

Definición del Proyecto: Modernización del Chatbot de Cootradecun

1. Introducción y Justificación

1.1. Problema Actual

La cooperativa Cootradecun opera un chatbot a través de Manychat en WhatsApp, enfocado en consultas y generación de certificados. Se ha identificado que el sistema actual presenta limitaciones significativas que restringen su adopción y eficiencia:

- Baja Usabilidad y Adopción:** El flujo de conversación es excesivamente condicional ("con muchas ramas de decisiones"), lo que resulta confuso para los asociados, especialmente para personas mayores, llevando a altas tasas de abandono en el primer menú.
- Información Estática:** La información del chatbot es fija y no permite la actualización ágil de nuevos procesos, normativas o documentos de crédito, limitando su utilidad.
- Alcance Limitado:** La interacción se restringe a un menú predefinido, sin capacidad para una conversación natural y fluida.

1.2. Solución Propuesta (Objetivo del Proyecto)

Crear un nuevo chatbot de **conversación fluida** e inteligente, capaz de entender y responder a consultas complejas, con el objetivo de aumentar la satisfacción del asociado, la eficiencia operativa y la adopción del canal digital.

El nuevo sistema se basará en **agentes LLM** (Large Language Models) y un diseño de arquitectura moderna.

2. Alcance y Requisitos

2.1. Plataforma Tecnológica

Componente Clave	Propósito
Agentes LLM	Habilitar la conversación fluida y el procesamiento del lenguaje natural.

RAG System	(Retrieval-Augmented Generation) Integrar y mantener actualizada la base de conocimiento.
Middleware	Langgraph, Langchain, FastAPI.
Canal	Meta Business (WhatsApp).
Hosting	GCP (Google Cloud Platform): Alojamiento escalable en la nube para la infraestructura del chatbot.

2.2. Requerimientos Funcionales

A. Generación de Certificados (Conexión a WebServices)

El chatbot debe permitir la generación y envío directo de los siguientes documentos al asociado, previa conexión a **WebServices externos** (a ser construidos por un tercero):

- Estados de cuenta (Feature Existente)
- Certificado de paz y salvo (Feature Existente)
- Certificado de cartera (Feature Existente)
- Certificado tributario (Feature Existente)
- Certificado de calidad de asociado (**Mejora:** *Se requiere definir claramente los criterios y la fuente de datos para esta certificación, ya que se indica que es "no muy clara".*)
- Certificado de ahorros (Feature Existente)
- Certificados de aportes (Feature Existente)

B. Autenticación (Revisada)

Los afiliados que requieran acceder a información personal (certificados, estados de cuenta, reportes tributarios) deberán autenticarse obligatoriamente mediante un **Código de Un Solo Uso (OTP)**.

El sistema debe permitir al asociado elegir el canal de recepción del OTP:

- **WhatsApp / SMS:** Envío del código directamente a través del **canal de WhatsApp** a la línea registrada. El envío por mensaje de texto (SMS) se considera una **alternativa de respaldo** en caso de fallos de entrega o si el asociado lo selecciona explícitamente.
- **Correo Electrónico:** Envío del código al correo electrónico registrado del asociado.

C. Consultas Generales (Q&A)

El chatbot funcionará como un motor de Q&A, con acceso a información de los siguientes departamentos:

- Atención al asociado
- Cartera
- Convenios
- Nóminas
- Vivienda
- Créditos

D. Funcionalidades de Enlace Web

El chatbot debe ser capaz de mostrar información y/o guiar al usuario a las siguientes funcionalidades de autogestión de la cooperativa, que actualmente se encuentran en la web:

- Actualización de datos
- Vinculación a la entidad
- Simulación y solicitud de créditos
- Realización de pagos de obligaciones
- Solicitud de auxilios

E. Aplicación Administrativa

Se requiere una aplicación con las siguientes funcionalidades para la gestión interna del sistema:

- **Módulo de Reportes y Monitoreo:**
 - Cantidad de usuarios activos.
 - Cantidad de conversaciones por fecha.
 - Temas más consultados (análisis de *intents*).
 - Módulo de uso de tokens (para monitoreo de costos LLM).
- **Módulo RAG:**
 - Interfaz para el manejo y actualización de archivos de la base de conocimiento (documentos, políticas, etc.) por parte del usuario administrador.
- **Módulo de Usuarios y Seguridad:**
 - Gestión de Usuarios, Roles y permisos de la aplicación administrativa.

