TP 4 - Agentes com LLMs

O objetivo deste trabalho é capacitar os alunos a conceituar, projetar, implementar e documentar uma solução baseada em agentes autônomos que utilizam Modelos de Linguagem (LLMs). Espera-se que os alunos demonstrem a habilidade de orquestrar um LLM com ferramentas externas e bases de conhecimento para resolver um problema complexo e bem definido.

Os alunos deverão desenvolver um projeto completo, dividido nas seguintes etapas:

Etapa 1: Proposição e Justificativa do Problema Escolha um problema do mundo real que possa ser eficientemente resolvido por um sistema de agentes com LLM. O problema deve ser específico e bem delimitado.

- **Requisito fundamental:** A solução para o problema proposto **obrigatoriamente** deve requerer o uso de:
    1. **Recuperação de Informação (RAG - Retrieval-Augmented Generation):** O agente deve ser capaz de consultar uma base de conhecimento privada (ex: arquivos PDF, TXT, CSV, etc.) para responder a perguntas ou guiar suas ações.
    2. **Pelo menos uma (1) ferramenta adicional:** Além da busca na base de conhecimento, o agente deve ser capaz de usar outra ferramenta, como, por exemplo:
        - Acesso a uma API pública (ex: previsão do tempo, cotação de ações, notícias).
        - Um interpretador de código Python para realizar cálculos ou análises.
        - Uma ferramenta de busca na web (ex: Google Search, DuckDuckGo).
        - Uma ferramenta para consultar um banco de dados SQL.
- **Justificativa:** Você deve descrever claramente o problema e argumentar por que uma arquitetura de agentes com as ferramentas escolhidas é a abordagem mais adequada para resolvê-lo.

Etapa 2: Desenho da Arquitetura e Fluxo Lógico Antes da implementação, você deve projetar a solução.

- **Fluxograma:** Crie um fluxograma detalhado que ilustre a arquitetura do seu sistema. O fluxograma deve mostrar:
    - O agente principal (orquestrador).
    - As ferramentas disponíveis (RAG, ferramenta adicional).
    - O fluxo de uma requisição do usuário.
    - Os pontos de decisão onde o agente escolhe qual ferramenta usar (ou se não precisa de nenhuma).
    - Como a informação flui entre o usuário, o agente e as ferramentas.
- **Descrição Lógica:** Acompanhando o fluxograma, descreva a lógica de funcionamento, incluindo os *prompts* principais que você planeja usar para instruir o agente a raciocinar e a tomar decisões.

Etapa 3: Implementação e Entrega Técnica Implemente a solução proposta.

- **Ambiente de Execução:** O projeto deve ser entregue como um notebook que possa ser executado no **Google Colab**.
- **Alternativa Local:** Caso não seja possível usar o Colab, o projeto deve ser entregue em um repositório ou arquivo compactado contendo:
    - Todo o código-fonte.
    - Um arquivo de configuração de ambiente (`requirements.txt` para pip ou `environment.yml` para Conda) que permita recriar o ambiente de execução de forma trivial.
- **Boas Práticas:** O código deve ser bem comentado. Evite expor chaves de API diretamente no código.

Entregáveis

1. Um **documento único (PDF)** contendo:
    - Nomes dos integrantes do grupo.
    - Descrição e justificativa do problema.
    - O fluxograma e a descrição da arquitetura.
2. O **notebook (`.ipynb`)** funcional ou a pasta compactada com o projeto local e seu arquivo de configuração.