



ENSINO MÉDIO INTEGRADO - INFORMÁTICA
Disciplina de Desenvolvimento Web | Backend + Frontend
Trabalho Avaliativo Final

Gil Eduardo de Andrade

SISTEMA - TEMA LIVRE

O aluno (ou dupla de alunos) deve(m) escolher um tema para o desenvolvimento de uma solução utilizando um dos frameworks front-end (Angular, React, Svelte e Vue) e back-end estudados (Laravel, FastAPI, Spring Boot e NestJS). O sistema deve conter, obrigatoriamente, os seguintes conceitos:

- a. Interface da aplicação (Front-end) desacoplada do Back-end
 - Devem ser implementadas no mínimo de 5 páginas;
 - As páginas devem ser organizadas em componentes;
 - O conceito de “state” deve ser utilizado no Front-end para manipular os dados apresentados.
- b. Conexão com Banco de Dados (MySQL, Postgres, Mongo ou Neo4j)
 - A base de dados deve conter um mínimo de 5 tabelas;
 - Todo o processo de criação e versionamento das tabelas do banco deve ser feito através do framework de back-end escolhido, utilizando processos estruturados tais como migração, entre outros.
- c. Autenticação
 - O sistema deve permitir que novos usuários sejam cadastrados;
 - O acesso a aplicação deve ser permitida apenas após o usuário ter efetuado o processo de autenticação;

- Os usuários devem estar vinculados a um papel na aplicação.

d. Autorização

- Cada papel deve conter um nível de privilégio, previamente definido;
- A aplicação deve verificar, em todas as requisições efetuadas pelo usuário autenticado, se o mesmo possui ou não para acessar o recurso (rota / endpoint) requisitado.
- Caso o papel do usuário autenticado não possua permissão de acesso a um recurso requisitado, a solicitação deve ser negada.

e. Organização de Código e Boas Práticas

- A aplicação deve estar organizada conforme princípios de boas práticas e arquitetura de software, tais como:
 - Separação Mínima de Tarefas, tais como Controladores, Serviços e Repositórios;
 - Estruturada em camadas, modular, entre outros.

A aplicação deve ser apresentada ao professor em sala de aula.

REPOSITÓRIOS DE SUPORTE (Docker):

Front-end

- [Angular](#)
- [React](#)
- [Svelte](#)
- [Vue](#)

Back-end

- [Laravel \(MySQL\)](#)
- [FastAPI - Clean Architecture\(Postgres\)](#)
- [SpringBoot \(MongoDB\)](#)
- NestJS (Neo4j)