

**Universidad de Alicante**

**Máster en Desarrollo de Aplicaciones y Servicios Web**

**Análisis de las Tendencias en Internet**

# **Práctica 1**

## ***Asistentes Conversacionales***

**José Daniel Maitán Jiménez**

**Jdmj3alu.ua.es**

**Alicante, Diciembre 2024**

## ÍNDICE DE CONTENIDOS

1. Objetivo y tareas del chatbot .....	4
2. Diseño de flujo de conversación.....	5
2.1 Descripción del Proceso .....	6
2.2. Parámetros involucrados .....	7
2.3. Ejemplo de uso .....	7
3. Servicios externos.....	7
4. Intenciones .....	10
5. Despliegue de chatbot.....	11
6. Enlaces a chatbot.....	13

## LISTA DE ILUSTRACIONES

Ilustración 1. Diagrama de flujo de funcionalidad "Ordenar pizza" .....	5
Ilustración 2. Procesamiento de intent "Order confirmation" .....	8
Ilustración 3. Procesamiento de intent "Check Order" .....	9
Ilustración 4. Bot de Telegram para conectarse a DialogFlow .....	11
Ilustración 5. Ejemplo de funcionamiento de chatbot en Telegram .....	12
Ilustración 6. Ejemplo de funcionamiento en versión web demo .....	13

## 1. Objetivo y tareas del chatbot

Para el desarrollo de esta práctica, se ha optado por la implementación de un chatbot para pedir pizza en un restaurante simulado. El objetivo del chatbot es facilitar el proceso de realizar pedidos de pizza de forma eficiente y automatizada. A través de una interfaz conversacional, los usuarios pueden explorar el menú, personalizar sus órdenes, confirmar pedidos y consultar el estado de los mismos utilizando un identificador único generado automáticamente. Este chatbot busca mejorar la experiencia del cliente al ofrecer una solución rápida, accesible y sin errores comunes asociados a los pedidos manuales.

Como tareas a realizar del chatbot se propone:

- **Interacción inicial y bienvenida:** Se dará la bienvenida al usuario y se le dará la información básica de cómo utilizar el chatbot.
- **Información:** En caso de que el usuario lo desee, el chatbot le dará toda la información sobre las tareas que puede realizar de forma detallada.
- **Consulta del menú:** Se le mostrará al usuario todas las opciones disponibles que tiene para ordenar, incluyendo tipos de pizza, tamaños y complementos.
- **Toma de pedidos:** la función principal del chatbot, donde el usuario podrá indicar su orden detalladamente (tipo de pizza, tamaño y complementos) para ser posteriormente procesada.
- **Confirmación de pedido:** el chatbot confirmará los detalles del pedido antes de finalizar. Una vez que se procese el pedido, se le devolverá al usuario un ID con su orden, la cual se guardará en una base de datos externa.
- **Consulta de pedidos existentes:** se permitirá que los usuarios consulten los detalles de un pedido previo utilizando el identificador suministrado a la hora de realizar un pedido.



Como se puede apreciar en la ilustración, se muestra el diagrama correspondiente al flujo de la conversación al utilizar la funcionalidad “Ordenar pizza”. Dicha funcionalidad permite a los usuarios realizar pedidos personalizados de pizzas a través del chatbot.

## 2.1 Descripción del Proceso

### 1. Inicio del Pedido

- El proceso comienza cuando el usuario expresa su intención de realizar un pedido, por ejemplo, diciendo: "Quiero pedir una pizza."
- Se activa el intent principal **Take Order Intent**, el cual recopila los detalles esenciales del pedido, como el tipo de pizza y el tamaño.

### 2. Selección de Ingredientes Extra (Opcional)

- Si el usuario no especifica toppings adicionales al inicio, el chatbot transiciona al intent **Ask Toppings Intent**, preguntando si desea agregar ingredientes extra.
- Según la respuesta del usuario:
  - Si proporciona toppings, estos se añaden al pedido.
  - Si dice "no" o no menciona toppings, el pedido continúa sin ingredientes adicionales.

