



Projeto Final – "Qualifica"

Versão sem sistema de vidas

O que sua equipe vai construir

Vocês vão criar um programa em Python que simula um aplicativo de aprendizagem profissional — parecido com o Duolingo, mas focado em habilidades como comunicação, lógica, tecnologia e organização no trabalho. Tudo será feito no computador, sem internet, usando apenas Python e o banco de dados SQLite3.

O sistema terá três tipos de usuários, cada um com funções diferentes:

- **Aluno:** faz atividades, ganha estrelas e acompanha seu progresso.
 - **Professor:** adiciona cursos e atividades pelo próprio programa.
 - **Coordenador:** vê relatórios gerais (ex: ranking, desempenho da turma).
-



Funcionalidades por tipo de usuário



1. Login com tipo de usuário

- Ao iniciar, o usuário escolhe: "Aluno", "Professor" ou "Coordenador".
- Depois digita nome e senha.
- Se for um nome novo, o sistema pergunta o tipo e cadastra.
- Cada tipo tem acesso a um menu diferente.



2. Funcionalidades do ALUNO

- Ver lista de cursos/módulos disponíveis
- Fazer atividades variadas:
 - Múltipla escolha
 - Verdadeiro ou Falso
 - Preencher lacunas
 - Ordenar etapas
 - Sequências lógicas
 - Correspondência (ligar colunas)
 - Classificação (separar categorias)
 - Escolha múltipla (várias corretas)
 - Palavra embaralhada
 - Mini-cenários com decisões
- Pedir dica antes de responder
- **Ganhar 1 estrela por acerto** *(sem penalidade por erro)*

- Ver seu progresso: estrelas acumuladas, atividades feitas hoje
- Resetar seu próprio progresso
- Ver o ranking dos 3 melhores alunos



3. Funcionalidades do PROFESSOR

- Adicionar novo curso/módulo (ex: "Gestão de Tempo")
- Adicionar nova atividade (escolhe o tipo, escreve pergunta, opções, resposta correta e dica)
- Ver a lista de cursos e atividades cadastradas
- Não vê estrelas nem ranking — só gerencia conteúdo
- Tudo é salvo no mesmo banco de dados, sem precisar editar código!



4. Funcionalidades do COORDENADOR

- Ver ranking completo da turma (todos os alunos, com estrelas e acertos)
- Ver quantas atividades foram feitas hoje por todos
- Ver desempenho por curso (ex: "80% acertaram o módulo de Comunicação")
- Não faz atividades nem edita conteúdo — só acompanha



5. Personalização extra: arquivo atividades_extra.txt

- Além do modo professor, o sistema lê um arquivo chamado atividades_extra.txt ao iniciar.
- Se houver novas atividades lá, ele adiciona automaticamente no banco.
- Isso permite ao professor preparar atividades com antecedência, sem usar o programa.



Como organizar o trabalho

Use funções com def para cada tarefa (ex: menu_aluno(), adicionar_curso(), ver_ranking()).

Separem o código em arquivos:

- main.py → menu principal, login e redirecionamento por tipo de usuário
- banco.py → conexão, criação de tabelas, leitura do atividades_extra.txt
- atividades.py → tipos de atividade, correção, dicas
- usuarios.py → cadastro, verificação de tipo, login
- progresso.py → estrelas, histórico, ranking

Tudo deve rodar localmente, sem internet.

A tabela usuarios terá uma coluna tipo (ex: "aluno", "professor", "coordenador").