

# TechLab Agentes

<b>Introdução.....</b>	<b>2</b>
<b>O Desafio.....</b>	<b>2</b>
Contexto Geral.....	2
Escopo do Projeto.....	2
Capacidades do Agente.....	3
Requisitos Técnicos.....	3
Tecnologias Sugeridas.....	5
<b>Informações Gerais e Recursos.....</b>	<b>5</b>
<b>Entrega do Projeto.....</b>	<b>6</b>
<b>Critérios e Avaliação.....</b>	<b>7</b>

## Introdução

**Início do Desafio: 25/06**

**Data de Entrega: 07/07**

O objetivo deste projeto é desenvolver um agente conversacional que facilite a integração de novos funcionários na Tech4ai, ajudando-os a se familiarizar rapidamente com a cultura, políticas, programas e ferramentas de trabalho da empresa.

Este documento descreve a proposta de trabalho do projeto a ser desenvolvido, define requisitos, sugere ferramentas, fornece informações sobre materiais que todos usarão, sobre entrega e avaliação.

## O Desafio

### Contexto Geral

A Tech4ai está buscando soluções inovadoras para melhorar o processo de onboarding de novos funcionários. Um agente conversacional pode oferecer suporte contínuo e personalizado, auxiliando na adaptação dos novos membros à empresa.

### Escopo do Projeto

O agente deve ser capaz de:

- Responder a perguntas frequentes sobre a empresa.
- Fornecer tutoriais detalhados sobre ferramentas internas.
- Agendar reuniões de boas-vindas.

## Capacidades do Agente

### 1. Responder Perguntas frequentes

O agente deve responder a perguntas comuns sobre a empresa, tais como missão, visão, valores, cultura, programas internos, políticas de trabalho remoto, horários, etc.

Com base em uma documentação que será fornecida.

### 2. Fornecer Tutoriais sobre Ferramentas Internas

O agente deve fornecer tutoriais passo a passo sobre como usar as ferramentas internas da empresa, como plataformas de comunicação e sistemas de gerenciamento de projetos.

A partir de acesso à internet, deverá fornecer tutoriais de acesso, uso e/ou instalação das seguintes ferramentas: Github, Vscod, Jira e Discord.

### 3. Agendar Reuniões de Boas-vindas

O agente deve integrar-se com sistemas de calendário (ex: Google Calendar) para agendar reuniões automaticamente, gerenciando autenticação, APIs externas e conflitos de agenda.

## Requisitos Técnicos

### 1. Gestão de Contexto

- O agente deve ser capaz de guardar as mensagens do usuário e gerenciar o contexto ao longo da conversa.
- Deve ser capaz de diferenciar usuários de alguma forma.
- Deve conversar de forma livre com o usuário dentro do escopo definido.

### 2. Bases de Conhecimento

- Devem haver bases de conhecimento para a documentação fornecida pela Tech (RAG).
- O agente deve conseguir acessar e buscar contexto nas bases para melhor responder ao usuário.

### 3. Ferramentas

- a. O agente deve implementar pelo menos três ferramentas específicas que ajudem na resolução dos desafios propostos.
- b. Essas ferramentas devem ser integradas e configuradas para funcionar de forma eficiente dentro do escopo do projeto.

### 4. Restrições e limitações

- a. O agente não deve falar sobre outras empresas.
- b. Não deve fornecer informações pessoais.
- c. Deve inibir discurso de ódio.
- d. Não pode aceitar requisições maliciosas (ex: injeção de prompt).

### 5. Interface de acesso

- a. O agente deve ser acessível através de uma API.
- b. A interface deve ser fácil de usar e intuitiva (ex: Swagger).
- c. Uma frontend simples é preferível, porém via terminal é aceitável.

### 6. OpenSource

- a. Apenas soluções open source e gratuitas devem ser utilizadas no desenvolvimento do projeto.
- b. O uso de ferramentas e bibliotecas proprietárias ou pagas é estritamente proibido.

