

USJT - UC Gestão de Qualidade de Software - Turma: ADS1AN-BUC1-6272430

RA 823159742 - Igor Cordeiro de Souza Pereira - 823159742@ulife.com.br

RA 823217461 - Lucca Palmieri Dittrich - 823217561@ulife.com.br

RA 823123930 - Eduardo Vieira de Jesus - 823123930@ulife.com.br

RA 82426451 - Eduardo Filipe Silva S. Santos - 82426451@ulife.com.br

Exercício prático 1:

- Tema:
Técnica de Testes Ágeis - TDD;
- Objetivo:
Aplicar a técnica de testes ágeis TDD para o exercício apresentado em aula pelo professor;
- Entrega:
Até o prazo estipulado pela atividade proposta pelo *Google Classroom*, criar um documento contendo os resultados solicitados e os seguintes dados dos integrantes do grupo:
 - Nome para o Grupo; e
 - RA, Nome completo, E-mail, Turma e Curso de cada integrante do Grupo.
- Observação:
Gerar um PDF desse documento e, cada integrante do grupo, devolver como atividade, uma cópia desse documento.

TDD

Função de teste

```
function ordenaTeste(proposto, esperado, inesperado) {  
  ordena(proposto);  
  // Verifica se o comprimento do vetor proposto é igual ao do esperado e diferente  
do inesperado.  
  const comprimentoIgual = proposto.length === esperado.length &&  
proposto.length !== inesperado.length;  
  // Verifica se o valor proposto está ordenado corretamente.  
  const arrayOrdenada = JSON.stringify(proposto) === JSON.stringify(esperado);  
  console.log('Teste ordena\n =====');  
  console.log("Comprimento correto:", comprimentoIgual);  
  console.log("Array está ordenada corretamente:", arrayOrdenada);  
}
```

Função que ordena

```
function ordena(arr) {  
  return arr.sort((a, b) => a - b);  
}
```

Run:

```
ordenaTeste(proposto, esperado, inesperado);
```

Valores:

```
const proposto = [7, 3, 2, 1];  
const esperado = [1, 2, 3, 7];  
const inesperado = [9];
```

Saída:

```
Teste ordena  
=====  
Comprimento correto: true  
Array está ordenada corretamente: true
```

Valores:

```
const proposto = [7, 2, 2, 1];  
const esperado = [1, 2, 3];  
const inesperado = [7];
```

Saída:

```
Teste ordena  
=====  
Comprimento correto: false  
Array está ordenada corretamente: false
```

Valores:

```
const proposto = [4, 2, 2, 1];  
const esperado = [1, 2, 3, 4];  
const inesperado = [7];
```

Saída:

```
Teste ordena  
=====  
Comprimento correto: true  
Array está ordenada corretamente: false
```

Exercício prático 2:

- Tema:
Técnica de BDD;
- Objetivo:
Aplicar a técnica de BDD para o exercício apresentado em aula pelo professor;
- Entrega:
Até o prazo estipulado pela atividade proposta pelo *Google Classroom*, criar um documento contendo os resultados solicitados e os seguintes dados dos integrantes do grupo:
 - Nome para o Grupo; e
 - RA, Nome completo, E-mail, Turma e Curso de cada integrante do Grupo.
- Observação:
Gerar um PDF desse documento e, cada integrante do grupo, devolver como atividade, uma cópia desse documento.

BDD

Example mapping

História:

Como um usuário,
Eu quero autenticar no sistema inserindo login e senha,
Para que eu possa acessar a área restrita.

Regras:

1 - Verificar credenciais corretas

Se o login e senha corresponderem aos dados armazenados no banco de dados, o acesso deve ser permitido.

2 - Rejeitar credenciais incorretas

Se o login e/ou senha estiverem incorretos, o sistema deve recusar o acesso e exibir uma mensagem de erro.

3 - Campos obrigatórios

Login e senha são campos obrigatórios. Se estiverem em branco, o sistema deve exibir uma mensagem informando que os campos são obrigatórios.

Exemplos:

1. Credenciais corretas

- Dado que o usuário insere o login "joao" e a senha "senha123"
 - E os dados correspondem ao armazenado no banco de dados
 - Então o sistema permite o acesso.
2. Senha incorreta
- Dado que o usuário insere o login "joao" e a senha "senhaerrada"
 - E a senha não corresponde ao armazenado no banco de dados
 - Então o sistema recusa o acesso e exibe a mensagem "Credenciais inválidas".
3. Login incorreto
- Dado que o usuário insere o login "usuarioerrado" e a senha "senha123"
 - E o login não corresponde ao armazenado no banco de dados
 - Então o sistema recusa o acesso e exibe a mensagem "Credenciais inválidas".
4. Campos obrigatórios não preenchidos
- Dado que o usuário não preenche o campo login e/ou senha
 - Então o sistema exibe a mensagem "Os campos de login e senha são obrigatórios".

Perguntas:

1. Qual é o número máximo de tentativas falhas permitidas antes de bloquear o usuário?
2. Por quanto tempo o usuário deve ser bloqueado após múltiplas tentativas falhas?
3. Existe um processo para recuperar o acesso, como redefinição de senha?
4. O sistema armazena hashes das senhas ou senhas em texto claro no banco de dados?