

## **Relatório de Atividades – Projeto de API com FastAPI e CSV**

Disciplina: Desenvolvimento de Software para Persistência

Professor(a): Francisco Victor

Dupla:

- Aluno 1: Elpidio Viana Cabral Neto - 536369
- Aluno 2: Renan Victor de Almeida Silva - 538428

Tema do Projeto: Sistema de Gerenciamento para RPG de Mesa

### **Resumo do Projeto**

Este trabalho teve como objetivo a criação de uma API RESTful utilizando FastAPI, com base em três entidades específicas do universo de RPG de mesa: Personagem, Habilidade e Equipamento. A persistência de dados foi feita por meio de arquivos .csv, e diversas funcionalidades foram implementadas de forma modular e reutilizável, conforme solicitado nas instruções do trabalho.

### **Divisão de Atividades da Dupla**

#### **Aluno 1: Renan Victor**

- Definiu as entidades do sistema com base no domínio RPG de mesa.
- Implementou as classes Python com Pydantic (Personagem, Habilidade, Equipamento) com validações apropriadas.
- Desenvolveu os endpoints de CRUD (F1) para as três entidades, garantindo que cada um atualizasse corretamente seu próprio CSV.
- Implementou as funcionalidades F2 (listar) e F3 (contagem) para todas as entidades.
- Implementou a funcionalidade F4 (compactar CSV em ZIP) com retorno por download.
- Auxiliou nos testes com Swagger UI, garantindo que todas as rotas estavam acessíveis e funcionais.

#### **Aluno 2: Elpidio Cabral**

- Desenvolveu os filtros por atributos (F5) usando parâmetros de query string nas rotas de listagem.
- Criou os endpoints para cálculo do hash SHA256 (F6) de cada CSV.

- Desenvolveu o sistema de logs (F7) com registro em arquivo .log, registrando inserções, atualizações, exclusões e falhas.
- Implementou a exportação para XML (F8) com download dos arquivos gerados para cada entidade.
- Implementou o consumo da API via terminal, com menu interativo e 100% funcional.
- Testou todas as funcionalidades.

### Funcionalidades Concluídas

Funcionalidade	Descrição	Status
F1	CRUD completo de Personagem, Habilidade e Equipamento	✓ Concluído
F2	Listagem de todos os registros do CSV em JSON	✓ Concluído
F3	Contagem total de registros no CSV	✓ Concluído
F4	Geração e download do arquivo ZIP com CSV	✓ Concluído
F5	Filtro por atributos via query params	✓ Concluído
F6	Cálculo do hash SHA256 dos arquivos CSV	✓ Concluído
F7	Sistema de logs para ações na API	✓ Concluído
F8	Exportação dos dados do CSV para XML com download	✓ Concluído

### Observações Finais

Todas as funcionalidades propostas foram implementadas com sucesso. Os testes foram realizados via Swagger UI (/docs), e os arquivos .csv foram atualizados corretamente durante as operações. Cada funcionalidade respeita a separação de contexto por entidade, conforme exigido. Os arquivos .zip, .xml e .log são gerados automaticamente no diretório de execução.