

A

AUNNA
innovation technologies

ANÁLISIS FUNCIONAL

Presentado a:



Murcia, a 06 de febrero de 2024

© AUNNA IT – Reservados todos los derechos.



www.aunnait.es



Título documento	Plan de proyecto		
Destinado a:	Diputación de Alicante		
Preparado por:	AUNNA IT	Fecha de versión	06/02/2024
Revisado por:		Fecha de revisión	

Versión	Fecha versión	Autor	Descripción
1.0	06/02/2024	AUNNA IT	Versión inicial del documento.
2.0	23/02/2024	AUNNA IT	Nuevo apartado 2.4.7. Gestión de conceptos y propiedades
3.0	13/03/2024	AUNNA IT	Modificaciones realizadas tras la revisión por parte de Suma.

Contenido

1. INTRODUCCIÓN	4
2. FUNCIONALIDAD DEL SISTEMA	4
2.1. Interfaz del chat	5
2.2. Diseño del flujo conversacional	6
2.3. Lógica de negocio.....	10
2.4. Inteligencia Artificial	11
2.4.1. Introducción	11
2.4.2. Alineación con los Requisitos Técnicos	11
2.4.3. Inferencia Semántica	11
2.4.4. Comparativa RASA – AIML.....	12
2.4.5. Mecanismos de aprendizaje.....	12
2.4.6. Terminología.....	14
2.4.7. Gestión de conceptos y propiedades	14
2.5. Intervención de un agente.....	16
2.6. Herramienta web de gestión	17
2.6.1. Login	18
2.6.2. Métricas.....	18
2.6.3. Clientes	19
2.6.4. Chatbots	20
2.6.5. Conversaciones.....	21
2.6.6. Temáticas e intenciones	23
2.6.7. Entrenamiento supervisado	26
2.6.8. Usuarios.....	28
2.6.9. Ajustes generales.....	28
3. CAPA DE INTEGRACIÓN	29
3.1. Autenticación	29
3.2. Clientes.....	30
3.3. Chatbot	32
3.4. Intenciones.....	34
3.5. Usuarios	37
3.6. Conversaciones	39
3.7. KPI	40
3.8. Agente.....	43



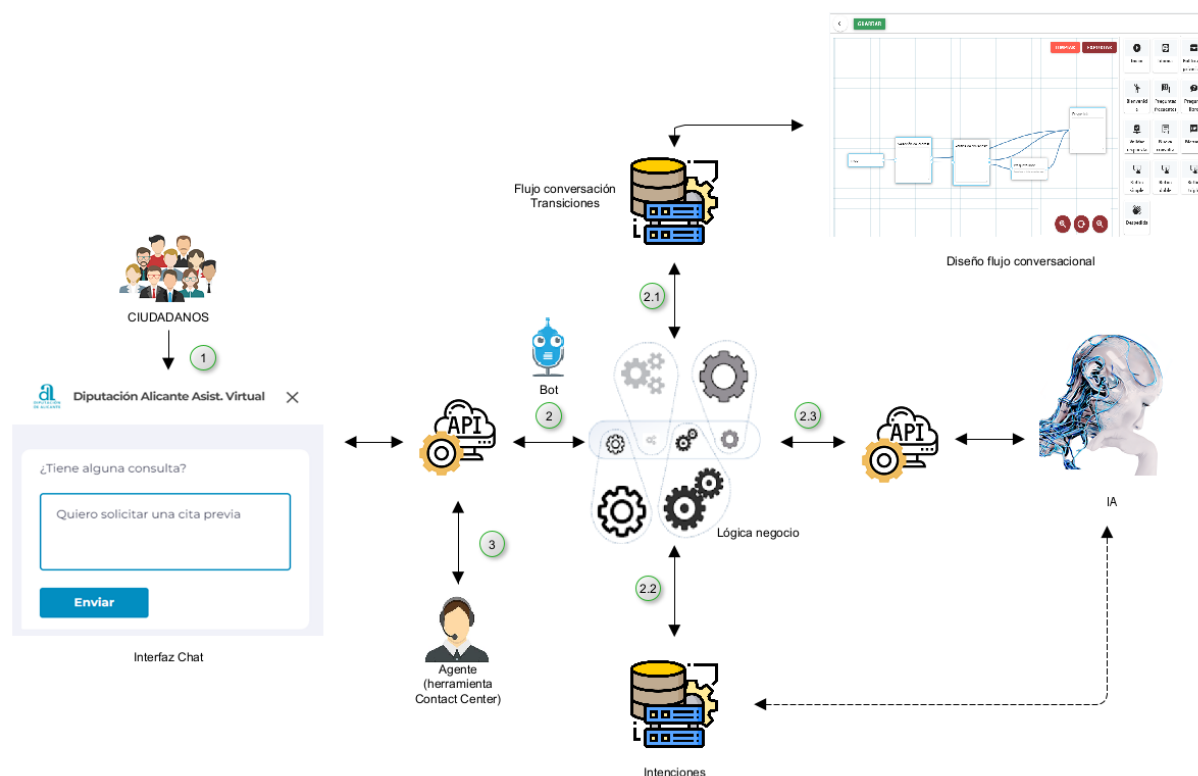
1. INTRODUCCIÓN

El presente documento constituye el análisis funcional para el desarrollo de un asistente conversacional de código abierto para la Diputación de Alicante, destinado a mejorar la interacción y atención al ciudadano. Este proyecto surge como respuesta a la necesidad de optimizar la prestación de servicios públicos, favoreciendo la accesibilidad, la eficiencia y la modernización de los procesos administrativos.

El Chatbot se concibe como una herramienta clave para ofrecer un canal de comunicación ágil y disponible las 24 horas del día, los 7 días a la semana, capaz de responder consultas, guiar en trámites y proporcionar información relevante de manera automatizada. Su implementación busca mejorar la experiencia del usuario, reducir los costos operativos y facilitar el acceso a los servicios públicos digitales.

Este análisis funcional proporciona una visión detallada de los requisitos y funcionalidades que se espera que el Chatbot cumpla, abarcando aspectos como la interfaz de usuario y las integraciones necesarias. Además, establece las bases para el desarrollo del sistema, asegurando su alineación con los objetivos estratégicos, cubriendo, por tanto, las necesidades específicas incluidas en el pliego para los municipios de Alicante, así como los organismos públicos del ámbito municipal.

2. FUNCIONALIDAD DEL SISTEMA



El desarrollo de este proyecto, en lo que respecta al asistente conversacional, involucra la integración de diversos componentes que quedan representados en el diagrama anterior.

En este contexto, se destacan seis elementos fundamentales que constituyen la estructura principal del sistema. Cada uno de estos elementos serán gestionados y administrados por medio de una herramienta web de gestión que se detalla en el apartado “Herramienta web de gestión”.

- (1) Interfaz del chat. Es la cara visible del Chatbot, utilizada por el ciudadano para interactuar con el asistente virtual o agente.
- Diseño de flujo conversacional. Permite diseñar la lógica conversación que será utilizada para guiar la conversación con el ciudadano.
- (2) Lógica de negocio. Es el nexo de unión de todos los componentes y el encargado de decidir cómo actuar ante cada solicitud del usuario.
 - (2.1) Base de datos de transiciones. Conocimiento para guiar la conversación, a partir del estado actual y una transición se decide el siguiente estado.
 - (2.2) Base de datos de intenciones. Las intenciones serán consultadas directamente sin pasar por el motor de IA cuando la solicitud del usuario proviene de los botones configurados para la FAQ.
 - (2.3) Integración con el motor de inteligencia artificial. La solicitud del usuario será enviada a la IA cuando se trate de una pregunta abierta.
- (3) Intervención de un agente a una conversación. En caso de ser necesario, un agente podrá intervenir en una conversación.

Cada uno de estos componentes desempeña un papel crucial en la funcionalidad global del sistema. A continuación, en los siguientes apartados, se detalla el funcionamiento de cada uno de estos componentes.

2.1. Interfaz del chat

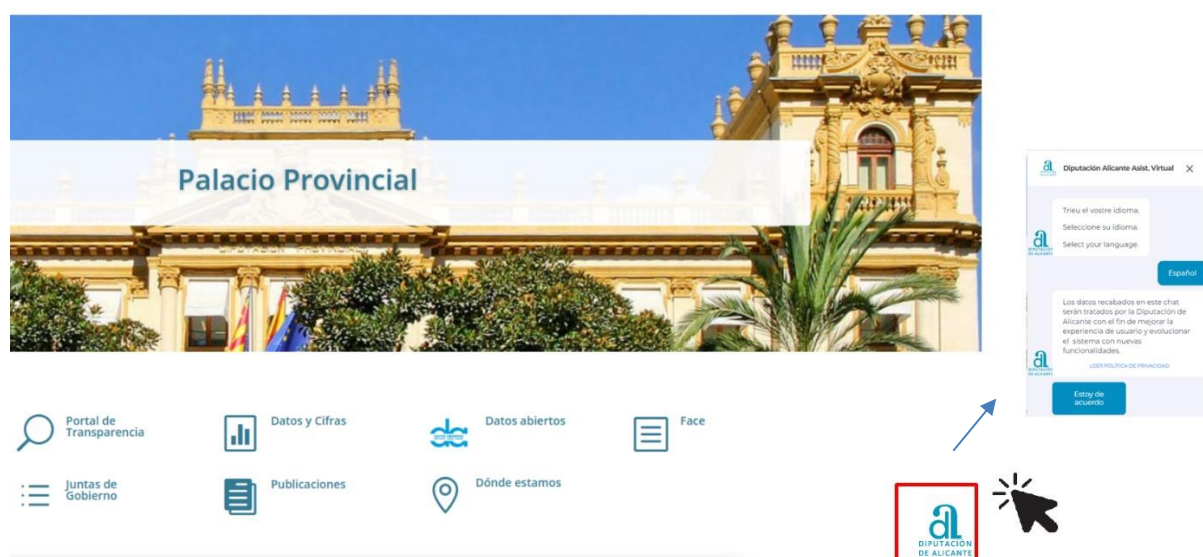
El asistente conversacional a desarrollar presenta una serie de requisitos funcionales y de usabilidad que garantizan una experiencia fluida e intuitiva para los usuarios.

En primer lugar, la interfaz del chat del asistente conversacional se diseñará de manera intuitiva y amigable, permitiendo su integración en cualquier página web de forma sencilla. Esta interfaz cumplirá con los estándares de usabilidad y accesibilidad definidos en el documento de requisitos, asegurando que todos los usuarios puedan interactuar de manera efectiva con el Chatbot.

