

Casos de Testes

CESLA

Automação de testes

1. Endpoint: Login

1.1 Caso de Teste #1: Cadastro de Login

Dado que o usuário possui um email e senha válidos

Quando o usuário envia uma solicitação POST para o endpoint de login com as credenciais

Então o sistema deve retornar um status code 200

E o corpo da resposta deve conter os campos "createdAt", "email", "token", "password" e "id" preenchidos

E o email e senha retornados devem ser do tipo string

1.2 Caso de Teste #2: Validação do Login com GET após POST

Dado que o usuário foi criado com sucesso e o "id" do usuário é conhecido

Quando o usuário realiza uma solicitação GET para o endpoint de login informando o parâmetro "id"

Então o sistema deve retornar um status code 200

E o corpo da resposta deve conter o mesmo "id", "createdAt", "email", "token" e "password" que foram recebidos na resposta do POST

E o email e senha retornados devem ser do tipo string

1.3 Caso de Teste #3: Listagem de logins registrados

Dado que o usuário deseja ver uma lista dos logins já registrados

Quando o usuário realiza uma solicitação GET para o endpoint de login sem o parâmetro de ID

Então o sistema deve retornar um status code 200

E o corpo da resposta deve conter uma lista de registros com as informações de "id", "createdAt", "email", "token" e "password"

E o email e senha retornados devem ser do tipo string

1.4 Caso de Teste #4: Exclusão de registro de login

Dado que um login foi registrado com um "id" conhecido

Quando o usuário realiza uma solicitação DELETE para o endpoint de login informando o parâmetro "id"

Então o sistema deve retornar um status code 200

E o login deve ser removido do sistema

2. Endpoint: Estudante

2.1 Caso de Teste #5: Registro de Novo Estudante

Dado que o usuário tem informações válidas para registrar um novo estudante

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name" com uma string válida, "email" com uma string válida, "birthdate" com uma string válida, "academic_record" com uma string válida e "cpf" com uma string válida

Então o sistema deve retornar um status code 201

E o corpo da resposta deve conter todos os dados enviados, incluindo um campo "id"

E o registro acadêmico deve ser imutável e único, e o CPF deve ser imutável

2.3 Caso de Teste #6: Consulta de Estudantes registrados

Dado que o usuário deseja ver uma lista dos estudantes já cadastrados

Quando o usuário realiza uma solicitação GET para o endpoint de estudante sem o parâmetro "id"

Então o sistema deve retornar um status code 200

E o corpo da resposta deve conter uma lista dos estudantes já registrados com as informações de "name", "birthdate", "email", "academic_record", "cpf" e "id"

E os parâmetros retornados "name", "birthdate", "email" e "cpf" devem ser do tipo string

2.4 Caso de Teste #7: Consulta de Estudante por ID

Dado que um estudante foi registrado com um "id" conhecido

Quando o usuário realiza uma solicitação GET para o endpoint de estudante informando o parâmetro "id"

Então o sistema deve retornar um status code 200

E o corpo da resposta deve conter os dados "name", "birthdate", "email", "academic_record", "cpf" e "id" do estudante correspondentes

E os parâmetros retornados "name", "birthdate", "email" e "cpf" devem ser do tipo string

2.5 Caso de Teste #8: Tentativa de consulta de Estudante por ID inexistente

Dado que um estudante foi registrado com um "id" conhecido

Quando o usuário realiza uma solicitação GET para o endpoint de estudante com o "id"

Então o sistema deve retornar um status code 404, pois o ID não deve ser encontrado

2.6 Caso de Teste #9: Exclusão de Estudante

Dado que um estudante foi registrado com um "id" conhecido

Quando o usuário realiza uma solicitação DELETE para o endpoint de estudante com o "id"

Então o sistema deve retornar um status code 200

E o estudante deve ser removido do sistema

2.7 Caso de Teste #10: Tentativa de exclusão de Estudante com ID inexistente

Dado que o usuário tem um id inexistente de estudante

Quando o usuário realiza uma solicitação DELETE para o endpoint de estudante com o "id"

Então o sistema deve retornar um status code 404, pois o id não deve ser encontrado

