

## Módulo de Equipes

Foi implementado o módulo Gestão de Equipes, responsável por cadastrar e administrar equipes vinculadas aos projetos do DevLab. O foco foi atender aos requisitos de:

- CRUD completo de equipes

\_Projeto Integrador - Professor...

- Relacionamentos exigidos entre Projeto ↔ Equipe e Usuário ↔ Equipe (liderança)

- Rota de relacionamento PUT/PATCH  
/api/equipes/{id}/definir-lider/

- Restrições de acesso (admin/coordenador gerencia; usuário comum vê apenas o que participa).

Modelo de dados e relacionamentos

No app equipe, o modelo Equipe foi criado seguindo os relacionamentos solicitados:

- Projeto ↔ Equipe (1:N): um projeto pode ter várias equipes; cada equipe pertence a um único projeto.
- Usuário ↔ Equipe (liderança 1:1): cada equipe tem um líder; um usuário pode ser líder de no máximo uma equipe.

\_Projeto Integrador - Professor...

- Membros (opcional): a equipe pode ter vários membros (usuários), modelados via relacionamento N:N, conforme permitido no enunciado.

Regras e validações aplicadas

- Líder deve ser usuário participante do projeto (consistência de dados entre projeto/equipe).
- Um líder por equipe e um usuário líder de no máximo uma equipe, garantindo o requisito 1:1.

Endpoints entregues no app equipe

Além do CRUD padrão do ModelAndViewSet (listar, detalhar, criar, atualizar, remover) - conforme requisito de rotas por entidade

Foi implementado:

- PUT/PATCH /api/equipes/{id}/definir-lider/

Define/atualiza o líder da equipe (rota exigida no enunciado).

- POST /api/equipes/{id}/adicionar-membro/

Adiciona um usuário como membro da equipe (valida participação no projeto).

- DELETE /api/equipes/{id}/remover-membro/{usuario\_id}/

Remove um usuário da equipe.

O enunciado também sugere rotas do tipo GET/POST /api/projetos/{id}/equipes/.

O modelo foi preparado com `related_name='equipes'` para facilitar essa implementação no app de Projetos.

### **Autenticação e autorização**

O app segue a regra de que criação/edição de equipes deve ser restrita a coordenadores/admins, enquanto usuários comuns acessam apenas o que está relacionado à sua participação.

### **Principais arquivos do módulo**

- equipe/models.py → modelo Equipe e validações
- equipe/serializers.py → serializers de saída (incluindo detalhes de líder/membros)
- equipe/views.py → EquipeViewSet + ações (definir-lider, adicionar/remover membro)
- equipe/urls.py → rotas registradas via DefaultRouter
- equipe/migrations/ → versionamento do banco