Para integrar el chat en cualquier página web, se utilizará un método que permita la inclusión mediante un *iframe*. Esta flexibilidad asegura que el chat pueda ser implementado de manera sencilla y compatible con una amplia variedad de plataformas y sistemas de gestión de contenido. El uso de un *iframe* permite incrustar el chat como un elemento independiente en la página.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Chatbot Example</title>
  </head>
  <body>
    ...
    <!-- Contenedor para el chat -->
    <script> ... </script>
  </body>
</html>
```

El Chatbot se presentará inicialmente de forma minimizada, representado por un icono visual. Además, el chatbot estará disponible en tres idiomas: castellano, valenciano e inglés, dando la opción al usuario de seleccionar el idioma de su preferencia.



El Chatbot estará integrado con un motor de Inteligencia Artificial que utilizará técnicas de Procesamiento de Lenguaje Natural (PLN) para comprender y responder preguntas abiertas de manera coherente y contextual.

La interfaz del chat admitirá la personalización de determinados aspectos visuales, como:

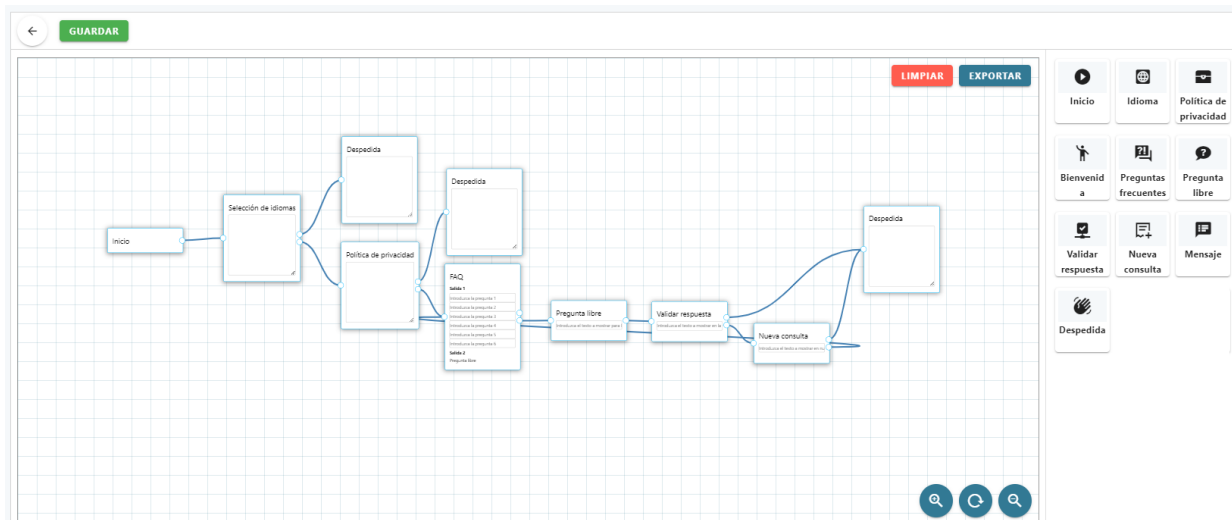
- Logo corporativo y el color de la interfaz.
- Título que aparecerá en la cabecera del chatbot.
- Ajustar el tamaño de la letra con el fin de proporcionar una mayor flexibilidad y comodidad durante la interacción, según las necesidades visuales o preferencias personales de cada usuario.
- Maximizar la interfaz del chat para obtener una visualización más amplia y cómoda. Esta función permite expandir la ventana del chat a pantalla completa, lo que facilita la lectura de los mensajes y la interacción con el chatbot. Al maximizar la interfaz, los usuarios pueden aprovechar al máximo el espacio disponible en la pantalla, lo que mejora la experiencia de usuario.

En cuanto a los requisitos de usabilidad, la interfaz del chatbot se basará en tecnología web y seguirá la metodología *Mobile First*, garantizando una experiencia óptima tanto en dispositivos móviles como en escritorio. Además, será compatible con los principales navegadores web y estará adaptada a diferentes resoluciones de pantalla y tamaños de dispositivo.

2.2. Diseño del flujo conversacional

El sistema permitirá definir el flujo conversacional de cada Chatbot de manera visual, a través de la configuración de un sistema de estados y transiciones. Esta funcionalidad permite una personalización completa del proceso de interacción con el usuario, lo que resulta fundamental para adaptarse a las necesidades específicas de cada caso de uso.

A través de esta configuración, se pueden diseñar flujos conversacionales que guíen al usuario desde el inicio de la conversación de manera intuitiva y eficiente.



Por medio del diagrama de flujo anterior podremos disponer de las siguientes operaciones:

- Inicio. Estado inicial desde el que arrancará la lógica conversacional. Este estado será único.
- Idioma. Se solicita al usuario la selección del idioma en el que quiere ser atendido. Previamente se debe indicar los idiomas que gestionará el Chatbot hasta un máximo de tres: castellano, valenciano e inglés. El aspecto visual será el siguiente:

- Política de privacidad. El usuario debe aceptar la política de privacidad para poder continuar con la conversación.

- Preguntas frecuentes. Configuración de los diferentes botones de acceso rápido a las preguntas frecuentes. Cada botón estará asociado a una intención. En el apartado “Temáticas e intenciones” se detalla cómo crear y gestionar cada una de las intenciones.

- Pregunta libre. El usuario podrá realizar libremente la pregunta que desee.
- Mensaje. Cuadro de texto para configurar un mensaje a trasladar al usuario tras una determinada acción.

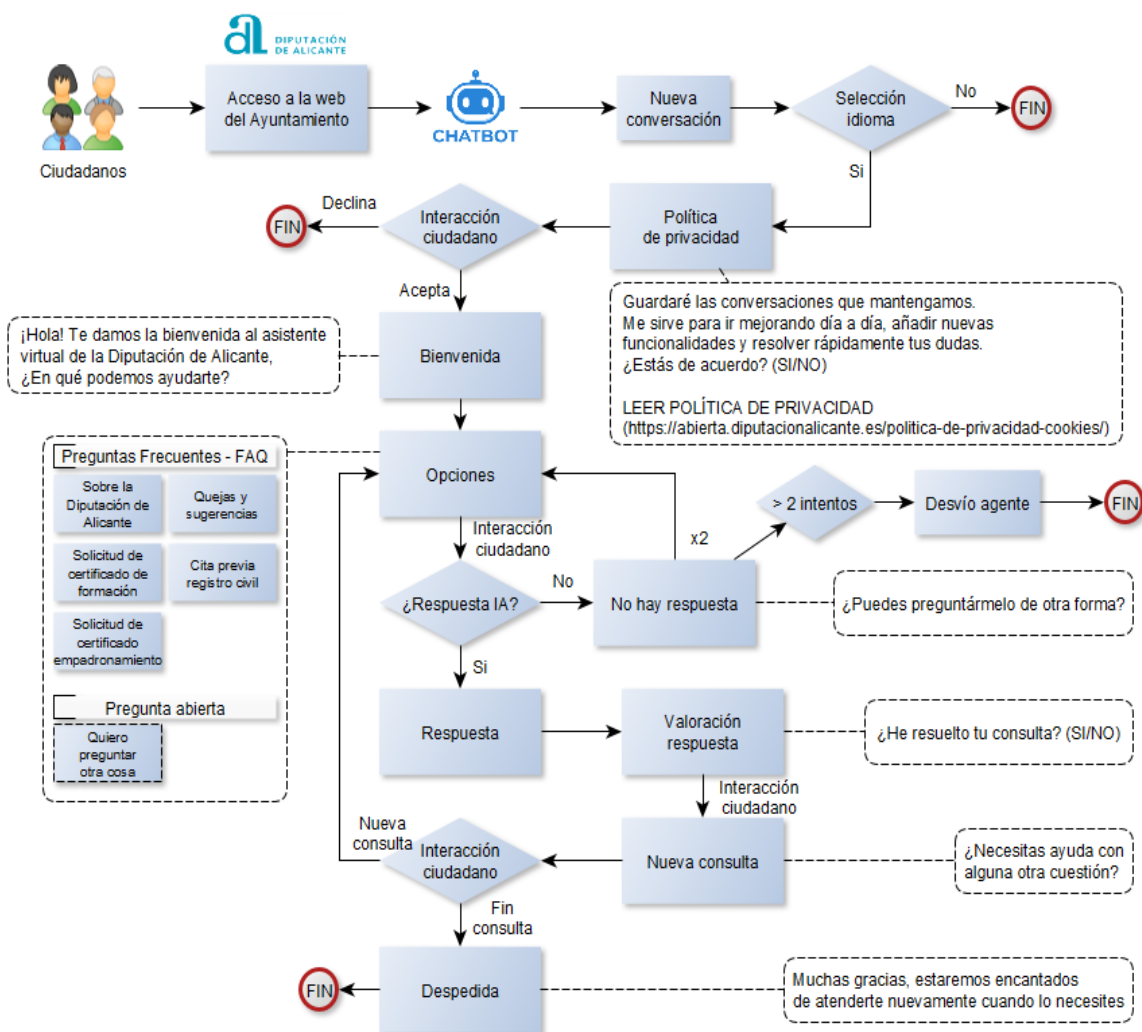
- Consulta resuelta. Se podrá solicitar al usuario que indique si la solicitud ha sido resuelta de manera satisfactoria.

- Otra gestión. Se da la opción al usuario para poder realizar otra gestión.

- Despedida. Estado final de la conversación, contendrá un cuadro de texto para configurar el mensaje de despedida.



Con el conjunto de funcionalidades descritas anteriormente, será posible generar flujos conversaciones que seguirán una lógica similar a la representada en el siguiente diagrama:



Esta lógica del flujo conversacional será exportada para incorporarla en cualquier página web a través del *iframe* correspondiente, tal y como se ha comentado en el apartado anterior.

Por otro lado, por medio de los roles y permisos definidos, se delimitarán las responsabilidades en relación a la creación, modificación y visualización de los flujos conversacionales. Mientras que el superadministrador y el administrador tienen el control completo sobre la creación y eliminación de

flujos, el rol de editor puede realizar adaptaciones específicas en los flujos de los clientes asignados. Por último, el rol de supervisor tiene acceso a la información general del Chatbot, incluyendo la visualización de su flujo conversacional.

Para garantizar la calidad del servicio, el Chatbot tendrá la posibilidad de solicitar el *feedback* correspondiente a los usuarios después de cada consulta, permitiendo validar si la respuesta fue satisfactoria y recopilar comentarios para mejorar las interacciones futuras. Además, se ofrecerá al usuario la posibilidad de realizar otra gestión tras cada solicitud.

2.3. Lógica de negocio

La lógica de negocio constituye el núcleo operativo del sistema de asistente conversacional, siendo la responsable de procesar las solicitudes de los usuarios y tomar decisiones en tiempo real. Su función principal es interpretar las interacciones del usuario y determinar la acción más adecuada en función de las necesidades y objetivos establecidos. Para lograr esto, la lógica de negocio se apoya en una serie de recursos y procesos clave que le permiten desempeñar su papel de manera eficiente.

La lógica de negocio interactúa principalmente con tres componentes:

- **Base de datos de transiciones.** Esta base de datos contiene información detallada sobre los posibles estados de la conversación y las transiciones entre ellos. Al recibir una solicitud de un usuario, la lógica de negocio consulta esta base de datos para determinar cuál es el siguiente estado apropiado en el flujo de la conversación. Esta capacidad garantiza que el sistema pueda guiar al usuario de manera coherente y contextualizada a lo largo de la interacción.
- **Base de datos de intenciones.** Además, la lógica de negocio cuenta con acceso a la base de datos de intenciones, especialmente cuando se trata de consultas provenientes de las preguntas frecuentes preconfiguradas mediante botones. Al recibir una solicitud de este tipo, la lógica de negocio consulta la base de datos de intenciones para obtener información relevante y precisa que pueda satisfacer la consulta del usuario de manera rápida y efectiva sin necesidad de interactuar con la IA.
- **IA.** Asimismo, la lógica de negocio tiene la capacidad de enviar consultas al motor de inteligencia artificial cuando la solicitud del usuario requiere un análisis más profundo o una respuesta basada en el procesamiento de lenguaje natural. El motor de IA utiliza técnicas avanzadas de procesamiento de lenguaje natural para comprender el contexto y el significado de las consultas del usuario.

La lógica de negocio constituye el eje central del sistema de asistente conversacional, actuando como el motor que impulsa la interacción entre el usuario y el sistema. Su capacidad para tomar decisiones y responder de manera efectiva a las solicitudes de los usuarios es fundamental para garantizar una experiencia fluida y satisfactoria. Para facilitar esta interacción, la lógica de negocio se comunica de manera bidireccional a través de APIs tanto con la interfaz del chat como con la inteligencia artificial (IA).

Por un lado, la comunicación vía API con la interfaz del chat permite que la lógica de negocio envíe y reciba mensajes del chat en tiempo real. Por otro lado, la comunicación vía API con la IA permite que la lógica de negocio envíe consultas y solicitudes de procesamiento de lenguaje natural al motor de IA.

Además, la lógica de negocio también desempeña un rol crucial en la detección y gestión de la intervención de un agente en una conversación. Cuando se detecta la necesidad de intervención de un

agente, la lógica de negocio actúa de manera proactiva para detener temporalmente al bot y permitir la participación del agente en la conversación. Este proceso se lleva a cabo de manera transparente para el usuario.

Una vez que el agente ha participado en la conversación y se ha resuelto la solicitud o consulta del usuario, éste deberá cerrar la conversación. Si esto no sucede, el sistema cerrará de manera automática la conversación cuando no se generan más mensajes por parte del usuario en un período de tiempo determinado. En ambos casos, la lógica de negocio toma las medidas necesarias para reactivar al bot y restablecer la interacción automatizada.

