Desarrollar las reglas del negocio del negociador Natalia por medio de whastapp. Con base a un flujo tipo yaml construirlo puede ser por flujo en app.diagram y convertirlo …. Pero son las reglas de negocio tipo:

graph TD

A[Inicio: Saludo + Opciones] -->|Opción A: Info Prod| B{Pregunta: Que producto?}

A -->|Opción B: Pedido| C[Pedir # Cliente]

A -->|Opción C: Rep| D[Mensaje: Transferir]

B -->|Respuesta: Laptop| E[Consultar DB Laptop]

B -->|Respuesta: Monitor| F[Consultar DB Monitor]

C -->|Cliente da #| G[Consultar DB Pedido]

E --> H[Mostrar Info Laptop + Opciones Fin]

F --> I[Mostrar Info Monitor + Opciones Fin]

G --> J[Mostrar Info Pedido + Opciones Fin]

H -->|Opción: Algo Más| A

H -->|Opción: Fin| K[Mensaje Despedida]

I -->|Opción: Algo Más| A

I -->|Opción: Fin| K

J -->|Opción: Algo Más| A

J -->|Opción: Fin| K

D --> Z[Fin Conversación]

K --> Z

Esto tiene que estar en código yamls para lectura del proceso.

# Este archivo es un ejemplo que define el flujo de la conversación.

pitches:

main\_menu: # ID del primer pitch

start\_node: bienvenida # El ID del primer nodo de este pitch

nodes:

bienvenida: # ID del Nodo "bienvenida"

type: bot\_message\_with\_options # Indica que muestra texto y botones

text: "Hola, soy tu asistente virtual. ¿En qué puedo ayudarte hoy? Por favor, elige una opción:"

options:

- text: "Quiero información sobre productos" # Texto del botón visible

value: "opt\_productos" # Valor interno que R2 enviará a R1

next\_node: productos\_q1 # ID del siguiente nodo a ir si se elige

- text: "Consultar estado de mi pedido"

value: "opt\_pedido"

next\_node: pedido\_q1

- text: "Hablar con un representante"

value: "opt\_representante"

next\_node: despedida\_agente

despedida\_agente: # ID del Nodo "despedida\_agente"

type: bot\_message # Solo muestra texto

text: "De acuerdo, te pondré en contacto con alguien."

action: transfer\_to\_agent # Una acción especial para R1

# No hay options, flujo termina o espera acción de R1

productos\_q1: # ID de otro Nodo/sub-flujo

type: bot\_message\_with\_options

text: "¿Sobre qué producto específico te gustaría saber más?"

options:

- text: "Laptop Modelo A"

value: "opt\_prod\_laptop\_a"

next\_node: productos\_show\_info # Ir al nodo que muestra info (compartido)

- text: "Monitor Modelo B"

value: "opt\_prod\_monitor\_b"

next\_node: productos\_show\_info # Ambos van al mismo nodo para mostrar info

productos\_show\_info:

type: bot\_message\_with\_action # Mostrar texto + realizar acción

text: "Consultando información del producto..."

action: lookup\_product\_info # R1 debe implementar esto

# R1 usará el valor recibido previamente (ej. "opt\_prod\_laptop\_a") para saber qué buscar.

# El texto de la respuesta final con la info DE LA DB se construirá en R1 y se enviará en el siguiente turno o se modificará este mensaje.

# Para simplificar, podemos decir que R1 prepara la info y la envía en el siguiente nodo.

next\_node: productos\_info\_mostrada # Ir al nodo después de la consulta

productos\_info\_mostrada:

type: bot\_message\_with\_options

text: "Aquí tienes la información: [RESULTADO DE LA CONSULTA DB]. ¿Algo más?"

# [RESULTADO DE LA CONSULTA DB] es un placeholder que R1 llenará dinámicamente

options:

- text: "Sí, otra cosa"

value: "opt\_algo\_mas"

next\_node: bienvenida # Volver al menú principal (conecta flujos)

- text: "No, gracias"

value: "opt\_fin"

next\_node: despedida\_final

pedido\_q1: # ID del Nodo para el pedido

type: bot\_message\_with\_options # Podría pedir # cliente

text: "Para consultar tu pedido, por favor, elige cómo quieres identificarlo:"

options:

- text: "Ingresar número de pedido"

value: "opt\_input\_pedido\_num" # R1 debe saber que después de este valor espera el número

next\_node: pedido\_wait\_number # Estado que espera el número

- text: "Ingresar # cliente"

value: "opt\_input\_cliente\_num"

next\_node: pedido\_wait\_client\_id

pedido\_wait\_number:

type: bot\_message # Este nodo no envía mensaje, es solo un estado que R1 interpreta

# R1, al ver este estado en el history, sabe que el PRÓXIMO mensaje del usuario es el número de pedido

action: wait\_for\_text\_input # R1 implementará esta acción/estado de espera

next\_node\_on\_input: pedido\_lookup\_by\_number # Si R1 recibe texto, va a este nodo

pedido\_lookup\_by\_number:

type: bot\_message\_with\_action

text: "Consultando pedido con el número proporcionado..."

action: lookup\_order\_by\_number # R1 implementará esto, usando el texto del user previo

next\_node: pedido\_show\_info

pedido\_show\_info:

type: bot\_message\_with\_options

text: "Estado del pedido: [RESULTADO DB PEDIDO]. ¿Algo más?"

options:

- text: "Sí, otra cosa"

value: "opt\_algo\_mas"

next\_node: bienvenida

- text: "No, gracias"

value: "opt\_fin"

next\_node: despedida\_final

despedida\_final: # Nodo de finalización

type: bot\_message

text: "De acuerdo. Gracias por tu consulta. ¡Hasta luego!"

action: end\_conversation # R1 debe marcar la conversación como inactiva