### 3. Verificación del pedido:

- Antes de pasar a procesar el pedido, el chatbot pasa al intent **Order Verification Intent**, en el cual el usuario puede visualizar el pedido a ser procesado.
- En caso de que el usuario esté de acuerdo con lo que ve, puede indicarlo diciendo “confirmar”.
- En caso contrario, el usuario puede cancelar toda la operación.

### 4. Confirmación del Pedido

- Una vez definidos los parámetros y confirmado el pedido, el chatbot transiciona al intent **Order Confirmation Intent**, donde:
  - Se confirma el pedido.

- Se utiliza el webhook para generar un ID único y almacenar los detalles del pedido.
- El chatbot responde con un mensaje personalizado que incluye el ID del pedido, indicando que ha sido confirmado.

## 2.2. Parámetros involucrados

- **pizzaType (Requerido):** Define el tipo de pizza, como Pepperoni, Hawaiana, o Vegetariana.
- **pizzaSize (Requerido):** Define el tamaño de la pizza: Pequeña, Mediana, Grande o Extra Grande.
- **toppings (Opcional):** Lista de ingredientes adicionales que el usuario puede elegir, como champiñones, pepperoni o tocineta.

## 2.3. Ejemplo de uso

Usuario: "Quiero una pizza Hawaiana Mediana."

Bot: "¿Quieres agregar ingredientes extra?"

Usuario: "Sí, champiñones y cebolla."

Bot: "Entendido, quieres una pizza hawaiana de tamaño mediano con champiñones y cebolla. Para procesar la orden por favor di 'confirmar'".

Usuario: "confirmar."

Bot: "Tu pedido de una pizza Mediana Hawaiana con champiñones y jalapeños ha sido confirmado. Tu ID de pedido es ABCD."

## 3. Servicios externos

En esta práctica, el chatbot implementa un webhook como un servicio externo para manejar funcionalidades complejas que no pueden ser resueltas directamente dentro de Dialogflow. La integración de un webhook, le permite al chatbot ampliar sus capacidades al procesar lógica personalizada y conectarse a una "base de datos" o sistemas externos.

En este proyecto se utilizó una API a la cual Dialogflow puede enviar solicitudes con la información proporcionada por el usuario en una conversación. El webhook se encarga de procesar la información y responde con datos dinámicos que enriquecen la experiencia del usuario.

A nivel de implementación, se ha optado por un webhook desarrollado en Node.js con el framework Express.js, desplegado en la plataforma Render. (el mismo se puede encontrar en el enlace <https://mdasw-ati.onrender.com/webhook>)

Como funcionalidades específicas de este webhook implementado se tienen:

- **Confirmar pedidos y generar un ID de pedido único:** cuando el usuario confirma un pedido, el webhook genera un identificador único de 4 caracteres (ID) para la orden. Este pedido es almacenado en una “base de datos” simulada (un array de objeto en memoria) con la información sobre el tipo de pizza, tamaño y toppings seleccionados.

```
if (intent === "Order confirmation")
{
    //Se toman los parámetros pasados al chatbot
    const parameters = req.body.queryResult.parameters;
    const pizzaType = parameters.pizzaType;
    const pizzaSize = parameters.pizzaSize || "Mediana";
    const pizzaToppings = parameters.topping || [];

    //Se crea el objeto order y se guarda en la "base de datos"
    const orderId = generateOrderId();
    const order = {
        id: orderId,
        type: pizzaType,
        size: pizzaSize,
        toppings: pizzaToppings,
    };
    orders.push(order);

    const responseText = `Tu pedido de una pizza ${pizzaSize} ${pizzaType}
    ${pizzaToppings.length > 0 ? `con ${pizzaToppings.join(", ")}` : "sin ingredientes extra"}
    ha sido confirmado. Tu número de pedido es ${orderId}.`;

    return res.json({ fulfillmentText: responseText });
}
```

Ilustración 2. Procesamiento de intent "Order confirmation"