2.4. Inteligencia Artificial

2.4.1. Introducción

La tecnología de Inteligencia Artificial seleccionada para el proyecto es RASA. RASA es una plataforma de código abierto que proporciona las herramientas necesarias para construir Chatbots y asistentes virtuales robustos, escalables y personalizables. Su arquitectura, basada en tecnologías avanzadas de procesamiento de lenguaje natural (NLP) y aprendizaje automático, permite desarrollar soluciones de Chatbot que pueden comprender e interactuar con los usuarios de manera natural y efectiva.

2.4.2. Alineación con los Requisitos Técnicos

La elección de RASA como la tecnología subyacente para el desarrollo del Chatbot municipal se fundamenta en su capacidad para cumplir y, en muchos casos, superar los requisitos técnicos especificados en el pliego de la licitación.

- Código Abierto y Sin Coste de Licencias. RASA, siendo una plataforma de código abierto, se alinea perfectamente con los requisitos de utilizar tecnologías públicas y sin costes de licencia, permitiendo una implementación y personalización flexible y sin restricciones.
- Multilingüismo. La capacidad de RASA para soportar múltiples idiomas asegura que el Chatbot pueda comunicarse efectivamente en castellano, valenciano e inglés, cumpliendo con las necesidades lingüísticas del proyecto.
- Interoperabilidad. RASA ofrece APIs robustas y capacidades de integración, permitiendo que el chatbot se conecte e interactúe con diversos sistemas y bases de datos, asegurando una operatividad cohesiva con los sistemas existentes de la administración.
- Seguridad: La plataforma permite implementar soluciones seguras, garantizando la protección de los datos de los usuarios y cumpliendo con las normativas de seguridad y privacidad pertinentes.

2.4.3. Inferencia Semántica

RASA utiliza tecnologías de NLP para realizar inferencias semánticas, permitiendo que el chatbot comprenda el significado detrás de las consultas de los usuarios y proporcione respuestas relevantes y contextualmente apropiadas.

Hará uso también de la biblioteca Spacy, que es ampliamente reconocida por su capacidad para procesar y entender el lenguaje natural. Spacy permite al Chatbot realizar análisis lingüísticos profundos, como la tokenización, la lematización, y la identificación de entidades, mejorando la capacidad del Chatbot para entender y responder a una variedad de consultas de los usuarios.

2.4.4. Comparativa RASA – AIML

En el pliego se propone el uso de la tecnología AIML. A continuación, se realiza una comparativa la cual condujo, finalmente, a hacer uso de RASA.

AIML es un lenguaje basado en XML que es usado para crear patrones de conversación en Chatbots. Aunque AIML tiene una larga trayectoria en el desarrollo de Chatbots, hay razones para preferir la estructura y funcionalidad de sistemas NLP, especialmente en el contexto de este proyecto, ya que AIML puede no alcanzar la potencia deseable.

- Patrones vs aprendizaje automático. AIML se basa en una estructura de patrones predefinidos para gestionar las conversaciones, RASA usa algoritmos de aprendizaje automático y procesamiento de lenguaje natural (NLP) para entender y generar respuestas. Esto permite a RASA proporcionar respuestas más dinámicas y contextuales, y adaptarse a las variaciones en las consultas de los usuarios.
- Flexibilidad y adaptabilidad. A diferencia de AIML, RASA puede aprender y adaptarse continuamente a través de las conversaciones con los usuarios, mejorando su capacidad para entender conversaciones, proporcionando una experiencia de usuario más rica y personalizada.

Las ventajas, por tanto, del aprendizaje son:

- Aprendizaje Continuo. RASA tiene la capacidad de aprender de las conversaciones con los usuarios y mejorar sus respuestas a lo largo del tiempo, lo que es importante para mantener la eficacia del chatbot en un entorno cambiante.
- Manejo de Conversaciones Complejas. RASA puede gestionar conversaciones más complejas y contextuales gracias a su capacidad para entender el contexto de la conversación, algo que es más limitado y complejo en un sistema basado en AIML.
- Integración de Tecnologías de NLP. La integración de tecnologías avanzadas de NLP, como Spacy, permite a RASA realizar inferencias semánticas y entender el lenguaje natural a un nivel más profundo, mejorando la calidad y la relevancia de las respuestas del chatbot.

2.4.5. Mecanismos de aprendizaje

Supervisado

El aprendizaje supervisado en Chatbots se centra en mejorar y optimizar las capacidades de interacción del bot mediante el uso de datos etiquetados y ejemplos de conversaciones previas. En este enfoque, el Chatbot se entrena utilizando un conjunto de datos que contiene ejemplos de interacciones de usuario, donde las entradas (mensajes del usuario) y las salidas deseadas (respuestas del bot) ya deben encontrarse definidas y etiquetadas. Este conjunto de datos sirve como guía, permitiendo que el Chatbot aprenda a predecir la salida adecuada para una entrada dada, basándose en los ejemplos proporcionados durante el entrenamiento.

En el contexto de los Chatbots, el aprendizaje supervisado permite que el sistema aprenda a identificar y entender las intenciones detrás de las consultas de los usuarios y a generar respuestas apropiadas, utilizando ejemplos previos como referencia. Este método de aprendizaje es fundamental para desarrollar Chatbots que puedan interactuar de manera efectiva y coherente con los usuarios, proporcionando respuestas y soluciones que se alineen con las expectativas y necesidades del usuario.



Es importante destacar que, aunque el Chatbot puede interactuar y conversar con los usuarios, el aprendizaje y la mejora continua a partir de estas conversaciones no ocurren automáticamente. Para implementar un mecanismo de aprendizaje continuo basado en las interacciones en tiempo real, es necesario establecer un método de feedback dentro de la interfaz del chat, donde las respuestas del bot pueden ser evaluadas y corregidas si es necesario, y posteriormente reintegradas en el sistema para mejorar y refinar sus capacidades de respuesta y conversación.

Fases de aprendizaje

- **Recopilación de Datos.** Esta fase implica la recolección de datos relevantes, que pueden incluir interacciones pasadas, ejemplos de conversaciones, y otros datos pertinentes que pueden ser utilizados para entrenar al chatbot, y su preparación en el formato y soporte adecuado para poder ser utilizada en el entrenamiento posterior.
- **Entrenamiento del Modelo.** Los datos recopilados se utilizan para entrenar al Chatbot, ajustando sus algoritmos y modelos de aprendizaje automático para mejorar su capacidad para comprender y responder a las consultas de los usuarios.
- **Implementación y Uso.** Una vez entrenado, el chatbot se implementa y comienza a interactuar con los usuarios, utilizando su conocimiento y habilidades recién adquiridas para gestionar las conversaciones.
- **Retroalimentación y Mejora Continua.** El chatbot recibe retroalimentación continua a través de las interacciones con los usuarios y los operadores, y esta información se utiliza para realizar mejoras y ajustes adicionales, asegurando una mejora continua y adaptación a las necesidades cambiantes.

Aprendizaje en RASA

La implementación del aprendizaje en RASA se realiza a través de un proceso que puede entenderse como una serie de pasos lógicos que permiten al chatbot aprender

- **Recopilación de Conversaciones.** Inicialmente, se recopilan ejemplos de conversaciones reales procedentes de chats anónimos y facilitadas por Suma.
- **Identificación de Intenciones.** Las conversaciones recopiladas se analizan para identificar "intenciones", que son las metas o deseos subyacentes que los usuarios expresan a través de sus mensajes.
- **Entrenamiento del Chatbot:** Una vez que las conversaciones y las intenciones están definidas y estructuradas, se introducen en RASA, que utiliza esta información para entrenar al Chatbot. Durante el entrenamiento, el Chatbot aprende a identificar intenciones de los mensajes de los usuarios y a responder de acuerdo con esta información.
- **Uso y Mejora Continua.** Posteriormente, cuando el chatbot interactúa con los usuarios, utiliza este aprendizaje para guiar sus respuestas y comportamientos. Además, las conversaciones que tiene con los usuarios pueden ser revisadas y, si es necesario, utilizadas para entrenamiento adicional y mejora del Chatbot.
- **Se habilitará la opción para cargar un archivo que contenga un conjunto de preguntas,** que representan las solicitudes que podría realizar un ciudadano, con el propósito de clasificar cada pregunta en el sistema de intenciones ya establecido. Una vez cargado el conjunto de preguntas, el sistema procederá a categorizar cada una de ellas en uno de los dos grupos siguientes:

- Preguntas categorizadas. El sistema ha sido capaz de asociar la nueva pregunta a una de las intenciones del Chatbot. El usuario administrador de la plataforma de gestión deberá validar si dicha asignación es correcta.
