SUPORTE ORIENTADO POR HEURÍSTICA COM IA – HELPDESK LEVEL 1

## Escrito por: Éder da Silva Paiva Vasconcelos Gonçalves

## **Para que serve esse projeto? O que ele faz?**

**Este projeto experimental serve como um assistente inteligente para equipes de suporte técnico. Através da IA Generativa, ele analisa tickets para fornecer informações cruciais para a tomada de decisão, como a prioridade estratégica, a correção da categorização, o tipo de requisição e o grau de dificuldade. Além disso, sugere ações, manuais técnicos e respostas iniciais, empoderando os analistas humanos e simplificando processos complexos.**

## **O que o motivou para criar esse projeto?**

**A minha motivação surgiu da vontade de explorar o potencial da Inteligência Artificial Generativa para resolver problemas reais e complexos do mundo corporativo. O suporte técnico, com seu grande volume de dados não estruturados e a necessidade de decisões rápidas e precisas, me pareceu um campo ideal para aplicar essa tecnologia. Quis demonstrar como a IA pode ir além de tarefas básicas e atuar como um verdadeiro "cérebro auxiliar", capaz de extrair insights valiosos de dados complexos e fornecer suporte inteligente à tomada de decisão humana em um contexto desafiador como o atendimento técnico.**

## **Qual ferramenta ou ferramentas de IA usou?**

**API GEMINI, EASYOCR, GTTS**

## **Qual foi o passo a passo que você usou para esse projeto?**

**O projeto nasceu da experiência prévia com Inteligência Artificial (dois projetos experimentais que realizei no último ano) e do desejo de explorar novas aplicações para a IA Generativa, especialmente em cenários complexos que demandam apoio à decisão humana. A motivação central foi auxiliar as equipes de suporte técnico, frequentemente sobrecarregadas com a análise de um grande volume de tickets, muitos deles contendo dados não padronizados e desafiadores de interpretar. Partindo dessa ideia inicial, o primeiro passo concreto foi conectar o sistema em Python, linguagem escolhida para o projeto, à base de dados do sistema de Helpdesk utilizado, permitindo o acesso automatizado aos tickets e suas informações. Essa etapa envolveu pesquisa e implementação da conexão, com auxílio da própria IA Gemini para identificar as bibliotecas Python mais adequadas, complementando com documentações online.**

**Um desafio crucial logo se apresentou: a necessidade de lidar com os anexos dos tickets, que frequentemente incluem imagens (como prints de tela com logs de erro) e documentos PDF (manuais técnicos extensos ou relatórios). Para extrair informações relevantes desses anexos, foram implementadas soluções específicas. Inicialmente, testou-se usar a própria API Gemini para OCR (Reconhecimento Óptico de Caracteres) em imagens e extração de texto de PDFs, mas logo se percebeu a limitação no número de requisições à API com a chave gratuita. Para otimizar o processo e economizar recursos da API, optou-se por integrar a biblioteca EasyOCR para realizar o OCR localmente nas imagens, mostrando-se uma solução eficaz e mais eficiente em termos de custo. Para PDFs, foi implementada uma função de conversão para texto, garantindo que o conteúdo textual desses documentos também pudesse ser analisado pela IA.**

**Superado o desafio dos anexos, o projeto avançou para a integração de manuais técnicos como fonte de conhecimento adicional para a IA. A ideia era que, ao fornecer manuais relevantes, a IA pudesse realizar análises mais precisas e contextuais. Contudo, o limite de dados da API Gemini impôs uma restrição: não era viável enviar todos os manuais técnicos de uma só vez. A solução inovadora encontrada foi criar um mecanismo para que a própria IA fosse capaz de selecionar o manual técnico mais apropriado para cada ticket. Para isso, foi desenvolvido um algoritmo onde a IA, após analisar o contexto geral do ticket, escolhe o manual mais relevante de uma lista predefinida, baseando-se em descrições resumidas dos manuais fornecidas ao sistema. Essa estratégia permitiu direcionar o foco da IA para informações mais pertinentes, otimizando o uso da API e melhorando a qualidade da análise.**

**Com os dados dos tickets, anexos convertidos em texto e, quando aplicável, o manual técnico selecionado, o passo seguinte foi criar o algoritmo central de análise com a API Gemini. Este algoritmo foi cuidadosamente estruturado para dividir a análise em etapas, visando otimizar as consultas à API e garantir que a IA recebesse todas as informações necessárias de forma organizada. O fluxo principal envolve a obtenção dos dados do ticket, o processamento dos anexos, a seleção do manual técnico pela IA, e, finalmente, o envio de todas essas informações, juntamente com uma "instrução complexa", para a API Gemini. Essa instrução detalha as análises desejadas, como priorização, categorização, identificação do tipo de requisição, sugestão de soluções e respostas iniciais, entre outros. Para a análise da fila de tickets como um todo, foi criada uma função separada, com foco na priorização e na avaliação da complexidade dos chamados, utilizando uma escala baseada na sequência de Fibonacci para graduar a dificuldade percebida pela IA.**

**Finalmente, para apresentar os resultados de forma clara e interativa, foi utilizada a biblioteca Streamlit para construir uma interface web simples. Essa interface permite visualizar as análises da IA diretamente na tela e também facilita o envio de relatórios completos por e-mail, nos formatos PDF, áudio (gerado com a biblioteca gTTS) e CSV, consolidando todas as informações relevantes para a equipe de suporte.**

## **O que você achou do resultado do seu projeto?**

**O resultado me parece promissor e acredito que a técnica pode ser adaptada para o uso da IA como “suporte” à análise e tomada de decisão também em outros departamentos da organização (Fiscal, RH, Compras, etc). No entanto, é claro que esta versão experimental precisa de muitas melhorias e eu gostaria de adicionar novas funcionalidades à ferramenta; como a consulta de tickets finalizados para obtenção de soluções pela IA a partir da própria base de dados; outra função interessante é usar a IA para a criação de um FAQ padronizado que seja útil não só para os analistas como também para os próprios usuários de sistemas. Além disso, o projeto baseou-se no uso de uma tecnologia experimental do google, o que torna a necessidade da revisão humana sobre o resultado da análise realizada pela IA de extra importância****, visto que a resposta da IA nem sempre está 100% correta: em alguns dos meus testes, a IA respondeu em inglês, errou no código do módulo do ERP (ao invés de 02, informou que era o módulo 27, por exemplo), etc. Essas imprecisões na resposta podem ser melhoradas com o tempo, com mais informação passada para a IA de forma estruturada e isso é algo que também precisa ser melhorado nesse projeto e esse foi justamente o desafio que queria testar com o uso da IA Generativa: avaliar seu desempenho em uma área complexo e com muitas “variáveis” diferentes a serem consideradas. Outra possibilidade bastante interessante para esse tipo de projeto é o treinamento da IA Generativa usando LangChain e dados estruturados para aperfeiçoar a resposta da IA diante de cenários complexos. Outro teoria bastante interessante é combinar o uso de IA Generativa com RPA (Robotização de Processos Automatizados), no entanto, este é um campo sobre o qual é preciso bastante cautela, visto que a IA Generativa apresenta um comportamento que não é 100% à prova de falha e usá-la para decidir o fluxo de um processo autônomo envolve algum risco.**