

Secretaria Virtual - Projeto Multiclínicas

Visão Geral

Este repositório contém a estrutura da **Secretaria Virtual Inteligente**, criada para atender múltiplas clínicas odontológicas e médicas, com foco em escalabilidade, personalização por unidade e automação via IA (OpenAI), Google Sheets e WhatsApp (Twilio).

Estrutura de Pastas

```
├── app/ | ├── agentes/ # Agente LangChain | ├── config/ #  
Identidade da clínica e secretária | ├── modelos/ # Tabelas do  
banco (SQLAlchemy) | ├── tools/ # Tools por especialidade, FAQ  
etc. | ├── utilitarios/ # Funções auxiliares (Twilio, GPT etc.) |  
├── rotas/ # Rota WhatsApp | └── testes/ ├── configs/ # Arquivos  
.json gerados por clínica ├── migracoes/ # Alembic ├──  
prompts_secretaria/ # Prompts por intenção ├── executar.py #  
Entrypoint do projeto ├── gerar_todos_jsons.py # Gera JSONs para  
todas clínicas ├── salvar_json_config.py # Salva arquivo .json  
individual ├── .env.example # Exemplo de variáveis ├──  
requirements.txt └── README_Secretaria_Virtual.md
```

Banco de Dados

Tabelas principais

- **clinicas**: id, nome, cidade, *nomesecretaria*, *contextocomunicacao*
- **leads**: campos de triagem inteligente, status e dados clínicos
- **profissionais**: nome, especialidade
- **especialidades**: palavras-chave, responsável clínico

- **perguntas_frequentes**: base de conhecimento (FAQ)
 - **contexto**: histórico da conversa com cada lead
 - **historico_conversas**: logs brutos de interação
-

Geração de JSONs por Google Sheets

Como funciona?

1. As clínicas preenchem um **Google Forms** que alimenta uma planilha com duas abas: `clinicas` e `especialidades`.
 2. O script `gerar_todos_jsons.py` acessa a planilha via API do Google e gera um `.json` de configuração por clínica.
 3. Esses arquivos ficam salvos em `/configs`, prontos para uso pela IA.
-

Deploy e Segurança

- O arquivo `credenciais_google.json` com a service account **NÃO DEVE SER COMMITADO**.
 - Adicione ao seu `.gitignore`: `credenciais_google.json .env *.swp __pycache__/ *.pyc`
 - Use `.env.example` como modelo para suas variáveis de ambiente.
-

Tecnologias Utilizadas

- Python 3.10+
 - Flask
 - SQLAlchemy + Alembic
 - OpenAI (Chat + Whisper)
 - Twilio API
 - Google Sheets API
 - LangChain
-

Links Úteis

- [Documentação Google Sheets API](#)

- [LangChain](#)
 - [OpenAI API](#)
 - [Twilio WhatsApp API](#)
-

Clínicas Atendidas

Os arquivos `.json` da pasta `configs/` representam cada clínica ativa no projeto, com personalização de:

- Nome da secretária IA
- Estilo de comunicação
- Profissionais e especialidades
- Público-alvo (crianças, TEA, adultos etc.)