- Preguntas no categorizadas. El sistema no es capaz de categorizar la pregunta a ninguna intención de la base de datos de conocimiento. En este caso, el usuario administrador de la herramienta de gestión deberá asignar la pregunta a la intención adecuada.

2.4.6. Terminología

NLP (Procesamiento de Lenguaje Natural). NLP es una rama de la IA que se trata en la interacción entre las computadoras y los humanos a través del lenguaje natural. El objetivo de NLP es leer, descifrar, entender y dar sentido a la lengua humana para la máquina. Esto se logra a través de diversas técnicas y algoritmos que permiten a la máquina interpretar, entender y responder a las solicitudes de los usuarios en un lenguaje natural.

Inferencia Semántica. La inferencia semántica se refiere al proceso mediante el cual los sistemas de IA deducen el significado o la intención detrás de las palabras y frases de texto. Este proceso va más allá de la simple coincidencia de palabras, permitiendo al sistema entender las relaciones y contextos en los que se usan estas palabras y frases. De esta forma puede generar respuestas o acciones que son relevantes y con un contexto correcto.

Tokenización. La tokenización es un elemento fundamental en NLP. Implica dividir el texto en unidades más pequeñas, los tokens, que generalmente corresponden a palabras, términos o en algunos casos partes de las palabras. Este proceso facilita el análisis posterior del texto, permitiendo que los algoritmos de NLP trabajen a nivel de palabra.

Lematización. La lematización es el proceso de reducir las palabras a su forma base o raíz, permitiendo que palabras con diferentes conjugaciones o formas se agrupen como una única entidad. Por ejemplo, las palabras "corriendo" y "corrió" se reducirían a su lema, que es "correr". Este proceso ayuda a simplificar el texto y a reducir la complejidad del lenguaje natural, facilitando el trabajo para los sistemas NLP.

Identificación de Entidades. La identificación de entidades se refiere al proceso de detectar y clasificar entidades nombradas, como nombres de personas, organizaciones, lugares, fechas, teléfonos, DNI... dentro del texto. Este proceso permite comprender mejor el contexto y el contenido del texto, identificando los elementos que pueden ser relevantes para entender y responder a las solicitudes.

2.4.7. Gestión de conceptos y propiedades

El sistema de chatbot a implementar permitirá a los administradores adaptar el comportamiento del chatbot a necesidades específicas mediante la creación y modificación de "Conceptos" y "Propiedades". Esta sección describe el mecanismo a través del cual se logra esta adaptabilidad, destacando la capacidad de personalización que ofrece el sistema a los administradores.

Conceptos

Los "Conceptos" actúan como entidades y *slots* dentro del *framework* de Rasa, representando los datos específicos que el chatbot necesita recopilar durante una conversación. Cada Concepto es definido por el administrador, lo que permite modificar la información que se puede recoger en cada propiedad. Un ejemplo podría ser:

- Concepto: “Servicios municipales”: limpieza, seguridad, aperturas, etc.
- Concepto: “Horarios de atención”: de mañana, de tarde, todo el día, etc.
- Concepto: “Impuestos”: IVTM, IAE, IBI, etc.
- Concepto: “Municipios”: Elche, Benidorm, Cox, etc.

Propiedades

Las "Propiedades", equivalentes a los formularios en Rasa, se componen de Conceptos y se activan cuando se detecta una intención específica del usuario. Al crear una Propiedad, el administrador selecciona qué Conceptos son necesarios para completarla, estableciendo así una estructura de recopilación de datos precisa y orientada al objetivo.

- Abierto: (limpieza, todo el día).
- Abierto: (aperturas, de mañana).
- Abierto (seguridad, todo el día).

Mecanismo de Funcionamiento y Configuración

Al detectar una intención relacionada con una Propiedad específica, el sistema activa el formulario correspondiente, iniciando un proceso automatizado que busca recopilar los Conceptos requeridos. Si la intención inicial ya incluye información sobre algunos de estos Conceptos, el sistema la reconoce y almacena automáticamente en los *slots* correspondientes, evitando preguntas redundantes al usuario.

La configuración de Conceptos y Propiedades se realiza a través de una interfaz gráfica diseñada para ser intuitiva y accesible para el administrador, eliminando la necesidad de interactuar directamente con el código subyacente de Rasa. Este enfoque garantiza que los administradores puedan enfocarse en la lógica de interacción y la experiencia del usuario, sin preocuparse por los detalles técnicos de la implementación.

Uso de Sinónimos para la Estandarización de Conceptos

Para mejorar la precisión y eficiencia del chatbot, el sistema permite la definición de sinónimos para los Conceptos. Esto facilita la estandarización de los valores recopilados, minimizando duplicidades y asegurando que el chatbot pueda interpretar y procesar variaciones en las respuestas de los usuarios de manera efectiva. Esta característica es fundamental para el entrenamiento y mejora continua del sistema, permitiendo la incorporación de nuevos elementos y variantes de forma sencilla.

Configuración de Respuestas Post-Formulario

El sistema proporciona a los administradores la capacidad de configurar las respuestas que el chatbot entregará una vez completado el formulario. Los administradores pueden definir respuestas específicas en la interfaz de administración, las cuales pueden ser condicionales basadas en la información recopilada por el formulario. Esto significa que el chatbot puede estar configurado para responder de manera diferente dependiendo de si puede proceder con el trámite solicitado por el usuario, si requiere información adicional, o si el trámite no es gestionable.



Parametrización de la Información de Respuesta

La capacidad de parametrizar las respuestas permite al chatbot entregar información detallada y específica relacionada con el trámite en cuestión. Por ejemplo, si el chatbot puede gestionar el trámite solicitado, proporcionará al usuario la información necesaria para avanzar, como pasos a seguir, documentos requeridos o enlaces útiles. En caso de que el chatbot determine que el trámite no es gestionable, informará al usuario adecuadamente, ofreciendo opciones alternativas o canales de soporte adicional si están disponibles.

Conclusión

A través de una interfaz intuitiva, los administradores pueden definir y modificar Conceptos y Propiedades, adaptando el comportamiento del Chatbot a las necesidades específicas de su audiencia, todo mientras mantienen control sobre el proceso de recopilación de datos y la lógica de conversación. Esta capacidad de adaptación asegura que el chatbot pueda evolucionar y mejorar continuamente, respondiendo a los cambios en las demandas de los usuarios y las necesidades de los administradores.

2.5. Intervención de un agente

El sistema estará preparado y facilitará la intervención o participación de un agente en una conversación cuando sea necesario. El sistema podrá detectar la necesidad de intervención de un agente en una conversación teniendo en cuenta lo siguiente:

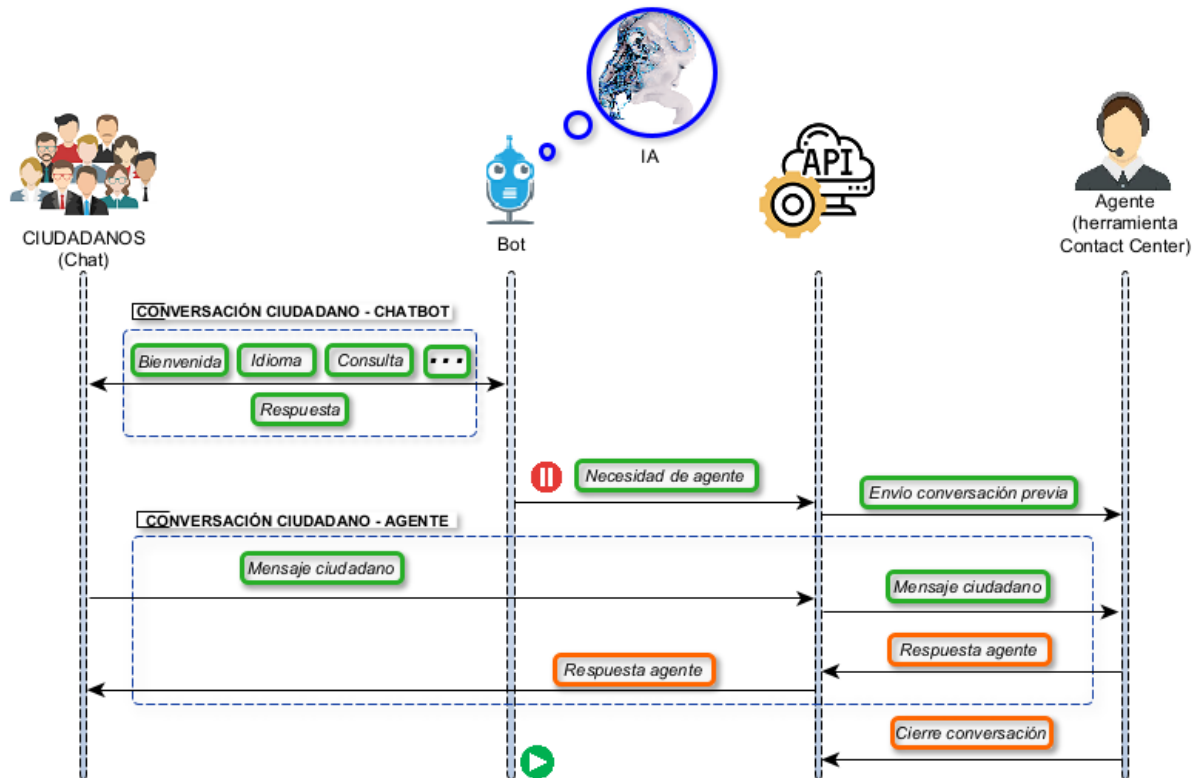
- El modelo de inteligencia artificial propone la intervención de un agente en base a la consulta generada por parte del usuario.
- El usuario indica por segunda vez consecutiva que la solicitud no ha sido resuelta de manera satisfactoria.
- El usuario indica de manera explícita el deseo de hablar con un agente.

Para lograr esto, se establece un mecanismo de notificación vía *webhook*, que alerta sobre la intervención requerida del agente. Esta notificación incluirá la conversación completa entre el ciudadano y el Chatbot, brindando contexto al agente para una mejor comprensión de la situación.

Durante la participación del agente en la conversación, el sistema debe facilitar un intercambio de mensajes fluido y de manera bidireccional. Esto implica que tanto el agente como el usuario pueden enviar mensajes entre sí de manera activa.

Por último, se contempla la posibilidad de cerrar una conversación por parte del agente, lo que activará nuevamente al asistente conversacional para continuar la interacción automática si es necesario.

A continuación, se adjunta un diagrama de flujo que representa de manera gráfica el detalle proporcionado anteriormente:



A nivel de configuración, la herramienta permitirá realizar los siguientes ajustes:

- Configurar si un Chatbot permite la intervención de un agente.
- Horarios de atención de un agente. Esta flexibilidad proporciona control sobre cuándo y cómo se realiza la transición entre la automatización y la intervención humana.
 - Configuración de horario especial. Este horario prevalecerá sobre el horario anterior.
- Configuración de festivos, de manera que el sistema cuente con la información suficiente para proceder o no con el desvío de la conversación hacia un agente.