2.3. Requerimientos No Funcionales

- **Canal:** El medio de comunicación principal del chatbot es **WhatsApp**.
- **Disponibilidad:** Debe garantizarse una disponibilidad del **99.9%** (operación 24/7).
- **Escalabilidad:** Debe permitir la comunicación y gestión de un alto volumen de usuarios concurrentes.

3. Riesgos del Proyecto y Mitigación Sugerida

Riesgo del Proyecto	Impacto Sugerido	Mitigación Propuesta
Data muy grande	Alto ▾	Implementar estrategias de <i>chunking</i> y optimización de índices RAG. Definir un alcance inicial del corpus de conocimiento y escalarlo progresivamente.
Capacidad de respuesta	Alto ▾	Realizar pruebas de carga (Stress Testing) antes del lanzamiento. Utilizar servicios <i>serverless</i> o escalables en GCP (ej. Cloud Run) para manejar picos de demanda.
Imprecisión en las respuestas	Alto ▾	Implementar un Módulo de "Human Handover" (traslado a agente humano) para respuestas con baja confianza. Incluir un <i>feedback loop</i> de usuario y monitoreo constante para refinar el modelo RAG y prompts.
Hueco de seguridad en la data	Crítico ▾	Implementar cifrado de datos en reposo y en tránsito. Asegurar que la autenticación (código SMS) cumpla con estándares de seguridad. Realizar auditorías de seguridad (Penetration Testing) al sistema de autenticación y WebServices.
Subestimar el tiempo del proyecto	Medio ▾	Desglosar el proyecto en fases (MVP con Q&A + 1 Certificado). Crear un cronograma detallado con contingencias y asegurar la disponibilidad del equipo

		técnico.
Acceso a la data con los webservices	Alto ▾	Dependencia Crítica: Formalizar la fecha de entrega de la API (WebServices) por parte del tercero. Establecer un contrato de nivel de servicio (SLA) para el acceso a dichos servicios. Definir un plan de contingencia si la entrega se retrasa.
Webservices - Precios no definidos	Alto ▾	Dependencia Financiera: Iniciar inmediatamente el proceso de cotización y negociación con el tercero para obtener los costos y plazos definitivos de la construcción de los WebServices.
Dependencia de la Información Interna (Datos Desactualizados o Faltantes)	Alto ▾	Establecer un Plan de Gobernanza de Contenido RAG . Designar responsables (stakeholders) por departamento para la creación y actualización periódica de la documentación. Incluir métricas de <i>frescura</i> de la información en el Módulo Administrativo y procesos de auditoría de contenido.
Costo Inesperado de LLM/Uso de Tokens	Alto ▾	Implementar límites de token y modelos de costo de las API en el módulo de monitoreo. Optimizar prompts para reducir el uso de tokens. Evaluar el uso de modelos más pequeños y eficientes (frente a modelos de vanguardia) para tareas específicas.

Integración Compleja de WebServices	Alto ▾	Designar un punto focal técnico para la integración de WebServices y asegurar que se definan mocks (simulaciones) y contratos de datos claros al inicio. Realizar pruebas de integración tempranas y continuas (CI/CD).
Baja Adopción del Nuevo Canal (por Hábitos Antiguos)	Medio ▾	Diseñar una campaña de marketing interno enfocada en los beneficios de la conversación fluida (fácil de usar). Incluir sesiones de capacitación enfocadas en la población mayor. Mantener el canal actual (Manychat/WhatsApp) por un periodo de coexistencia y transición.