- **Consultas detalles de un pedido por ID:** el webhook permite que el usuario consulte los detalles de un pedido previamente realizado, buscando el ID proporcionado en la “base de datos”. Si el ID es válido, devuelve al usuario la información de la orden (tipo, tamaño y toppings). Si no, notifica que el pedido no fue encontrado.

```

if (intent === "Check order")
{
  const orderId = String(req.body.queryResult.parameters.orderID).toUpperCase();

  //Busca la orden en la "base de datos"
  const order = orders.find((order) => order.id === orderId);

  if (order)
  {
    const responseText = `Aquí están los detalles del pedido con ID ${orderId}:
    pizza ${order.size} ${order.type} ${order.toppings.length > 0 ? `con ${order.toppings.join(", ")}` : "sin ingredientes extra"}.`;
    return res.json({ fulfillmentText: responseText });
  }
  else
  {
    const responseText = `Lo siento, no pude encontrar ningún pedido con el ID ${orderId}. Por favor, verifica el ID e intenta de nuevo.`;
    return res.json({ fulfillmentText: responseText });
  }
}

```

*Ilustración 3. Procesamiento de intent "Check Order"*

El flujo de datos entre dialogflow y el webhook se daría de la siguiente manera:

- **Entrada:** Dialogflow envía una solicitud HTTP (en formato JSON) al webhook cuando se activa un intent que requiere lógica avanzada. Por ejemplo, el intent "Order Confirmation" envía los parámetros del pedido al webhook.
- **Procesamiento:** El webhook procesa los parámetros proporcionados por el usuario, ejecuta la lógica requerida (como generar el ID de pedido o buscar en la base de datos), y prepara una respuesta dinámica.
- **Salida:** El webhook responde con un mensaje que Dialogflow utiliza para interactuar con el usuario. Este mensaje incluye detalles personalizados, como el ID del pedido o la información completa de un pedido consultado.

Como ejemplos de interacción con el webhook se tienen:

- **Pedido de Pizza:**
  - **Usuario:** "Quiero una pizza de Pepperoni Mediana."

- **Webhook:** Genera un ID único y guarda el pedido en la "base de datos".
- **Bot:** "Tu pedido de una pizza Mediana Pepperoni ha sido confirmado. Tu número de pedido es ABCD."
- **Consulta de Pedido:**
  - **Usuario:** "Quiero consultar mi pedido con el ID ABCD."
  - **Webhook:** Busca el ID en la base de datos y devuelve los detalles del pedido.
  - **Bot:** "Aquí están los detalles de tu pedido: pizza Mediana Pepperoni sin ingredientes extra."

## 4. Intenciones

Dentro de la plataforma Dialogflow, se han definido el siguiente listado de intents para manejar el flujo de la conversación:

- **Default Fallback Intent:** Responde cuando el chatbot no entiende la intención del usuario. Ayuda a redirigir la conversación o a pedir más información.
- **Greeting:** Da la bienvenida al usuario e introduce las opciones que el chatbot puede ofrecer, como ver el menú o realizar un pedido.
- **Information:** Da información detallada sobre el chatbot, sus funciones y ejemplos de cómo utilizarlo.
- **Menu:** Proporciona una lista de las pizzas disponibles, tamaños y posibles ingredientes extra. Permite al usuario explorar las opciones antes de realizar un pedido.
- **Take Order:** Recoge los detalles iniciales del pedido, como el tipo de pizza y el tamaño. Actúa como punto de partida para el flujo de creación del pedido.
- **Ask Toppings:** Este intent se activa para preguntar al usuario si desea agregar ingredientes extra (toppings) a su pizza. Forma parte del flujo de personalización del pedido.
- **Order Verification:** Solicita una verificación del pedido antes de su confirmación definitiva, asegurándose de que todos los detalles sean correctos.
- **Order Confirmation:** Confirma el pedido del usuario, registra los datos a través del webhook y genera un ID único para el pedido.

- **Order Canceled:** Gestiona las respuestas en caso de que el usuario desee cancelar el proceso de pedido, despidiéndose y acabando la conversación de manera educada.
- **Check Order:** Permite al usuario consultar los detalles de un pedido previo utilizando el número de ID generado. Utiliza el webhook para recuperar la información almacenada.
- **Farewell:** Maneja las despedidas del usuario. Finaliza la conversación de manera educada y cortés.

