

1º Relatório - Extensão III - 17/10/2025

Equipe: José Fortunato, Gabriel Marques, Larissa Kelly, Marina Carvalho, Lyedson Silva.

Projeto: ExamForge

Definição do Projeto – Dificuldades e Decisões

No início do projeto, nós da equipe pensamos em desenvolver uma plataforma web, na qual o usuário pudesse realizar seu cadastro e gerar simulados com base em materiais de estudo que ele fornecesse separados por áreas do conhecimento. A ideia era que a LLM interpretasse o conteúdo e desenvolvesse simulados personalizados, contendo questões objetivas e subjetivas, de acordo com o perfil e interesse do usuário, permitindo que o usuário salvasse e organizasse seus simulados dentro da plataforma por disciplinas, podendo acessar, editar e refazer esses simulados posteriormente.

Essa proposta inicial visava oferecer uma experiência ampla e flexível, integrando diversas funcionalidades relacionadas ao estudo autônomo. Entretanto, durante as reuniões internas e os primeiros estudos técnicos, a equipe identificou diversas dificuldades e limitações importantes para o desenvolvimento desta proposta, tais como:

- Escopo muito amplo, inviável para o tempo de desenvolvimento disponível na disciplina e para a capacidade de entrega do grupo.
- Concorrência com ferramentas já consolidadas, como o ChatGPT, que já oferecem geração de questões e textos de forma mais robusta.
- Desafio técnico de criar um modelo próprio capaz de gerar diferentes quantidades e tipos de questões sem alucinações, além de integrá-las ao front-end de maneira dinâmica, uma vez que não haveria padronização nas telas.
- Ausência de um público-alvo bem definido, já que não estava claro se a aplicação seria voltada para alunos, professores ou o público em geral.

Link Slide da primeira proposta:

https://www.canva.com/design/DAG0Bf_C9Io/kUC_o9wAYJdTu3opYj8W-Q/edit?utm_content=DAG0Bf_C9Io&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Diante dessas dificuldades, após novas discussões e alinhamentos, nos da equipe decidimos repensar o escopo do projeto com o objetivo de torná-lo mais objetivo, funcional e condizente com o tempo disponível. Assim, chegamos à proposta atual: Uma plataforma web que não necessita de cadastro, e tem como foco principal a geração de simulados de múltipla escolha baseados em materiais fornecidos pelo usuário, permitindo o usuário exportar como pdf para estudo.

Nessa nova proposta, o usuário pode:

- Fazer o upload de seus materiais de estudo;

- Definir a quantidade de questões, o tempo de realização e se haverá múltiplas alternativas corretas;
- Realizar o simulado diretamente na plataforma;
- Receber o resultado com correção automática e explicações detalhadas das respostas;
- E, ao final, exportar o simulado em PDF para uso posterior ou anotações.

Essa reformulação tornou o projeto mais enxuto, viável e direcionado, mantendo a essência da proposta inicial, que seria o uso de IA como apoio ao aprendizado , mas com uma implementação mais realista e funcional dentro do escopo da disciplina.

Link do slide da reformulação da proposta:

https://www.canva.com/design/DAG19ofRBS0/uR_t3BPNzSR5LLxOhLO0Hw/edit?utm_content=DAG19ofRBS0&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton

Link Figma:

<https://www.figma.com/design/HSRTCvgfCwPV4nz6m47Ei3/ExamForge?node-id=319-116&p=f&t=W0iPoSIE7fTaWJAQ-0>

LLMs, RAG e Engenharia de Prompt

Como a maior parte dos membros da equipe ainda não possuía familiaridade com a implementação de modelos de linguagem (LLMs) ou aplicações que fizessem a sua utilização, as primeiras semanas do projeto foram dedicadas a estudos teóricos e práticos sobre o funcionamento dessas ferramentas, incluindo RAG (Retrieval-Augmented Generation) e técnicas de fine-tuning e engenharia de prompt.

Os primeiros testes foram realizados com um modelo sem *system prompt* bem definido. Solicitamos que o modelo gerasse cinco questões de múltipla escolha sobre um tema definido. O resultado foi o esperado: o modelo alucinou, retornando respostas fora do escopo solicitado e, em alguns casos, respostas em inglês, demonstrando a necessidade de maior controle no contexto da geração

Nos testes seguintes passamos a realizar com prompts mais elaborados, mas ainda sem o uso de RAG. O *prompt* já incluía uma persona para o modelo, que foi definida como: “Assistente virtual com foco em criação de questões de múltipla escolha”, seguido de um exemplo de estrutura de questões, servindo como guia de formatação. Os resultados foram mais satisfatórios, já era seguido o padrão estabelecido. O modelo passou a retornar o número correto de questões, todas estruturadas e coerentes com o padrão estabelecido.

Em seguida, o mesmo modelo foi testado com o uso de RAG, permitindo requisições mais específicas a partir de arquivos reais (PDFs de disciplinas). Os retornos iniciais foram satisfatórios, com as questões sendo geradas de forma consistente e sem alucinações. Em síntese, essas foram as decisões mais importantes que tivemos nessa fase:

- Os *system prompts* devem ser bem definidos e objetivos, contendo exemplos de estrutura esperada das questões.

- O uso de RAG será adotado como estratégia principal para gerar questões baseadas no conteúdo enviado, por atender de forma eficaz às necessidades do projeto.

Ainda há alguns pontos a serem discutidos, como a escolha definitiva do modelo de linguagem mais adequado para o nosso objetivo, qual base de dados é a melhor opção e se existe alguma maneira de melhorar o prompt, mas a base principal do projeto já está definida.

Próximos Passos:

1. Dar continuidade ao desenvolvimento do projeto, a partir do protótipo apresentado em sala, verificando a viabilidade de implementar as funcionalidades de:
 - 1.1. Geração automática de questões de múltipla escolha;
 - 1.2. Controle de tempo (temporizador) durante os simulados;
 - 1.3. Envio e validação de arquivos pelo usuário;
 - 1.4. Exportação do simulado em formato PDF.
2. Estudar a viabilidade de novas funcionalidades, como a inclusão de questões subjetivas e variações no formato das questões de múltipla escolha.
3. Explorar o uso da FastAPI c, visando facilitar a integração entre o frontend e os serviços de geração de questões via LLM.
4. Analisar e propor melhorias no sistema de correção, tornando-o mais preciso e ajustado às respostas fornecidas pelo usuário.
5. Iniciar o desenvolvimento do frontend, definindo a identidade visual e o design da interface, com base no protótipo construído no Figma.
6. Finalizar a documentação inicial do projeto, consolidando os requisitos e elaborando o diagrama de sequência representando o fluxo geral do sistema.