

Plataforma ABRIR - AI & Chatbot

Nome do Integrante	RMs
Julia Silva Spanopoulos	566754
Julia Valério Guimarães da Silva	568275
Thiago Gramorelli Lima	567630

Link para a **captura de tela** que comprova o funcionamento do bot: [AQUI](#)

- O vídeo tem 3 minutos, mas já nos primeiros segundos é possível verificar que tanto a funcionalidade de Texto quanto a de Voz funcionam.

Propósito do Chatbot

O propósito deste chatbot é atuar como um assistente virtual para a plataforma ABRIR, focado em guiar os usuários pelos desafios do futuro do trabalho. Sua função principal é desmistificar termos complexos (como IA, ODS e Gig Economy), analisar a urgência de requalificação para profissões específicas (como Analista de RH ou Telemarketing) e explicar os 5 pilares da plataforma (Acessibilidade, Base, Reskilling, Inovação e Renda). Além disso, o bot cadastra usuários para receber novidades sobre artigos, cursos e vagas, servindo como um ponto de acesso direto ao conhecimento e às oportunidades oferecidas pela ABRIR.

Relação com ODS (10 pontos)

O chatbot se relaciona diretamente com múltiplos Objetivos de Desenvolvimento Sustentável (ODS) mencionados. A sua funcionalidade central de analisar riscos de carreira (#RISCO_PROFISSAO) e focar no pilar "Reskilling" (Requalificação) está diretamente alinhada com a **ODS 4 (Educação de Qualidade)**, ao promover a aprendizagem contínua, e a **ODS 8 (Trabalho Decente e Crescimento Econômico)**, ao preparar os trabalhadores para as transformações do mercado. Adicionalmente, o pilar "Acessibilidade", que visa levar esse conhecimento a todos, explicando termos complicados, técnicos, em inglês e oferecendo a possibilidade de utilizar o bot por voz ou de forma escrita, aborda diretamente a **ODS 10 (Redução das Desigualdades)**.

Intenções e Entidades

Com base no arquivo `ABRIR-dialog.json`, aqui está a lista de intenções e entidades configuradas:

Intenções

1. #DEFINIR_TERMOS : Identifica que o usuário quer saber o significado de algum termo (Ex: "O que é IA?").
2. #RISCO_PROFISSAO : Identifica que o usuário quer saber se sua profissão está em risco (Ex: "Minha profissão vai acabar?").
3. #SOBRE_ABRIR : O usuário quer saber o que é a plataforma ABRIR (Ex: "O que vocês fazem?").
4. #NEWSLETTER : Identifica a intenção de se cadastrar para receber emails (Ex: "Quero receber novidades").
5. #INICIAR : Intenção de início de conversa e saudação, essencial para funcionamento ideal no Telegram (Ex: "Olá", "Começar").

Entidades

O bot define **4 tipos de entidades**. Para cumprir o requisito de "5 entidades", listo abaixo os 4 tipos definidos e 5 *exemplos de valores* de entidades (que são capturados por esses tipos):

Tipos de Entidade (Definidas no JSON):

1. @email (**Entidade com Regex**)
 - **Tipo:** Padrão (Regex)
 - **Regex:** \b[A-Za-z0-9._%+-]+@[A-Za-z0-9.-]+\.[A-Za-z]{2,}\b
 - **Propósito:** Capturar o email do usuário para o cadastro na newsletter.
2. @pilar_abrir
 - **Tipo:** Sinônimos (Lista de valores)
 - **Propósito:** Identificar qual dos 5 pilares da ABRIR o usuário quer conhecer.
3. @profissao
 - **Tipo:** Sinônimos (Lista de valores)
 - **Propósito:** Capturar a profissão do usuário para a análise de risco.
4. @termo_chave
 - **Tipo:** Sinônimos (Lista de valores)
 - **Propósito:** Identificar qual termo técnico o usuário deseja que seja definido.