### 7. LLM

- a. O modelo de linguagem a ser usado como motor do agente deve ser o `llama3-70b-8192`.
- b. Deverá ser usada a API de acesso aos modelos fornecida pela **Groq**, acessível em [console.groq](https://console.groq.com).  
Os modelos são open e o nível gratuito é suficiente.
- c. Outros modelos fornecidos pela plataforma podem ser usados, contanto que não seja o motor principal de inferência.

### 8. Linguagem

- a. Deve ser usada uma Stack em python para o projeto todo.
- b. Obrigatoriamente o backend deve ser totalmente desenvolvido em python.

- c. O frontend pode ser desenvolvido em outra linguagem caso seja mais cômodo.

## Tecnologias Sugeridas

- **Frameworks de Agentes:** [Langchain](#), [LLamaIndex](#), [CrewAi](#), [Autogen](#), [memGPT](#), etc.
- **RAG:** [FAISS](#), [ChromaDb](#)
  - Geralmente são usados via conectores de frameworks (e. g., [faiss via langchain](#))
- **Banco de Dados (Se necessário):** Livres para escolha do candidato (e.g., [SQLite](#), [PostgreSQL](#), [MongoDB](#), etc.).
- **API:** [FastAPI](#), [Flask](#).
- **Documentação de API:** Swagger ou equivalente.
- **Frontend Simples em Python:** [Streamlit](#), [Dash](#), [Panel](#), [Gradio](#), etc.

## Informações Gerais e Recursos

### 1. Datas:

- a. Início do Desafio: 25/06
- b. Data de Entrega: 07/07
- c. Período de Análise: 08 e 09/07
- d. Datas de Apresentação: 10 e 11/07
- e. Divulgação do Resultado: 12/07

### 2. Canal de Comunicação:

- a. Grupo no Whatsapp com os candidatos e o time IAG.
- b. A finalidade do grupo é facilitar a comunicação.
- c. Dúvidas gerais podem e devem ser sanadas por este meio.
- d. Todas as informações serão disponibilizadas por este canal.

### 3. Documentação e Bases de Conhecimento:

- a. Documento da Proposta (Este documento).
- b. Documentos para Bases de Conhecimento:
  - i. Estrutura organizacional, cultura e programas.

### 4. Mentorias e Suporte

- a. **Local:** Sala do Coworking ou Salas de Reuniões na INCIT.
- b. Agendamentos de 15 minutos em um painel de disponibilidade.
- c. Deve ser feita pelo menos **uma mentoria obrigatória**.
- d. Caso veja a necessidade de utilizar mais de 15 minutos, marque 2 mentorias em sequência (máximo 2 em sequência).
- e. O link para agendamento é o seguinte: [Calendar](#).

## Entrega do Projeto

### 1. Formato de Entrega:

- a. Um repositório no GitHub deve ser criado para o projeto, onde todo o código e a documentação serão armazenados.
- b. O projeto deve ser devidamente documentado, incluindo todas as instruções necessárias para instalação, configuração e uso. Estas instruções devem estar contidas no README.MD.
- c. O link do repositório deverá ser enviado via formulário.
- d. O repositório deverá ser público após o envio e não deve ser mais atualizado.

### 2. Apresentação

- a. Será feita uma apresentação do projeto de 10 minutos nos dias 10 e 11 de julho.
- b. A apresentação deve seguir o padrão:
  - i. Apresentação pessoal.
  - ii. Demonstração do resultado.
  - iii. Explicação das etapas de desenvolvimento como quiser.

## **Cr terios e Avalia  o**

### **1. Avalia  o**

- a. Vamos avaliar todos os  mbitos do projeto com crit rios objetivos.
- b. A classifica  o final ser  divulgada no dia 12/07.
- c. As m tricas de avalia  o e notas poder o ser auditadas caso necess rio.
- d. Usaremos o formul rio de inscri  o como m todo de desempate.

### **2. Crit rios**

- a. Qualidade de C digo e Desempenho
- b. Adequa  o ao Escopo
- c. Inova  o
- d. Apresenta  o
- e. Criatividade e Autonomia
- f. Documenta  o e Organiza  o