- Mensajes
 - Fuera de horario y festivo.
 - Inicio de intervención del agente.
 - Respuestas ante consultas que escapen del alcance de la base de conocimiento del Chatbot.

2.6. Herramienta web de gestión

La solución de Chatbot propuesta va acompañada de una herramienta web de gestión, proporcionando a los usuarios una plataforma centralizada y completa para administrar y supervisar todos los aspectos relacionados con la implementación, operación y análisis del chatbot.

El punto de entrada principal a la herramienta web de gestión es a través del proceso de inicio de sesión (login), donde los usuarios autorizados pueden acceder de manera segura a todas las funcionalidades y datos relevantes. Una vez dentro, los usuarios tienen acceso a un conjunto integral de métricas y KPIs que ofrecen una visión detallada del rendimiento y la eficacia del chatbot en tiempo real.

La herramienta web de gestión también ofrece funcionalidades específicas para gestionar clientes, chatbots, conversaciones, temáticas e intenciones, usuarios y ajustes. Los usuarios pueden agregar y administrar nuevos clientes, configurar múltiples chatbots con sus respectivos flujos conversacionales, supervisar y analizar conversaciones individuales, gestionar y organizar temáticas e intenciones para mejorar la precisión en las respuestas del chatbot y ajustar configuraciones para adaptar el comportamiento y la apariencia del chatbot.

A continuación, en los sucesivos apartados, se analiza, con mayor detalle, cada una de las funcionalidades presentes en dicha herramienta web.

2.6.1. Login

La herramienta web contará con un sistema de login con el fin de garantizar un acceso seguro a la plataforma mediante el uso de credenciales de usuario y contraseña, evitando con ello el acceso no autorizado y garantizando, por tanto, la confidencialidad de los datos almacenados en la plataforma.

The image shows a login interface with the title 'Acceso' and the subtitle 'Iniciar sesión en su cuenta'. It contains two input fields: 'Correo electrónico*' and 'Contraseña*'. Below these fields is a blue button labeled 'Acceso'.

Los usuarios de la plataforma contarán con los roles y privilegios de acceso adecuados. Los perfiles definidos son: superadministrador, administrador, editor y supervisor. Cada uno de estos roles tendrá diferentes niveles de acceso y permisos dentro de la plataforma.

El título de la plataforma, será configurable y visible desde la pantalla de inicio de sesión.

2.6.2. Métricas

La herramienta de gestión del Chatbot debe proporcionar un panel de control completo que incluya indicadores clave de rendimiento (KPIs) y estadísticas relevantes sobre el desempeño del Chatbot. Este panel de control permitirá monitorear y evaluar la eficacia del Chatbot en la interacción con los usuarios.

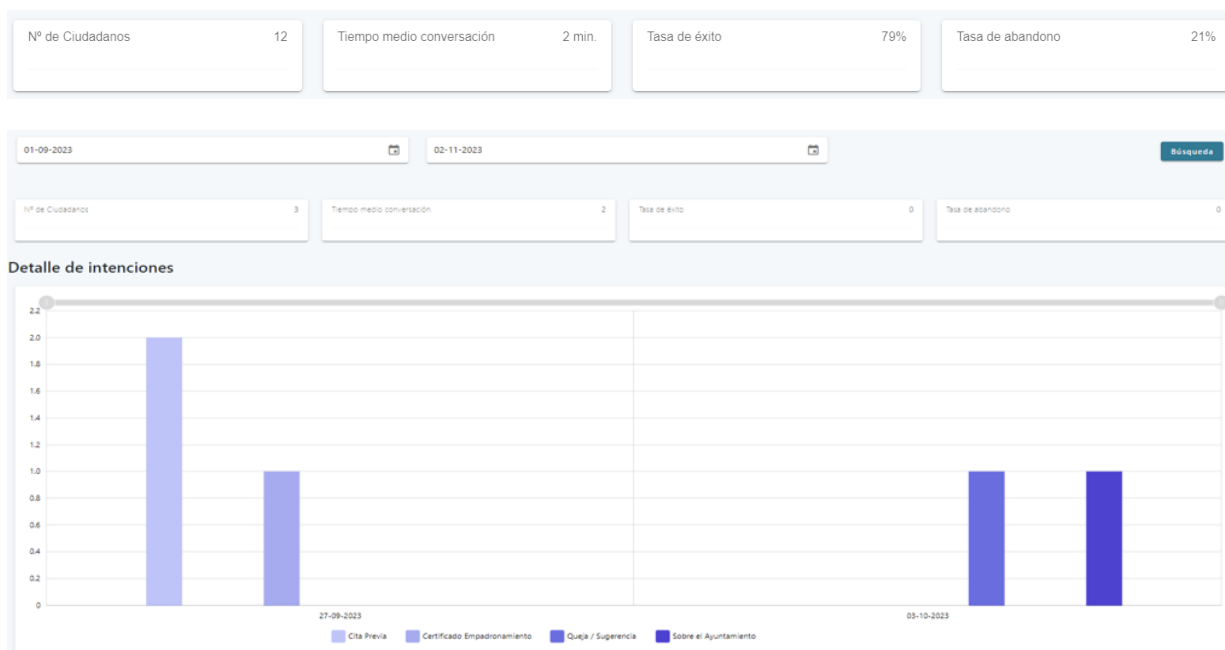
Dentro de este panel de control, la herramienta deberá ofrecer gráficos visuales que representen de manera clara y comprensible las diferentes métricas. Toda esta información podrá ser consultada en un intervalo de fechas predefinido.

Entre los indicadores y métricas que se podrán consultar se encuentran:

- Total de ciudadanos que han mantenido una conversación con el Chatbot.
- El tiempo medio de conversación. Tiempo medio de todas las conversaciones en el rango de fechas seleccionado. Una conversación finaliza en los siguientes casos:
 - El ciudadano indica que no desea realizar otra gestión.

- El ciudadano no interactúa durante un tiempo previamente configurado en el apartado “Ajustes”.







- La tasa de éxito del Chatbot basada en el *feedback* del ciudadano. Porcentaje de respuestas correctas sobre el total de respuestas del asistente conversacional para el tipo de pregunta abierta
- El número de chats abandonados. Porcentaje de conversaciones que no llegan a un estado final sobre el total de conversaciones. El ciudadano deja de interactuar con el Chatbot y el tiempo máximo para responder es excedido.
- Porcentaje de las intenciones más consultadas por los ciudadanos.



Es importante destacar que estas estadísticas podrán visualizarse de manera individual para cada cliente y Chatbot. Además, el superadministrador tendrá acceso a visualizar las estadísticas de todos los clientes y Chatbots, mientras que los administradores, editores y supervisores solo podrán consultar las estadísticas de sus propios clientes.

2.6.3. Clientes

La herramienta de gestión permitirá crear, visualización, editar y modificar los diferentes clientes.

Clientes				AGREGAR
Nombre	Información	Creador	Fecha creación	
Cliente 1	Información cliente 1	Superadministrador	2/2/2024, 20:18:22	  
Cliente 2	Información cliente 2	Superadministrador	2/2/2024, 20:15:54	  

Items per page: 10 1-10 of 17

Cada cliente registrado en el sistema contendrá una serie de datos almacenados. Estos datos incluyen:

- Nombre

- Información adicional de interés.
- Usuario que realizó la creación del cliente.
- Fecha de creación.

Además, desde cada cliente se puede acceder al listado de Chatbots disponibles, por tanto, un cliente puede albergar múltiples Chatbots.



La creación y administración de los diferentes clientes será exclusiva del superadministrador. Además, tendrá también la responsabilidad de crear y asignar los administradores a cada cliente.

Los administradores y editores tendrán la capacidad de modificar el nombre e información adicional de interés de los clientes. Además, los administradores podrán gestionar los usuarios con acceso a los diferentes Chatbots asociados a cada cliente, lo que incluye la capacidad de añadir y eliminar usuarios según sea necesario.

2.6.4. Chatbots

La herramienta web debe permitir la creación, visualización, edición y modificación de los diferentes Chatbots. Para lograr esto, el sistema mantendrá una lista actualizada de todos los Chatbots existentes, con una serie de datos asociados a cada uno de ellos. Esta información incluye:

- Nombre del Chatbot.
- Cliente al que pertenece.
- Los idiomas configurados para gestionar las conversaciones.
- Usuario que realizó la creación del Chatbot y la fecha de creación.

Chatbots					AGREGAR	RECARGAR
Nombre	Cliente	Idioma	Creador	Fecha creación		
Chatbot 1	Cliente 1	Castellano, valenciano, inglés	Superadministrador	2/2/2024, 17:44:45		
Chatbot 2	Cliente 2	Castellano	Administrador	2/2/2024, 20:19:40		
					Items per page: 10	1-2 of 2

Además, se proporcionará acceso al listado de intenciones y a la configuración del flujo conversacional para cada Chatbot.

Los superadministradores y administradores serán los únicos roles con privilegios para crear y administrar los Chatbots.

La herramienta permitirá definir, para cada Chatbot, el flujo conversacional correspondiente, de manera visual, mediante la configuración de un sistema de estados y transiciones. Esto incluye la posibilidad de exportar la lógica del flujo conversacional para ser incorporada a cualquier página web. Para mayor información ir al apartado “Diseño del flujo conversacional”.

Finalmente, deben existir una serie de ajustes y configuraciones adicionales para cada Chatbot, incluyendo:

- Especificación de la capacidad de intervención de un agente en las conversaciones.

- Configuración del horario de atención del agente y horario especial que prevalecerá sobre el anterior.

Configuración horario

Día	Inicio Franja 1	Fin Franja 1	Inicio Franja 2	Fin Franja 2
Lunes	00:00	14:00	14:00	23:59
Martes	00:00	14:00	14:00	23:59
Miércoles	00:00	14:00	14:00	23:59
Jueves	00:00	14:00	14:00	23:59
Viernes	00:00	14:00	14:00	23:59
Sábado	00:00	14:00	14:00	23:59
Domingo	00:00	14:00	14:00	23:59

- Gestión de diferentes mensajes.
 - Fuera de horario o festivo, para trasladar al usuario que en esos momentos la conversación no puede ser atendida por un agente.
 - Mensaje para indicar que un agente entrará a atender la conversación.
 - Mensaje para indicar que el Chatbot no admite la intervención de un agente ante dicha petición por parte del usuario.
 - Mensaje genérico ante una petición para la que el Chatbot no tiene respuesta, ya que escapa del alcance de la base de datos de conocimiento.
- Gestión de festivos.

Festivos			AÑADIR
			Búsqueda
Nombre	Día	Descripción	
Festivo Local	05/02/2024	Información a trasladar al usuario, motivo del festivo	
Filas por página 10 1-1 de 1			

- Tiempo de inactividad del chat.
- Configuración de idiomas
- Porcentaje de precisión del Chatbot para determinar la asociación a una intención.

2.6.5. Conversaciones

La herramienta web de gestión proporcionará un listado detallado de todas las conversaciones mantenidas por el Chatbot, lo que permite un seguimiento exhaustivo de la interacción entre el asistente virtual y los usuarios. La información que se mostrará este listado será:

- Fecha de creación de la conversación
- Identificador de usuario. Este identificador será aleatorio, ya que no se solicita ningún tipo de información al ciudadano para ser identificado.
- Fecha del primer y último mensaje intercambiado.

- Duración total de la conversación.