2.8 Caso de Teste #11: Tentativa de Edição de Registro Acadêmico

Dado que um estudante possui um registro acadêmico

Quando o usuário tenta atualizar o registro acadêmico

Então o sistema deve retornar um status code 400

E a atualização do registro acadêmico deve falhar, pois o campo não é editável

2.9 Caso de Teste #12: Tentativa de Edição de CPF

Dado que um estudante possui um cpf

Quando o usuário tenta atualizar o cpf

Então o sistema deve retornar um status code 400

E a atualização do cpf deve falhar, pois o campo não é editável

2.10 Caso de Teste #13: Registro de Novo Estudante com registro acadêmico já existente

Dado que o usuário tem informações válidas para registrar um novo estudante
Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name" com uma string válida, "email" com uma string válida, "birthdate" com uma string válida e "cpf" com uma string válida
E informando um dado já existente para o parâmetro "academic_record"
Então o sistema deve retornar um status code 400
E a inserção do novo registro deve falhar, pois o campo "academic_register" deve ser único

2.11 Caso de Teste #14: Edição de Campos com método PUT

Dado que um estudante foi registrado com dados editáveis
Quando o usuário atualiza o nome, email e birthdate por meio do método PUT
Então o sistema deve refletir essas mudanças nas consultas subsequentes
E a resposta deve confirmar a atualização dos campos editáveis
E o registro acadêmico e o CPF devem continuar inalterados pois são campos não editáveis

2.12 Caso de Teste #15: Inclusão de novo registro com campo obrigatório em branco: nome

Dado que o usuário deseja incluir um novo estudante com a informação de nome em branco e as demais válidas
Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "email" com uma string válida, "birthdate" com uma string válida, "academic_register" com uma string válida e "cpf" com uma string válida
E informando vazio para o parâmetro "name"
Então o sistema deve retornar um status code 400
E a inserção do novo registro deve falhar, pois o campo "name" deve ser obrigatório

2.13 Caso de Teste #16: Inclusão de novo registro com campo obrigatório em branco: email

Dado que o usuário deseja incluir um novo estudante com a informação de email em branco e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name" com uma string válida, "birthdate" com uma string válida, "academic_register" com uma string válida e "cpf" com uma string válida

E informando vazio para o parâmetro "email"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "email" deve ser obrigatório

2.14 Caso de Teste #17: Inclusão de novo registro com campo obrigatório em branco: academic_record

Dado que o usuário deseja incluir um novo estudante com a informação de registro acadêmico em branco e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name", "email" com uma string válida, "birthdate" com uma string válida e "cpf" com uma string válida

E informando vazio para o parâmetro "academic_record"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "academic_record" deve ser obrigatório

2.15 Caso de Teste #18: Inclusão de novo registro com campo obrigatório em branco: cpf

Dado que o usuário deseja incluir um novo estudante com a informação de cpf em branco e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name", "email" com uma string válida, "birthdate" com uma string válida e "academic_register" com uma string válida

E informando vazio para o parâmetro "cpf"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "cpf" deve ser obrigatório

2.16 Caso de Teste #19: Inclusão de novo registro com campo obrigatório nulo: nome

Dado que o usuário deseja incluir um novo estudante com a informação de nome faltando e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "email" com uma string válida, "birthdate" com uma string válida, "academic_register" com uma string válida e "cpf" com uma string válida

E não informando o parâmetro "name"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "name" deve ser obrigatório

2.17 Caso de Teste #20: Inclusão de novo registro com campo obrigatório nulo: email

Dado que o usuário deseja incluir um novo estudante com a informação de email faltando e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name" com uma string válida, "birthdate" com uma string válida, "academic_register" com uma string válida e "cpf" com uma string válida

E não informando o parâmetro "email"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "email" deve ser obrigatório

2.18 Caso de Teste #21: Inclusão de novo registro com campo obrigatório nulo: academic_record

Dado que o usuário deseja incluir um novo estudante com a informação de registro acadêmico faltando e as demais válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name", "email" com uma string válida, "birthdate" com uma string válida e "cpf" com uma string válida

E não informando o parâmetro "academic_record"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "academic_record" deve ser obrigatório

2.19 Caso de Teste #22: Inclusão de novo registro com campo obrigatório nulo: cpf

Dado que o usuário deseja incluir um novo estudante com a informação de cpf faltando e as restantes válidas

Quando o usuário envia uma solicitação POST para o endpoint de estudante informando os parâmetros: "name", "email" com uma string válida, "birthdate" com uma string válida e "academic_register" com uma string válida

E não informando o parâmetro "cpf"

Então o sistema deve retornar um status code 400

E a inserção do novo registro deve falhar, pois o campo "cpf" deve ser obrigatório