

Contents

1	Mas afinal, o que o back end vai fazer?	1
1.1	Cadastrar um novo usuário	1
1.2	Mandar dados pro front end	2
1.3	Editar um usuário	2
1.4	deletar um usuário	2
1.5	Verificar se um usuário já existe no banco de dados	2
1.6	Autenticar um usuário	2
1.7	Endpoints da API:	3
1.7.1	nov (POST)	3
1.7.2	cur (GET)	3
1.7.3	edit (POST)	3
1.7.4	del (POST)	3
1.7.5	auth (POST)	3
1.7.6	verU (GET)	3

1 Mas afinal, o que o back end vai fazer?

- Receber currículo do front end
- Cadastrar um novo currículo
- Deletar um currículo cadastrado
- Mandar currículo pro front end
- Verificar se um currículo já existe no banco de dados
- Verificar se um usuário digitou CPF e senha corretos

1.1 Cadastrar um novo usuário

Um usuario registra todos os seus dados no sistema e então o front end envia esses dados na forma de JSON pro back end. O front end deve verificar se o CPF já foi cadastrado antes de enviar os dados pro backend, ele faz isso pela funcao verUsuario.

Isso é feito no endpoint nov.

1.2 Mandar dados pro front end

O frontend pode consultar por um currículo já cadastrado. O frontend faz uma busca usando o id do currículo. Se o currículo existe, o backend retorna como JSON o currículo do banco de dados. Se não existe, o backend retorna NULL.

Isso é feito no endpoint cur.

1.3 Editar um usuário

Primeiramente, no front end o usuário digita CPF e senha. O back end então tem que dizer se esse usuário tem ou não autorização de editar os dados. Se o back dizer pro front que sim, o frontend pede os dados do currículo associado ao CPF e vai pra tela de edição, onde da pra editar tudo menos CPF e senha. O front então manda esses dados de volta pro backend pra ser salvos.

Isso é feito no endpoint edit.

1.4 deletar um usuário

Igual no editar o usuário, primeiramente o usuário entra CPF e senha. O frontend verifica se o currículo associado existe no banco de dados e verifica se a senha ta certa. Se existir e a senha estiver correta, o banco de dados retorna autorização pro front end. Quando o front end recebe a autorização, ele faz um pedido pro banco de dados deletar o arquivo.

Isso é feito no endpoint del.

1.5 Verificar se um usuário já existe no banco de dados

Um pedido é enviado ao backend pedindo confirmação sobre se já existe um usuário no banco de dados associado a um CPF.

Isso é feito no endpoint verU.

1.6 Autenticar um usuário

Autentica um usuário pelo cpf e pela senha pra conceder permissão de editar dados do currículo ou de deletar seu currículo do banco de dados.

Isso é feito no endpoint auth.

1.7 Endpoints da API:

1.7.1 nov (POST)

recebe dados de um currículo pra gravar ele no disco.

1.7.2 cur (GET)

retorna os dados de um currículo para o frontend

1.7.3 edit (POST)

edita um currículo já existente com dados novos

1.7.4 del (POST)

diz pro backend deletar um currículo se o cpf e senha fornecidos forem válidos.

1.7.5 auth (POST)

verifica se um usuário existe e forneceu a senha correta.

1.7.6 verU (GET)

verifica se um usuário já existe no banco de dados