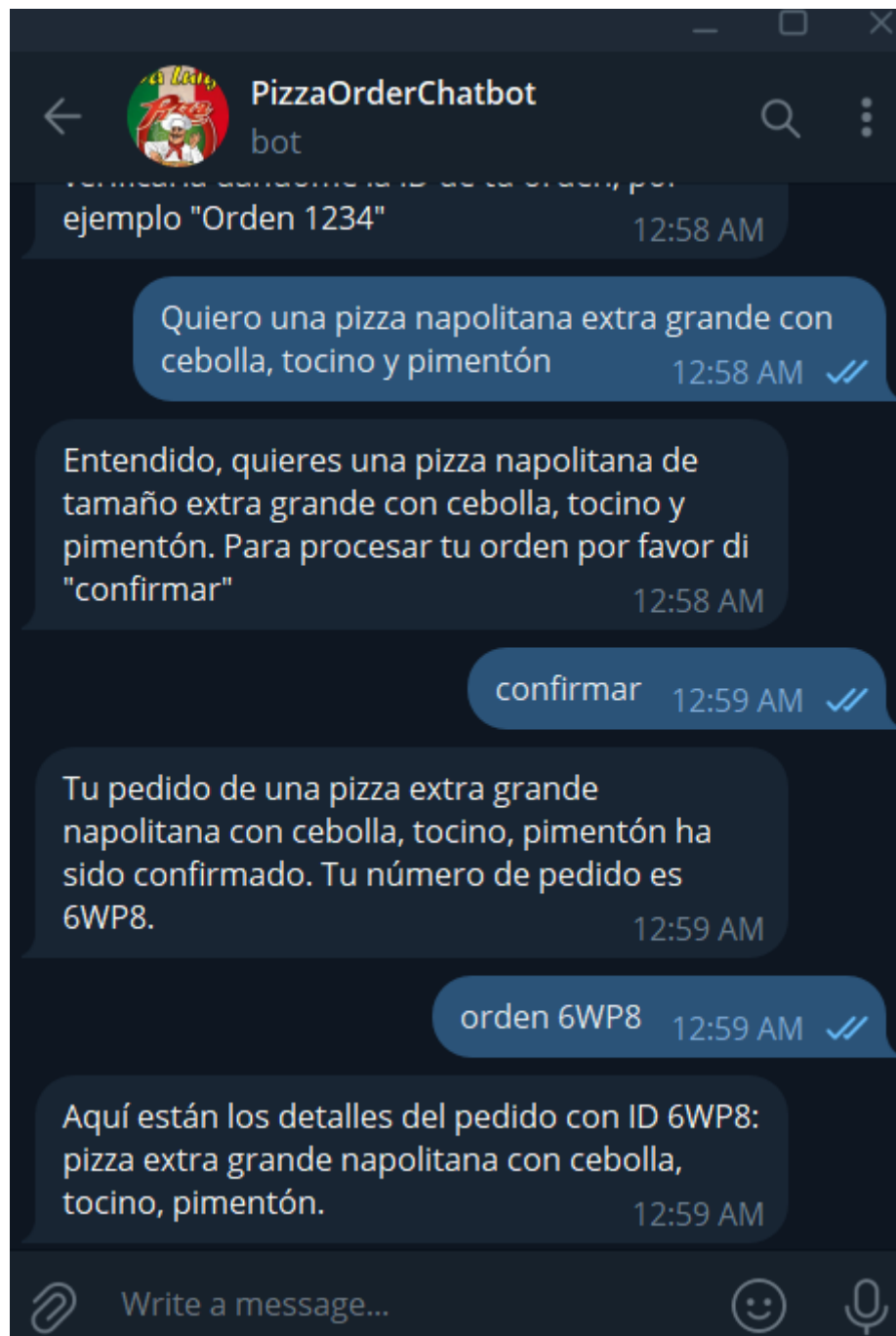
## 5. Despliegue de chatbot

A la hora de desplegar el chatbot, se han optado por 2 plataformas:

- **Bot de Telegram:** Se creó un bot con el nombre PizzarOrderChatbot. El cual es de utilidad para ver las respuestas con un espaciado estéticamente agradable.



*Ilustración 4. Bot de Telegram para conectarse a DialogFlow*



*Ilustración 5. Ejemplo de funcionamiento de chatbot en Telegram*

- **Web demo:** Para probar la integración web, se ha optado por crear un sitio estático, el cual fue subido a github y puesto en marcha con la ayuda de Render. En dicho sitio se ha colocado una ventana de chat utilizando un elemento iframe proporcionado por Dialogflow.