- Número total de mensajes intercambiados entre el usuario y el Chatbot.
- La intervención o no de un agente.
- Estado de la conversación: en curso o finalizada.
- Resultado de la conversación a partir del *feedback* generado por parte del ciudadano, que puede ser: positiva, negativa o sin valoración.

Conversaciones						
Fecha creación	Usuario	Fecha primer mensaje	Fecha último mensaje	Horas y minutos	Mensajes totales	
27-09-2023 16:52:36	Usuario1695833555	27-09-2023 18:52:36	27-09-2023 18:56:46	00:04	49	
28-09-2023 06:55:10	Usuario1695884109	28-09-2023 08:55:10	28-09-2023 08:55:10	00:01	4	
03-10-2023 10:50:40	Usuario1696330240	03-10-2023 12:50:40	03-10-2023 12:52:58	00:02	24	
Filas por página 10 1-3 de 3 < >						

Por su parte, la herramienta debe permitir también la visualización del contenido completo de cada conversación, lo que implica acceder a todos los mensajes intercambiados entre el usuario, el Chatbot y el agente, en su caso.

Conversación			
Fecha creación	Tipo usuario	Intención	Mensaje
2023-09-27 18:52:36	Bot		¡Hola! Le damos la bienvenida al asistente virtual de la web de la Diputación de Alicante
2023-09-27 18:52:36	Bot		Seleccione una de estas opciones: Sobre la diputación de Alicante, Quejas y sugerencias, Solicitud de certificado de información, Solicitud de certificado de empadronamiento, Quiero preguntar otra cosa
2023-09-27 18:52:39	Usuario	Información	Sobre la diputación de Alicante
2023-09-27 18:52:40	Bot		Puede encontrar más información sobre la diputación de Alicante en el siguiente enlace
2023-09-27 18:52:50	Bot		¿Hemos resuelto su consulta?
2023-09-27 18:52:50	Usuario		Sí
2023-09-27 18:53:33	Bot		¿Desea realizar otra consulta?
2023-09-27 18:53:33	Usuario		No
2023-09-27 18:53:40	Bot		Gracias por usar el asistente virtual de la Diputación de Alicante

CERRAR

Por último, las conversaciones debe poder ser filtradas en función de diferentes criterios:

- Fecha de la conversación.
- Estado.
- Resultado de cada conversación.

Búsqueda ^

05-02-2024

05-02-2024

Buscar

2.6.6. Temáticas e intenciones

La herramienta web de gestión permitirá la visualización, creación, modificación y eliminación de las diferentes temáticas, que actúan como contenedores de intenciones. Esta responsabilidad recae en el superadministrador, quien será el encargado de generar una lista única de temáticas. Este enfoque centralizado facilitará el mantenimiento y el intercambio de estos bloques de intenciones entre los diferentes Chatbots.

Por otro lado, la herramienta debe permitir la creación, modificación y eliminación de intenciones que guíen las respuestas del Chatbot. Los roles autorizados para gestionar estas intenciones son los superadministradores, administradores y editores. Cada nueva intención creada deberá asociarse a una de las temáticas previamente establecidas por el superadministrador.

La herramienta debe mostrar un listado detallado de las intenciones creadas dentro de la configuración de cada Chatbot. Este listado incluirá información relevante como:

- Nombre de la intención.
- Temática.
- Fecha de creación.
- El usuario que realizó la creación.
- Fecha de modificación.
- Método de creación (nueva o importada).

CHATBOTS > CONOCIMIENTO

Chatbot: Gestión tributaria

TEMÁTICAS

INTENCIONES

CONCEPTOS

LISTAS

Búsqueda

Intenciones

Buscar

AGREGAR

RECARGAR

Intención	Información	Temática	Fecha creación	Fecha modificación	Método creación	Creador	Opciones
Pagar impuesto	N/A	Pagos	13/3/2024, 15:03:08	13/3/2024, 15:03:08	WEB	devAunna SuperAdmin	<div><div></div><div></div><div></div></div>

Elementos por página:

10

1-1 de 1

Además, se debe permitir acceder al detalle de cada intención para realizar las modificaciones pertinentes.

1 Datos intención

2 Conceptos

3 Preguntas

4 Respuestas

5 Gestión respuestas

Temáticas
Pagos

Intención
Pagar impuesto

Información
N/A

ANTERIOR

SIGUIENTE

Cada intención queda definida por un conjunto de preguntas y respuestas, lo que es una regla para el modelo de IA. El sistema debe permitir, por tanto, la creación y administración de preguntas y respuestas en los tres idiomas admitidos (castellano, valenciano e inglés).

Por último, el sistema permitirá administrar y mantener un listado de sinónimos por cada Chatbot con el fin de optimizar y evitar la redundancia en el conjunto de preguntas asociadas a cada intención. Por ejemplo, las siguientes tres preguntas podrían ser representadas en una única:

- ¿Cómo puedo pagar el IBI?
 - ¿Cómo puedo pagar la contribución?
 - ¿Cómo puedo pagar la casa?
- ¿Cómo puedo pagar el [IBI](tipo de impuesto)?
Sinónimos: IBI, contribución, casa.

Para conseguir lo anterior, la creación de una intención consta de 5 pasos:

1. Temática y nombre de la intención.
2. Asociar conceptos a una intención.
3. Preguntas de la intención.
4. Respuesta positiva y negativa.
5. Combinación de conceptos para indicar si devolvemos una respuesta positiva o negativa.

A continuación, se adjuntan una serie de pantallazos que representan los puntos anteriores:

1 Datos intención

2 Conceptos

3 Preguntas

4 Respuestas

5 Gestión respuestas

Búsqueda

Conceptos

Buscar

AGREGAR

RECARGAR

<input checked="" type="checkbox"/>	Nombre	Pregunta	Fecha de creación	Opciones
<input checked="" type="checkbox"/>	Tipo de impuesto	¿Para qué impuesto desea realizar la gestión?	13/3/2024, 14:52:20	
<input checked="" type="checkbox"/>	Municipios	¿Para qué municipio desea realizar la gestión?	13/3/2024, 14:56:08	

1

Datos intención

2

Conceptos

3

Preguntas

4

Respuestas

5

Gestión respuestas

Pregunta (Castellano)  

Me gustaría pagar el

ANTERIOR

Concepto


Tipo de impuesto

Termino

IBI

AGREGAR

CANCELAR

Pregunta (Castellano)  

Me gustaría pagar el [IBI](Tipo de impuesto)

1

Datos intención

2

Conceptos

3

Preguntas

4

Respuestas

5

Gestión respuestas

Respuesta positiva

Respuesta (Castellano)  

Puede realizar el pago del impuesto (Tipo de impuesto) accediendo al siguiente enlace...

Respuesta negativa

Respuesta (Castellano)  

No gestionamos el pago para el impuesto (Tipo de impuesto), debe dirigirse a la oficina...

ANTERIOR

SIGUIENTE

1	2	3	4	5
Datos intención	Conceptos	Preguntas	Respuestas	Gestión respuestas
Group			Respuesta	
▼ IAE (3)				
		Elche	<input type="checkbox"/>	
		Cox	<input checked="" type="checkbox"/>	
		Benidorm	<input type="checkbox"/>	
> IVTM (3)				
▼ IBI (3)				
		Elche	<input checked="" type="checkbox"/>	
		Cox	<input checked="" type="checkbox"/>	
		Benidorm	<input type="checkbox"/>	

Cada concepto queda representado por una lista de términos, y cada término por un conjunto, opcional, de sinónimos:

Nombre de Lista

Tipos impuestos

Agregar terminos asociados +

IAE

IBI

Nombre

IBI

Sinónimos

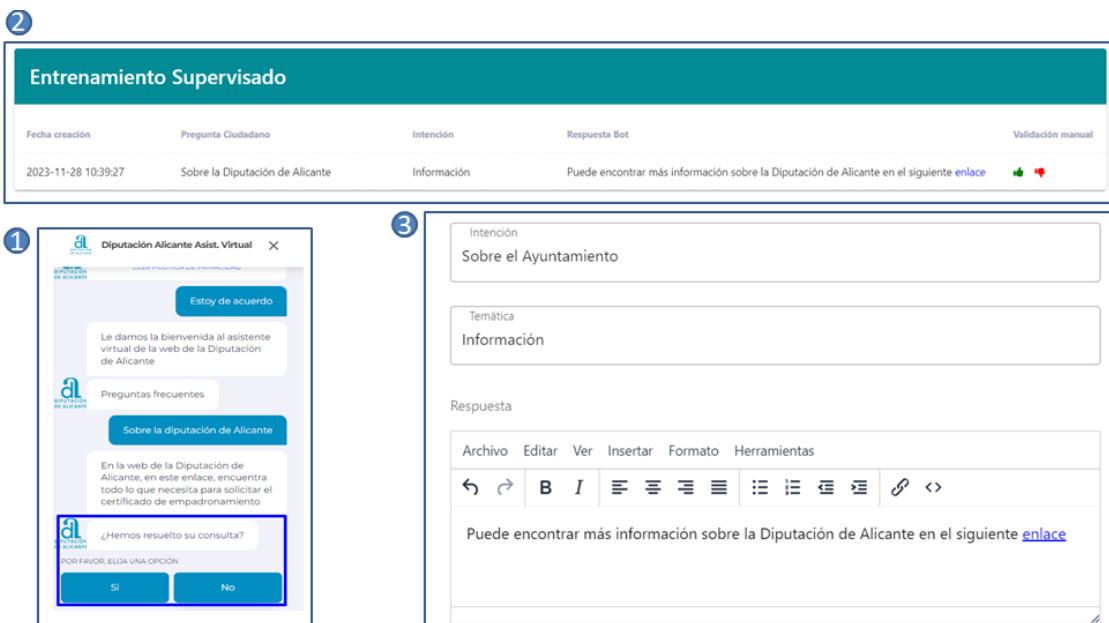
Casa

Contribución

piso

IVTM

2.6.7. Entrenamiento supervisado



La herramienta debe facilitar un entorno de entrenamiento supervisado, donde los superadministradores y administradores puedan mejorar las respuestas del Chatbot. Para lograr esto, la herramienta proporcionará una interfaz intuitiva para la revisión y validación de las respuestas generadas por el Chatbot. Esta interfaz estará estructurada en cuatro secciones claramente definidas (2):

- **Respuestas negativas:** El usuario ha indicado que la solicitud no ha sido resuelta de manera satisfactoria por parte del Chatbot (1). Esto permite identificar las respuestas que necesitan mejorar o ajustarse para ofrecer una mejor experiencia al usuario.
- **Respuestas sin valoración:** El usuario no ha indicado si la respuesta del Chatbot fue satisfactoria o no (1). Esta categoría puede ayudar a identificar áreas de mejora donde los usuarios podrían no estar completamente satisfechos con la respuesta proporcionada.
- **Respuesta positiva:** El usuario indica que la solicitud ha sido resuelta de manera satisfactoria por parte del Chatbot (1). Esto permite identificar qué respuestas son efectivas y están cumpliendo con las expectativas de los usuarios.
- **Sin categorizar:** Solicitudes que no pueden ser asociadas a ninguna intención existente en el Chatbot (1). Esto puede deberse a que la intención aún no ha sido creada o que la solicitud se encuentra fuera del alcance de las respuestas definidas para el Chatbot. Esta categoría proporciona información valiosa sobre áreas donde el Chatbot puede necesitar ampliar su base de datos de conocimiento.

los superadministradores y administradores de la herramienta podrán:

- Validar como positiva la respuesta del Chatbot (pulsa sobre *thumbs up*).
- Validar como negativa la respuesta del bot (pulsa sobre *thumbs down*). En este caso aparece el pop-up (3) para asociar la solicitud del ciudadano a una intención y temática.
 - La solicitud del ciudadano pertenece a otra intención y temática existentes:
 - Selección de la temática adecuada.
 - Tras seleccionar la temática, se carga la lista de intenciones de esa temática.

- Tras seleccionar una intención de la lista se carga la respuesta del Chatbot y las preguntas asociadas.
