# 

# Alunos:

Pedro Dias

Tiago Rato

Gonçalo Fernandes

# Documento Rest

## 1. AmigosController

Este controlador é responsável pela gestão de amizades no sistema. As funcionalidades incluem:

- Listar amizades aceites (`GET /api/amigos`).

- Listar amigos de um utilizador específico (`GET /api/amigos/user/{userId}`).

- Listar amizades pendentes de um utilizador (`GET /api/amigos/user/{userId}/pending`).

- Obter uma amizade específica pelo ID (`GET /api/amigos/{id}`).

- Criar uma nova amizade (`POST /api/amigos`).

- Eliminar uma amizade por ID (`DELETE /api/amigos/{id}`).

- Aceitar um pedido de amizade (`PATCH /api/amigos/{utilizador1Id}/accept/{utilizador2Id}`).

## 2. AtividadesController

Este controlador gere as operações relacionadas com atividades. As funcionalidades incluem:

- Listar todas as atividades (`GET /api/atividades`).

- Obter uma atividade por ID (`GET /api/atividades/{id}`).

## 3. CategoriasController

Gere as categorias. Inclui:

- Listar todas as categorias (`GET /api/categorias`).

- Obter uma categoria por ID (`GET /api/categorias/{id}`).

## 4. DespesasController

Este controlador lida com a gestão de despesas. As funcionalidades incluem:

- Listar todas as despesas (`GET /api/despesas`).

- Obter uma despesa específica pelo ID (`GET /api/despesas/{id}`).

## 5. GrupoController

Este controlador é responsável pela gestão de grupos:

- Listar todos os grupos (`GET /api/grupo`).

- Obter detalhes de um grupo por ID (`GET /api/grupo/{id}`).

## 6. UgController

Lida com operações entre utilizadores e grupos, incluindo:

- Criar um novo grupo (`POST /api/grupos/create`).

- Adicionar um utilizador a um grupo (`POST /api/grupos/{grupoId}/add-user/{userId}`).

- Remover um utilizador de um grupo (`DELETE /api/grupos/{grupoId}/remove-user/{userId}`).

- Eliminar um grupo (`DELETE /api/grupos/{grupoId}`).

- Listar utilizadores num grupo (`GET /api/grupos/{grupoId}/users`).

- Listar os grupos de um utilizador (`GET /api/grupos/user/{userId}/groups`).

## 7. UserController

Este controlador gere os utilizadores, incluindo:

- Listar todos os utilizadores (`GET /api/user`).

- Obter um utilizador por ID (`GET /api/user/{id}`).

- Procurar utilizadores por nome (`GET /api/user/search`).

- Registar um novo utilizador (`POST /api/user/register`).

- Iniciar sessão e gerar token JWT (`POST /api/user/login`).

- Validar token JWT (`GET /api/user/validate`).