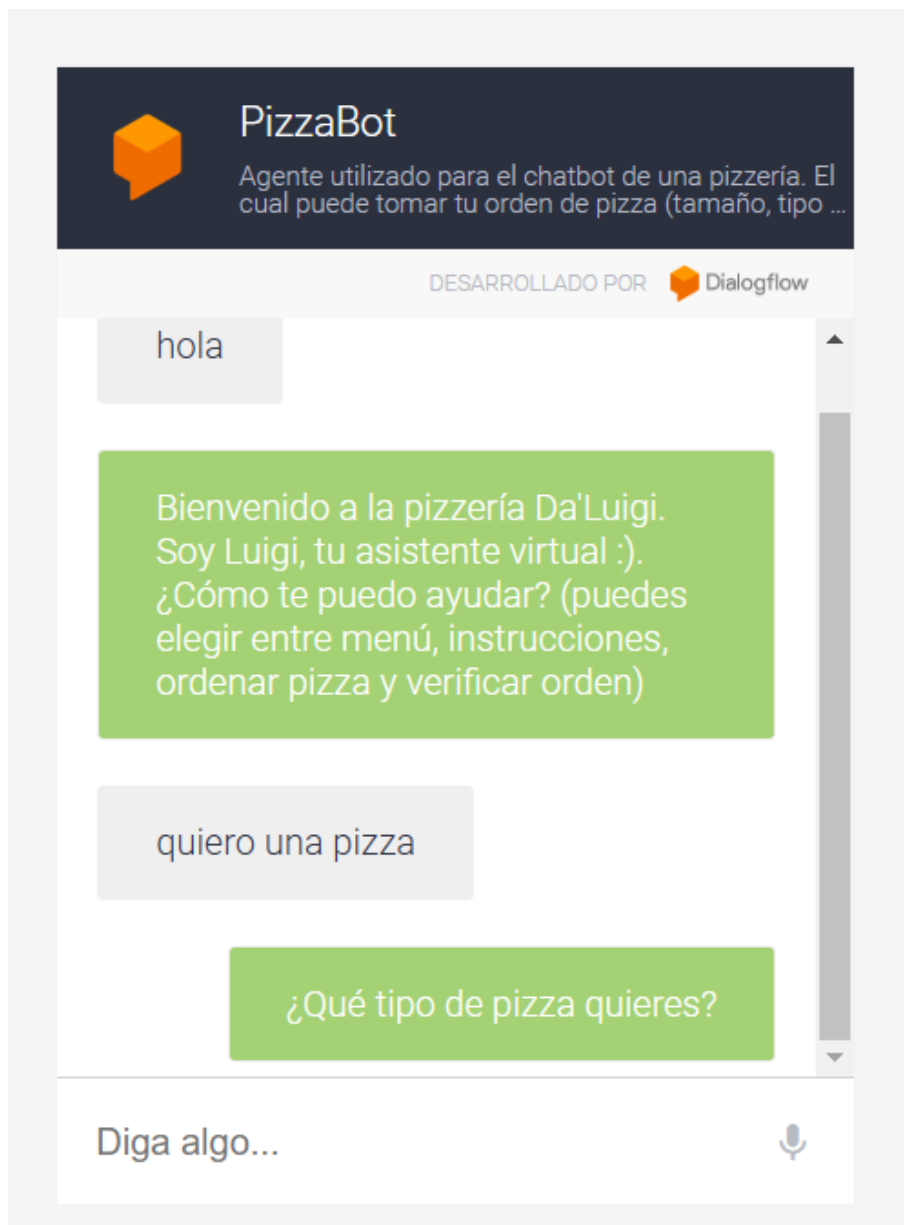


Ilustración 6. Ejemplo de funcionamiento en versión web demo

## 6. Enlaces a chatbot

- **Versión web demo:** <https://masterdasw-ati-webdemo.onrender.com/>
- **Versión de Telegram:** <https://t.me/PizzaOrderATIBot>
- **Ejemplo de funcionamiento en video:** <https://youtu.be/bAEDHF1tMM0>