementarias: correctitud y representatividad de los procesos modelados, claridad del flujo conversacional y viabilidad técnica de integración con sistemas existentes. No se incluyeron perfiles clínicos, dado que el agente no aborda decisiones médicas ni realiza indicaciones de salud; asimismo, los aspectos de cumplimiento y operación productiva formal se consideran fuera del alcance del prototipo, tal como se definió en la sección de contexto.

Objeto y criterios de la validación experta

El objeto de la validación experta fue el agente conversacional desarrollado como prototipo, junto con los procesos administrativos modelados y su implementación como flujos conversacionales para la atención de afiliados de obras sociales.

La validación se orientó a analizar si el agente propuesto cumple adecuadamente con los objetivos definidos para este trabajo, evaluando su comportamiento desde una perspectiva funcional, conversacional y técnica. En particular, se consideraron los siguientes criterios de evaluación:

- Representatividad de los procesos: se analizó si los trámites modelados reflejan de manera correcta y completa los procesos administrativos reales de una obra social, sin introducir simplificaciones que distorsionen su funcionamiento.

- Coherencia del flujo conversacional: se evaluó si la secuencia de interacción propuesta resulta lógica, comprensible y consistente para un afiliado, manteniendo el contexto del trámite a lo largo de la conversación.
- Adecuación de la información entregada: se analizó si el nivel de información brindado por el agente es suficiente para orientar al afiliado, evitando tanto la omisión de datos relevantes como la exposición innecesaria de información sensible.
- Tratamiento de datos personales y sensibles: se evaluó si el agente aplica criterios prudentes de identificación y acceso a la información, alineados con prácticas habituales del dominio y con consideraciones normativas.
- Viabilidad técnica del enfoque: se analizó si la arquitectura y el modelo de integración propuestos resultan compatibles con sistemas de gestión existentes en obras sociales.

Quedaron explícitamente fuera del objeto de validación la resolución administrativa final de los trámites, la toma de decisiones clínicas, así como la operación del agente en un entorno productivo real, incluyendo aspectos de desempeño, escalabilidad, auditoría y monitoreo continuo. Esta delimitación permite centrar la validación en la corrección conceptual y la viabilidad del diseño, en coherencia con el carácter académico del prototipo.

Metodología de la validación experta

La validación experta se realizó mediante un enfoque cualitativo, basado en juicio experto, adecuado para la evaluación de prototipos académicos orientados a explorar decisiones de diseño y viabilidad conceptual. La metodología adoptada se apoyó en la observación directa, la demostración guiada del sistema y el intercambio estructurado con los expertos participantes.

La validación se llevó a cabo a través de una sesión de demostración en la cual el autor de este trabajo ejecutó manualmente los distintos procesos implementados por el agente conversacional, explicando en cada caso el objetivo del trámite, el flujo propuesto y las decisiones de diseño relevantes. Durante esta instancia, los expertos analizaron el comportamiento del agente y formularon observaciones en función de su experiencia en el dominio.

Los procesos evaluados durante la validación fueron:

- Identificación del afiliado.
- Solicitud de afiliación.
- Solicitud de reintegro.

- Consultas de prestaciones (cobertura y topes).
- Consulta de historial de consumos.
- Registro y consulta de reclamos.

Para guiar la evaluación y asegurar una revisión sistemática, se plantearon preguntas orientadoras comunes a todos los procesos:

- ¿El proceso representa correctamente el trámite administrativo en el contexto de una obra social?
- ¿La secuencia de pasos propuesta resulta lógica y completa?
- ¿El inicio, desarrollo y cierre del trámite se encuentran claramente delimitados?
- ¿Existen pasos innecesarios o faltantes en el flujo presentado?

Además de los procesos, se analizaron de manera transversal el flujo conversacional, el nivel de información entregada, el tratamiento de datos personales y sensibles y la viabilidad técnica del enfoque, de acuerdo con los criterios definidos en la sección anterior.

La metodología adoptada permitió recoger observaciones cualitativas relevantes sobre el comportamiento del prototipo, identificar oportunidades de mejora y validar las decisiones de diseño, sin recurrir a métricas cuantitativas, las cuales no resultan pertinentes para el alcance exploratorio de este trabajo.

Resultados de la validación experta

Resultados sobre los procesos administrativos

En relación con los procesos implementados por el agente conversacional, los expertos coincidieron en que los trámites modelados representan de manera adecuada los procesos administrativos habituales de una obra social. La evaluación permitió verificar que los flujos definidos mantienen correspondencia con la práctica real del dominio, respetando la secuencia lógica de pasos y los puntos de inicio y cierre propios de cada trámite.

Desde el punto de vista funcional, se destacó positivamente la correcta delimitación del alcance del agente, el cual actúa como interfaz de atención y gestión inicial sin asumir responsabilidades que corresponden a los sistemas administrativos centrales. Esta separación fue considerada adecuada y coherente con las prácticas habituales de informatización del sector.

Durante la demostración guiada, no se identificaron omisiones relevantes en los procesos evaluados ni pasos que distorsionaran el funcionamiento real de los trámites. Como observación puntual de mejora, los expertos señalaron que, en el proceso de solicitud de reintegro, podría mostrarse explícitamente el ítem agregado al momento de su carga, ya que el agente confirma actualmente la acción mediante un mensaje genérico. Esta observación fue considerada una mejora de usabilidad, sin impacto en la corrección funcional del proceso.

