


UNIVERSIDADE FEDERAL DO CEARÁ		 UNIVERSIDADE FEDERAL DO CEARÁ CAMPUS DE ITAPAJÉ
PROFESSOR:	ALDÍSIO G. MEDEIROS	
DISCIPLINA:	PROGRAMAÇÃO WEB	
ASSUNTO:	Criação de validações de frontend com a linguagem JavaScript; Criação de endpoints via API REST com Python utilizando Flask. Utilização da autenticação JWT;	

ORIENTAÇÕES

1. É necessária a criação de endpoints que tornem os dados de exibição da lista de produtos de forma dinâmica (todas as informações tais como a listagem de produtos e a imagem do produto devem estar armazenadas no backend)
2. O frontend deve exibir todas as informações apresentadas no trabalho 1, porém trazendo as informações do backend;
3. É necessário a criação de um endpoint para autenticar o usuário (email/login + senha)
4. Crie um formulário de login (email/login + senha) que direciona o usuário para a tela de carrinho de compras, quando o usuário autentica corretamente. Caso ele erre as credenciais de acesso, apresente uma mensagem de erro;
5. Todos os formulários devem possuir validação de preenchimento e de formatação utilizando JavaScript:
 - a. No cadastro do usuário deve validar todos os campos incluindo a checagem da confirmação da senha se está igual a primeira;
 - b. Todas as mensagens de erro devem aparecer no próprio frontend, ao lado ou abaixo do respectivo campo.
6. A tela de carrinho de compras, ao usuário clicar em excluir item (ou seja reduzir a quantidade para zero) você deve remover o item da listagem (utilize JavaScript)
7. Construa estruturas de tabelas para produtos, categorias e usuários;
8. O cadastro de produtos e categorias não precisa ser dinâmico (não é necessário CRUD)
9. O cadastro de um novo usuário deve ser armazenado em um banco de dados, pelo backend (você deve apresentar os dados que foram salvos na(s) tabela(s))

Critérios de Avaliação

1. Validação dos campos no formulário de cadastro do usuário e login do usuário, via Javascript;
2. Endpoint do tipo GET para a listagem de categorias;
3. Exibição do tipo GET da lista de categorias de forma dinâmica (trazendo todos os dados do backend)
4. Endpoint do tipo GET para a listagem de produtos filtrando por categoria (recebendo o ID como parâmetro);

5. Exibição da lista de categorias, produtos, detalhes do produto de forma dinâmica (trazendo todos os dados do backend de acordo com a categoria)
6. Endpoint do tipo GET para trazer os detalhes e as informações do produto (recebendo um id de produto como parâmetro)
7. Endpoint do tipo POST para cadastro de usuário e um endpoint do tipo POST para autenticação de usuário;
8. Fluxo de autenticação completo (tela de login -> Autenticação das credenciais -> tela de carrinho de compras)