- La pregunta formulada por el ciudadano debe quedar cargada con el resto de preguntas ya configuradas.
- La solicitud del ciudadano pertenece a otra intención diferente a la seleccionada por parte del asistente virtual y NO existe. En este caso se debe crear la nueva intención.
- La solicitud del ciudadano escapa del alcance de la base de conocimiento del Chatbot. En este caso el asistente virtual no es capaz de categorizar la pregunta a ninguna intención. Estas preguntas quedan almacenadas en la cuarta y última pestaña. El administrador podrá crear la nueva intención o descartar la solicitud del usuario.

2.6.8. Usuarios

La herramienta debe permitir agregar, modificar y eliminar cuentas de usuarios con diferentes niveles de acceso. Para ello, la herramienta contará con los diferentes perfiles: superadministrador, administrador, editor y supervisor.

Cualquier usuario de la herramienta podrá acceder a su perfil para consultar su información personal y actualizar su nombre y contraseña. Además, podrán finalizar su sesión en cualquier momento, lo que garantiza la privacidad y seguridad de sus datos al utilizar la herramienta.

El superadministrador tendrá la capacidad de asignar roles específicos (administrador, editor, supervisor) a cada usuario registrado. Además, será el encargado de indicar para cada uno de ellos los clientes a gestionar.

Los usuarios solo pueden acceder a las funciones y secciones autorizadas por su rol específico, lo que contribuye a mantener la seguridad y la integridad de la plataforma.

Usuarios

Seleccionar estado (T...

Nuevo usuario

Filas por página10

Búsqueda

Usuarios	Rol	Activo	Acción
<div>ÁF</div> <div>Ángel Fuentes</div> <div>afuentes@aunnait.es</div>	SuperAdmin	<div></div>	<div></div> <div></div>
<div>BO</div> <div>bot1</div> <div>bot@gmail.com</div>	Sistema	<div></div>	<div></div> <div></div>
<div>CU</div> <div>customer1</div> <div>customer1@gmail.com</div>	Cliente	<div></div>	<div></div> <div></div>

1 de 1

Página anterior

1

Página siguiente

2.6.9. Ajustes generales

La plataforma web permitirá a los superadministradores realizar los siguientes ajustes a nivel general:

- Cargar y modificar el logo corporativo de la herramienta. Este logo se mostrará tanto en la cabecera superior de la plataforma web como en el *footer*.
- Personalizar el color corporativo de la interfaz de la herramienta web.



- Personalizar el título de la plataforma. Este título aparecerá en la página de login y *footer*.

3. CAPA DE INTEGRACIÓN

Una de las funciones principales de la capa de integración es permitir que todas las operaciones realizadas desde la interfaz de usuario de la herramienta web de gestión se lleven a cabo de manera transparente a través de API.

Además, la capa de integración facilita el acceso a la información generada en tiempo real por el chatbot, incluyendo conversaciones en curso, estadísticas, métricas, entre otros datos relevantes. Esta capacidad de acceso en tiempo real permite a los usuarios monitorear y analizar el rendimiento del chatbot.

Otro aspecto de la capa de integración es su capacidad para permitir el acceso de un agente a las conversaciones en curso. Esto facilita la intervención humana cuando sea necesario. La capa de integración actúa como un canal de comunicación bidireccional, facilitando la interacción entre el chatbot y los agentes humanos.

A continuación, se detallan cada uno de los métodos que estarán disponibles por medio del Servicio Web correspondiente a construir.

3.1. Autenticación

El método `getToken` devuelve un token de seguridad que es necesario para realizar el resto de peticiones que se detallan a continuación. Este token tendrá una validez de 5 minutos desde el momento en que se solicita, tiempo que se renovará por 5 minutos después de cada petición. Una vez caducados esos 5 minutos, se debe volver a solicitar un nuevo token.

Obtención de token de autenticación	
Método	<code>getToken</code>
Tipo	POST
Entrada	<pre>{ Usuario: String Clave: String }</pre>
Salida	<pre>OK: { estado: String, token: String }</pre>

	<pre>KO: { estado: String, motivo_error: string }</pre>
--	---

3.2. Clientes

El sistema permitirá la creación, modificación, obtención y eliminación de los diferentes clientes por medio del servicio web correspondiente.

Mediante la petición a los siguientes servicios, se puede obtener y editar la información relacionada con los clientes. El método *getClientes* nos proporciona el *cliente_id*, necesario para poder identificar a cada cliente en el resto de los *endpoints*.

Obtener clientes	
Método	getClientes
Tipo	GET
Entrada	<pre>{ token: String (Obtenido de getToken) }</pre>
Salida	<pre>OK: { estado: String, clientes: { { cliente_id: String nombre: String, informacion: String, usuario: String, fecha_creacion: fecha } } }</pre>
	<pre>KO: { estado: String, motivo_error: String }</pre>

El único campo modificable en un cliente es su nombre, siendo el resto de datos fijos o asignados directamente por el sistema, a modo de identificador interno.

Modificar cliente	
Método	modificarCliente

Tipo	POST
Entrada	<pre>{ token: String (Obtenido de getToken), cliente_id: String (Obtenido de getCientes), nombre: String }</pre>
Salida	OK: <pre>{ estado: String }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Para la creación de un cliente, es necesario conocer su nombre, siendo este dato obligatorio. El otro campo aceptado en la creación de cliente, además del token, es la información de interés. La información de interés es un dato opcional. Este método devuelve el identificador de cliente asignado a nivel interno en el sistema.

Crear cliente	
Método	crearCliente
Tipo	PUT
Entrada	<pre>{ token: String (Obtenido de getToken), nombre: String, informacion: String }</pre>
Salida	OK: <pre>{ estado: String, cliente_id: String }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Eliminar cliente	
Método	eliminarCliente

Tipo	DELETE
Entrada	<pre>{ token: String (Obtenido de getToken), cliente_id: String (Obtenido de getClientes) }</pre>
Salida	OK: <pre>{ estado: String }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

3.3. Chatbot

El sistema permitirá la creación, modificación, obtención y eliminación de los diferentes Chatbots por medio del servicio web correspondiente.

El endpoint `getChatbots` devuelve un listado de los identificadores de los chatbots, para poder usarlos en el resto de las peticiones asociadas a estos.

Obtener Chatbots	
Método	getChatbots
Tipo	GET
Entrada	<pre>{ token: String (Obtenido de getToken) }</pre>
Salida	OK: <pre>{ estado: String, listado_chatbots:{ {chatbot_id: String}, ... } }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Obtener Chatbot	
Método	getChatbot
Tipo	GET
Entrada	<pre>{ token: String (Obtenido de getToken), chatbot_id: String }</pre>
Salida	OK: {estado, datos del chatbot}
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Modificar Chatbot	
Método	modificarChatbot
Tipo	POST
Entrada	<pre>{ token: String (Obtenido de getToken), chatbot_id: String (Obtenido de getChatbots), chatbot: JSON (JSON con el nuevo chat) }</pre>
Salida	OK: <pre>{ estado: String }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Crear Chatbot	
Método	crearChatbot
Tipo	PUT
Entrada	<pre>{ token: String (Obtenido de getToken), chatbot: JSON (JSON con el nuevo chat) }</pre>

	}
Salida	OK: { estado: String, chatbot_id: String }
	KO: { estado: String, motivo_error: String }

Eliminar Chatbot	
Método	eliminarChatbot
Tipo	DELETE
Entrada	{ token: String (Obtenido de getToken), chatbot_id: String (Obtenido de getChatbots) }
Salida	OK: { estado: String }
	KO: { estado: String, motivo_error: String }

3.4. Intenciones

El sistema permitirá la creación, modificación, obtención y eliminación de las diferentes intenciones por medio del servicio web correspondiente.

Obtener intenciones	
Método	getIntenciones
Tipo	GET
Entrada	{ token: String (Obtenido de getToken) }

Salida	OK: <pre> { estado: String, lista_intenciones{ { intencion: { clave: String, frases: { {frase: String}, ... } respuestas: { {frase: String}, ... } }, ... } } } </pre>
	KO: <pre> { estado: String, motivo_error: String } </pre>

Modificar intención	
Método	modificarIntencion
Tipo	POST
Entrada	<pre> { token: String (Obtenido de getToken), intencion: { clave: String, frases: { {frase: String}, ... } respuestas: { {frase: String}, ... } } } </pre>
Salida	OK: <pre> { estado: String } </pre>
	KO: <pre> { estado: String, motivo_error: String } </pre>

	}
--	---

Crear intención	
Método	crearIntencion
Tipo	PUT
Entrada	<pre>{ token: String (Obtenido de getToken), intencion: { clave: String, frases: { {frase: String}, ... } respuestas: { {frase: String}, ... } } }</pre>
Salida	OK: <pre>{ estado: String, clave: String }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

Eliminar intención	
Método	eliminarIntencion
Tipo	DELETE
Entrada	<pre>{ token: String (Obtenido de getToken), intencion_id: String (Obtenido de getIntenciones) }</pre>
Salida	OK: <pre>{ estado: String }</pre>
	KO: <pre>{</pre>