Resultados sobre el flujo conversacional

Respecto del flujo conversacional, los expertos evaluaron que la interacción propuesta resulta clara, consistente y fácil de seguir para un afiliado. Se observó que el agente mantiene correctamente el contexto del trámite en curso y evita confusiones incluso cuando el usuario encadena distintas consultas o procesos dentro de una misma interacción.

Se valoró positivamente el uso de confirmaciones previas a acciones relevantes, ya que estas contribuyen a reducir errores y refuerzan la comprensión del estado del trámite por parte del usuario. Asimismo, los mensajes generados por el agente fueron considerados adecuados para guiar al afiliado, indicando de manera explícita qué información se espera en cada paso.

Como posible línea de mejora futura, los expertos sugirieron la incorporación de listas numeradas para la selección de opciones, permitiendo que el afiliado responda mediante un número. Esta mejora apunta a simplificar la interacción y reducir ambigüedades propias del lenguaje natural, sin afectar el diseño conversacional general validado.

Resultados sobre la información entregada

En cuanto a la información entregada por el agente, los expertos consideraron que el nivel de detalle proporcionado es suficiente y adecuado al canal conversacional, permitiendo orientar al afiliado sin sobrecargarlo con información innecesaria. Se destacó que el agente ajusta el contenido de las respuestas según el tipo de trámite, priorizando información resumida y contextualizada.

Los expertos señalaron que, en sistemas productivos reales, la información presentada al usuario suele ser más extensa; no obstante, coincidieron en que dicha diferencia se explica por la presencia de mecanismos de autenticación más robustos en otros canales, como aplicaciones móviles o portales web. En este contexto, el enfoque adoptado por el prototipo fue considerado apropiado para un canal conversacional, especialmente en relación con consultas sobre consumos, topes y prestaciones.

Como sugerencia de mejora, se indicó que podría reforzarse al inicio de la interacción la comunicación de los procesos que el agente es capaz de cubrir, con el objetivo de establecer expectativas claras en el afiliado desde el comienzo de la conversación.

Resultados sobre el tratamiento de datos personales y sensibles

En relación con el tratamiento de datos personales y sensibles, los expertos evaluaron como adecuada la estrategia de identificación previa implementada por el agente para acceder a información sensible. Se observó que el sistema expone únicamente la información necesaria para cada trámite, evitando la visualización de datos personales, de salud o económicos que no resultan pertinentes al contexto de la consulta.

Se valoró positivamente que el agente aplique criterios prudentes en el acceso a la información y mantenga una separación clara entre respuestas informativas y aquellas situaciones que requieren una gestión administrativa posterior. Este enfoque fue considerado consistente con prácticas habituales del dominio y con consideraciones normativas aplicables al tratamiento de datos sensibles.

Como aspecto a considerar a futuro, los expertos señalaron el riesgo operativo asociado a la falta de actualización de los datos de contacto en las bases de afiliados. En particular, se mencionó que el correo electrónico del afiliado no siempre se encuentra actualizado en los sistemas core. Como posible línea de evolución, se propuso la integración con bases de usuarios de aplicaciones móviles o portales web, donde los datos de contacto suelen mantenerse actualizados y vinculados al afiliado.

Resultados sobre la viabilidad técnica

Desde el punto de vista técnico, los expertos coincidieron en que el enfoque propuesto resulta viable y compatible con los sistemas de gestión utilizados por obras sociales. Se destacó positivamente la separación entre la lógica conversacional y la lógica de negocio, así como la definición clara de responsabilidades de las herramientas utilizadas por el agente.

Los expertos consideraron que la arquitectura del prototipo permite una evolución progresiva hacia un entorno productivo, sin requerir cambios estructurales significativos. Como sugerencias de mejora futura, se mencionó la conveniencia de incorporar un panel de administración que permita parametrizar el comportamiento del agente sin modificaciones de código, así como mecanismos adicionales de control, tales como filtros de mensajes, limitaciones de interacción y estrategias de fallback ante fallos.

Estas observaciones fueron consideradas mejoras incrementales, propias de una etapa de maduración del sistema, y no afectan la validez del enfoque técnico evaluado.

Conclusión de la validación experta

La validación experta realizada permitió evaluar el prototipo de agente conversacional desde una perspectiva funcional, conversacional y técnica, en relación con las prácticas habituales del dominio de obras sociales. A partir de la revisión guiada de los procesos implementados y del intercambio con los expertos participantes, se constató que los flujos propuestos representan de manera adecuada los trámites administrativos considerados y que el comportamiento del agente resulta coherente con los circuitos institucionales vigentes.

Asimismo, la evaluación evidenció que el diseño conversacional facilita la interacción con el afiliado, manteniendo el contexto de los trámites y proporcionando información suficiente para orientar al usuario sin exponer datos innecesarios o sensibles. Desde el punto de vista técnico, el enfoque adoptado fue considerado viable y compatible con sistemas de gestión existentes, destacándose la correcta separación entre la lógica conversacional y la lógica de negocio.

Las observaciones y sugerencias realizadas por los expertos se orientaron principalmente a mejoras de usabilidad y a posibles líneas de evolución futura del sistema, sin cuestionar las decisiones de diseño adoptadas ni requerir modificaciones estructurales del prototipo. En consecuencia, la validación experta respalda la adecuación del agente conversacional dentro del alcance definido para este trabajo académico.