**Lógica Conversacional y Flujo General**

1. **Diseño del flujo lógico del chatbot (tipo árbol de decisiones)**
   * /start → Test DISC → Entrevista → Feedback

/start

↓

Mensaje de bienvenida + explicación del bot

↓

Selecciona test de perfilamiento: DISC o Hogan

↓

Test de 4–5 preguntas (según test elegido)

↓

Resultados almacenados en user\_data["perfil"]

↓

Inicio de simulación de entrevista con preguntas personalizadas

↓

Evaluación final / Feedback motivacional

↓

Fin de sesión

* + Estados a Implementar

|  |  |
| --- | --- |
| Estado | Función |
| START | Muestra bienvenida y opciones |
| TEST\_SELECT | Usuario elige DISC o Hogan |
| DISC1–DISC4 | Test DISC paso a paso |
| HOGAN1–HOGAN4 | Alternativa para test Hogan |
| ENTREVISTA | Inicio de entrevista simulada |
| CIERRE | Feedback final y despedida |

El flujo está dividido en 4 bloques: Inicio, Test DISC/Hogan, Simulación de entrevista, Cierre.

**START**

* **Nombre del estado:** START
* **Función:** Da la bienvenida al usuario y explica la finalidad del bot.
* **Transición siguiente:** TEST\_SELECT
* **Acciones:**
  + Muestra mensaje tipo:

“¡Hola! Soy *Tlabaja*, tu entrenador para entrevistas. Antes de comenzar, necesito saber un poco más sobre ti.”

* + Ofrece selección entre test DISC o HOGAN usando botones.

**TEST\_SELECT**

* **Nombre del estado:** TEST\_SELECT
* **Función:** Permite al usuario elegir entre dos test de perfilamiento: DISC o Hogan.
* **Transición siguiente:** DISC1 o HOGAN1 según selección.
* **Acciones:**
  + Guarda elección en context.user\_data["test"]
  + Muestra primera pregunta del test elegido.

**TEST DISC (4 Estados)**

Estados secuenciales donde se realizan preguntas del test DISC (4 en total).

**▪️ DISC1**

* **Pregunta:** “¿Prefieres trabajar en grupo o solo?”
* Guarda como context.user\_data["disc1"]
* → DISC2

**▪️ DISC2**

* **Pregunta:** “¿Tomas decisiones rápidamente o prefieres analizar?”
* Guarda como context.user\_data["disc2"]
* → DISC3

**▪️ DISC3**

* **Pregunta:** “¿Qué te motiva más: logros personales o ayudar al equipo?”
* Guarda como context.user\_data["disc3"]
* → DISC4

**▪️ DISC4**

* **Pregunta:** “¿Te incomodan los conflictos?”
* Guarda como context.user\_data["disc4"]
* Aquí se calcula un perfil preliminar (por ejemplo, “Colaborador”, “Analítico”) y se guarda como:
  + context.user\_data["perfil"] = "Colaborador" (esto puede ser más complejo luego)
* → ENTREVISTA

**TEST HOGAN (4 Estados)**

Alternativa si el usuario selecciona Hogan. Puedes variar las preguntas.

**▪️ HOGAN1**

* “¿Tiendes a asumir liderazgo en grupos?”
* → HOGAN2

**▪️ HOGAN2**

* “¿Eres emocionalmente reactivo ante críticas?”
* → HOGAN3

**▪️ HOGAN3**

* “¿Prefieres estructura o improvisación en el trabajo?”
* → HOGAN4

**▪️ HOGAN4**

* “¿Te cuesta desconectarte del trabajo?”
* → Guarda como DISC y pasa a ENTREVISTA

**ENTREVISTA**

* **Nombre del estado:** ENTREVISTA
* **Función:** Simula preguntas de entrevista de trabajo.
* **Acciones:**
  + Envía pregunta: "¿Cuáles son tus fortalezas?"
  + Espera respuesta.
  + Aquí se ejecuta la lógica:
    - Si la pregunta es común (existe en JSON), responde desde faq.json
    - Si no, consulta API de Gemini / OpenAI
  + Opcional: Simular cronometrado usando asyncio.sleep() entre preguntas
* **Transición:** Cierre automático luego de 2-3 preguntas → CIERRE

**CIERRE**

* **Nombre del estado:** CIERRE
* **Función:** Brinda un resumen y feedback del entrenamiento.
* **Acciones:**
  + Lee perfil generado (DISC u Hogan)
  + Muestra mensaje como:

“Gracias por participar, según tu perfil *Colaborador*, te recomendamos reforzar tus habilidades de liderazgo.”

* + Opción de reiniciar con /start o cerrar.