Arquitetura de Diálogo e Design (Slots e Digressões)

Para este protótipo, a arquitetura do diálogo foi focada em criar uma **experiência de usuário fluida e flexível**, em vez de depender de uma ampla gama de entidades. Embora o sistema utilize entidades-chave para capturar informações essenciais (como

@email , @profissao e @termo_chave), o núcleo da navegação é gerenciado por recursos mais avançados do Watson Assistant:

1. **Slots (Preenchimento de Lacunas):** Em vez de tentar entender tudo de uma vez, o bot usa "Frames" (nós-pai, como #RISCO_PROFISSAO e #NEWSLETTER) que contêm **Slots**. Esses slots ativamente "caçam" a informação que precisam. Por exemplo, o nó RISCO_PROFISSAO foi projetado especificamente para ser preenchido pela variável \$risco_profissao (que usa a entidade @profissao). Isso permite que o bot cole as informações necessárias de forma contextual.
2. **Multiple Conditioned Responses (MCRs):** O bot não dá apenas uma resposta por nó. Ele usa múltiplas respostas condicionadas para reagir de forma diferente com base na informação coletada nos **Slots**. Por exemplo, dentro do Frame #RISCO_PROFISSAO , existem respostas separadas e específicas usadas de exemplo para "Telemarketing", "Designer" e "Contador", o que torna a conversa relevante.
3. **Digressions (Digressões):** Este é o recurso principal que permite ao usuário "se mover de forma mais livre". Ao configurar os "Frames" com digress_out: "allow_all" , você permite que o usuário mude de assunto no meio de uma conversa. Um usuário pode estar no fluxo de "Definir Termos" e, de repente, perguntar sobre "Risco da Profissão". O bot pode "pular" para o outro fluxo e, dependendo da configuração, retornar ao ponto original ou seguir o novo caminho.

Essa abordagem foca menos em *identificar* dezenas de entidades e mais em *gerenciar o estado* da conversa, permitindo que o usuário navegue entre os principais pilares do bot sem ficar preso em um fluxo linear.

Plano de Implementação Futura (LLM)

O plano de implementação futura reconhece os limites do protótipo atual (que é baseado em fluxos e respostas pré-definidas) e propõe uma evolução para um **modelo baseado em LLM (Large Language Model)**.

O protótipo atual é excelente em tarefas estruturadas: ele só pode definir os termos que conhece (como "IA" ou "ODS") e analisar as profissões pré-cadastradas (como "Analista de RH" ou "Motorista").

A integração de uma LLM permitiria ao bot realizar **tarefas generativas e não estruturadas**, movendo-o de um assistente informativo para um verdadeiro "coach de carreira" personalizado. Os recursos planejados incluem:

- **Análise de Perfil/CV:** Um LLM pode ler e compreender dados não estruturados, como um currículo ou portfólio em PDF, e gerar sugestões personalizadas de

reskilling ou *upskilling* com base no perfil *único* daquele usuário, em vez de respostas genéricas.

- **Prática de Entrevistas:** O bot poderia simular uma entrevista de emprego de forma dinâmica. Em vez de seguir um script, ele poderia fazer perguntas relevantes, entender as respostas do usuário em linguagem natural e fornecer feedback construtivo sobre o desempenho.
 - **Explicações Profundas e Éticas:** Em vez de apenas dar a definição de "IA", o bot poderia discutir conceitos complexos, como os usos éticos da IA, seus impactos sociais e nuances filosóficas, gerando respostas que vão muito além do que foi pré-programado.
-

Note

SPEECH TO TEXT (STT)

API Key: eWJt-u5JOrrsj_WIJ4QyBhOdTCHi37gYgvYkXL3g8EJ

URL: <https://api.us-south.speech-to-text.watson.cloud.ibm.com/instances/82e2f7ad-1018-4a7d-bd70-2aa096cb3fb5>

TEXT TO SPEECH (TTS)

API Key: D7xwaJ-Vi9fAYvWo0C0tZmawX7UEtUogVP9uah6Nj4BG

URL: <https://api.us-south.text-to-speech.watson.cloud.ibm.com/instances/fb490549-4b01-42e8-8f43-10ce7ed23359>

API Key Watsonx: WwGc9PogmdQtnfMr-9onX3FYuAhNQpGtoKBpMj6_7mSB

URL Watsonx: <https://api.us-south.assistant.watson.cloud.ibm.com/instances/ba19b410-ba20-4d13-b2f0-5a5634e5e685>

Assistant ID: 876da074-1782-4f10-9c96-7a2f270ee680

Token do Bot do

Telegram: 8467314389:AAGVxHMtjiWZE0HnzcyKtOiMIRy1IQdPNR4

Link do Bot no Telegram: t.me/FIAP_ABRIRbot