```

    estado: String,
    motivo_error: String
  }

```

3.5. Usuarios

El sistema permitirá la creación, modificación, obtención y eliminación de los diferentes usuarios por medio del servicio web correspondiente.

Obtener usuarios	
Método	getUsuarios
Tipo	GET
Entrada	<pre> { token: String (Obtenido de getToken) } </pre>
Salida	OK: <pre> { estado: String, listado_usuarios: { { usuario: String, usuario_id: String, informacion: String }, ... } } </pre>
	KO: <pre> { estado: String, motivo_error: String } </pre>

Modificar usuario	
Método	modificarUsuario
Tipo	POST
Entrada	<pre> { token: String (Obtenido de getToken), usuario_id: String (Obtenido de getUsuarios), usuario: String, password: String } </pre>

Salida	OK: { estado: String }
	KO: { estado: String, motivo_error: String }

Crear usuario	
Método	crearUsuario
Tipo	PUT
Entrada	{ token: String (Obtenido de getToken), nombre_usuario: String, password: String }
Salida	OK: { estado: String, usuario_id: String }
	KO: { estado: String, motivo_error: String }

Eliminar usuario	
Método	eliminarUsuario
Tipo	DELETE
Entrada	{ token: String (Obtenido de getToken), usuario_id: String (Obtenido de getUsers) }
Salida	OK: { estado: String }
	KO: { estado: String, }

	<pre> motivo_error: String } </pre>
--	-------------------------------------

3.6. Conversaciones

El sistema permitirá descargar las conversaciones en un rango de fechas.

Descargar conversaciones	
Método	descargaConversaciones
Tipo	get
Entrada	<pre> { token: String (Obtenido de getToken), fecha_inicio: Fecha, fecha_fin: Fecha } </pre>
Salida	<pre> OK: {estado, conversacion} { estado, conversaciones: { conversacion: { conversacion_id: String, detalle: { emisor: String, receptor: String, fecha_mensaje: Fecha, mensaje: String }, ... }, ... } } </pre>
	<pre> KO: { estado: String, motivo_error: String } </pre>

Descargar conversación	
Método	descargaConversacion
Tipo	get
Entrada	<pre> { token: String (Obtenido de getToken), fecha_inicio: Fecha, </pre>

	<pre> fecha_fin: Fecha, conversacion_id: String </pre>
Salida	<pre> OK: { estado, conversacion: { { emisor: String, receptor: String, fecha_mensaje: Fecha, mensaje: String }, ... } } </pre>
	<pre> KO: { estado: String, motivo_error: String } </pre>

3.7. KPI

El sistema permitirá, por medio del servicio web correspondiente, el acceso a la información que el Chatbot gestione en tiempo real, total de chats activos.

Total de chats activos	
Método	getChatsActivos
Tipo	get
Entrada	<pre> { token: String (Obtenido de getToken) } </pre>
Salida	<pre> OK: { estado: String, num_chats_activos: entero } </pre>
	<pre> KO: { estado: String, motivo_error: String } </pre>

El sistema permitirá el acceso, por medio del servicio web correspondiente, a consultar la información de chats abandonados en un rango de fechas.

Obtener conversaciones abandonadas	
Método	getConversacionesAbandonados
Tipo	get
Entrada	<pre>{ token: String (Obtenido de getToken), fecha_inicio: Fecha, fecha_fin: Fecha }</pre>
Salida	OK: {estado, num_chats_abandonados}
	KO: <pre>{ estado: String, motivo_error: String }</pre>

El sistema permitirá el acceso, por medio del servicio web correspondiente, a consultar el tiempo medio invertido en atender las conversaciones en un rango de fechas.

Obtener tiempo medio de atención	
Método	getTiempoAtencionMedio
Tipo	get
Entrada	<pre>{ token: String (Obtenido de getToken), fecha_inicio: Fecha, fecha_fin: Fecha }</pre>
Salida	OK: <pre>{ estado: String, tiempo_segundos: entero }</pre>
	KO: <pre>{ estado: String, motivo_error: String }</pre>

El sistema permitirá el acceso, por medio del servicio web correspondiente, al *feedback* emitido por parte del ciudadano: correcto, no correcto, no contesta en un rango de fechas. Esta información será expresada en porcentaje.

Obtener feedback	
Método	getFeedback
Tipo	get
Entrada	<pre>{ token: String (Obtenido de getToken), fecha_inicio: Fecha, fecha_fin: Fecha }</pre>
Salida	<pre>OK: { estado: String, feedback: { correcto: porcentaje, no_correcto: porcentaje, no_contesta: porcentaje, } }</pre>
	<pre>KO: { estado: String, motivo_error: String }</pre>

El sistema permitirá el acceso, por medio del servicio web correspondiente, a la información sobre las intenciones consultadas en un rango de fechas. Esta información será expresada en porcentaje.

Las intenciones configuradas se obtendrán en el siguiente endpoint. La respuesta será un listado de todas las intenciones configuradas, así como sus respuestas y frases asociadas. Como detalle, tanto las frases asociadas a una intención como las respuestas, será un listado de tamaño variable.

Obtener intenciones	
Método	getIntenciones
Tipo	get
Entrada	<pre>{ token: String (Obtenido de getToken), fecha_inicio: Fecha, fecha_fin: Fecha, }</pre>
Salida	<pre>OK: { estado: String, intenciones: { {</pre>

	<pre> clave: String, frases: { {frase: String}, {frase: String}, ... }, respuestas: { {frase: String}, {frase: String}, ... } } ... } </pre>
	<pre> KO: { estado: String, motivo_error: String } </pre>

3.8. Agente

El sistema permitirá la intervención de un agente en una conversación mediante el servicio web correspondiente.

- El sistema notificará, vía webhook, sobre la necesidad de intervención de un agente en una conversación. Esta necesidad irá acompañada del resto de mensajes generados previamente entre el ciudadano y el Chatbot (conversación completa).
- El sistema enviará vía webhook los mensajes generados por parte del ciudadano.
- El sistema deberá permitir la recepción de los mensajes enviados por parte del agente, con el fin de hacerlos llegar al ciudadano.
- El sistema deberá permitir el cierre de una conversación notificado por parte del agente. Ante esta situación, el agente queda fuera de la conversación activándose nuevamente la intervención del bot.

Webhook Intervención Agente	
URL	URL Destino
Tipo	Post
Entrada	<pre> conversacion: { conversacion_id: String, detalle: { emisor: String, receptor: String, fecha_mensaje: Fecha, mensaje: String }, ... } </pre>

	<pre> }, ... } </pre>
--	-----------------------

Webhook Envío Mensajes	
URL	URL Detino
Tipo	Post
Entrada	{mensaje_usuario: String}

Recibir Mensaje de Agente	
Método	recibirMensaje
Tipo	get
Entrada	<pre> { token: String (Obtenido de getToken), conversacion_id: string, mensaje: string, } </pre>
Salida	<pre> OK: { estado: String } </pre>
	<pre> KO: { estado: String, motivo_error: String } </pre>

Cerrar Conversación	
Método	cerrarConversacion
Tipo	get
Entrada	<pre> { token: String (Obtenido de getToken), conversacion_id: string } </pre>
Salida	<pre> OK: { </pre>



	<pre>estado: String }</pre>
	<pre>KO: { estado: String, motivo_error: String }</pre>