

CENTRO UNIVERSITÁRIO NOSSA SENHORA DO PATROCÍNIO
ANÁLISE E DESENVOLVIMENTO DE SISTEMAS

KAIO VINÍCIUS DE ASSIS

RGM 33690197

Kauan Pavan de Farias

RGM: 32614489

Vanessa Gyrlany RibeiroPinto

RGM: 31673261

Bruno Batista Nogueira

RGM: 35818891

Eduardo Mito

RGM: 34604880

Bruno Rufatto Oliveira Coleraus

RGM: 33368546

Kaique Rebonato

RGM: 33515646

Gabriel Henrique de Almeida Neves

RGM: 34988025

PROGRAMAÇÃO ORIENTADA A OBJETOS

Desenvolvimento de Formulário de Cadastro

SALTO
2024

1. INTRODUÇÃO

Nosso projeto está sendo criado para fornecer dados sensíveis a usuários específicos, então foi necessário criar uma área de controle de acesso e, com isso, acabou-se criando uma área de validação de usuários.

Utilizando tecnologias web padrão, como HTML, CSS e JavaScript, o formulário foi projetado para fornecer feedback imediato aos usuários e garantir a integridade das informações submetidas durante o processo de cadastro.

Os formulários de cadastro desempenham um papel essencial em muitas aplicações web, permitindo que os usuários forneçam informações pessoais para acessar serviços e recursos oferecidos pelo sistema. No entanto, é fundamental garantir a validação dos dados inseridos assim garantindo a funcionalidade e segurança da aplicação. Neste contexto, a validação em tempo real surge como uma forma eficaz para melhorar a experiência do usuário, fornecendo feedback imediato sobre a validade dos dados inseridos e evitando erros de entrada.

O formulário de cadastro foi desenvolvido utilizando HTML para a estruturação dos elementos, CSS para a estilização e JavaScript para a lógica de validação em tempo real. Cada campo do formulário foi associado a eventos de digitação (keyup) que acionam funções de validação para verificar a conformidade dos dados inseridos.

No código abaixo, não há criação explícita de classes. Todas as funcionalidades foram implementadas utilizando atributos e métodos dentro do escopo global do JavaScript, sem a definição de uma classe específica para encapsular esses elementos.

2. ATRIBUTOS E MÉTODOS

Atributos:

1. **nome, labelNome, validNome:** Representam respectivamente o input do nome, a label associada ao input do nome e uma flag que indica se o nome é válido.

2. **usuario, labelUsuario, validUsuario:** Representam respectivamente o input do usuário, a label associada ao input do usuário e uma flag que indica se o usuário é válido.
3. **senha, labelSenha, validSenha:** Representam respectivamente o input da senha, a label associada ao input da senha e uma flag que indica se a senha é válida.
4. **confirmSenha, labelConfirmSenha, validConfirmSenha:** Representam respectivamente o input da confirmação de senha, a label associada ao input da confirmação de senha e uma flag que indica se a confirmação de senha é válida.
5. **msgError, msgSuccess:** Representam respectivamente os elementos HTML para exibir mensagens de erro e sucesso.

Métodos:

1. **addEventListener:** usada para vincular um manipulador de eventos a um elemento HTML.
2. **cadastrar():** Função que realiza o cadastro, verificando se todos os campos estão preenchidos corretamente e, em caso afirmativo, armazena os dados no localStorage. Esta função também manipula a exibição de mensagens de sucesso ou erro.

3. CÓDIGO

```
let nome = document.querySelector('#nome');
let labelNome = document.querySelector('#labelNome');
let validNome = false;

let usuario = document.querySelector('#usuario');
let labelUsuario = document.querySelector('#labelUsuario');
let validUsuario = false;

let senha = document.querySelector('#senha');
```

```

let labelSenha = document.querySelector('#labelSenha');
let validSenha = false;

let confirmSenha = document.querySelector('#confirmSenha');
let labelConfirmSenha =
document.querySelector('#labelConfirmSenha');
let validConfirmSenha = false;

let msgError = document.querySelector('#msgError')
let msgSuccess = document.querySelector('#msgSuccess')

//verificando preenchimento correto
nome.addEventListener('keyup', () => {
  if (nome.value.length <= 2) {
    labelNome.setAttribute('style', 'color: red');
    labelNome.innerHTML = 'Nome *Insira no mínimo 3 caracteres';
    nome.setAttribute('style', 'border-color: red');
    validNome = false;
  } else {
    labelNome.setAttribute('style', 'color: green');
    labelNome.innerHTML = 'Nome';
    nome.setAttribute('style', 'border-color: green');
    validNome = true;
  }
});

usuario.addEventListener('keyup', () => {
  if (usuario.value.length <= 2) {
    labelUsuario.setAttribute('style', 'color: red');
    labelUsuario.innerHTML = 'Usuário *Insira no mínimo 3
caracteres';
    usuario.setAttribute('style', 'border-color: red');

```

```

        validUsuario = false;
    } else {
        labelUsuario.setAttribute('style', 'color: green');
        labelUsuario.innerHTML = 'Usuário';
        usuario.setAttribute('style', 'border-color: green');
        validUsuario = true;
    }
});

senha.addEventListener('keyup', () => {
    if (senha.value.length <= 4) {
        labelSenha.setAttribute('style', 'color: red');
        labelSenha.innerHTML = 'Senha *Insira no mínimo 5 caracteres';
        senha.setAttribute('style', 'border-color: red');
        validSenha = false;
    } else {
        labelSenha.setAttribute('style', 'color: green');
        labelSenha.innerHTML = 'Senha';
        senha.setAttribute('style', 'border-color: green');
        validSenha = true;
    }
});

//Confirmando que mesmo que o user muda a senha a confirmação
acompanha
confirmSenha.addEventListener('keyup', () => {
    if (senha.value !== confirmSenha.value || senha.value.length < 5)
    {
        labelConfirmSenha.setAttribute('style', 'color: red');
        labelConfirmSenha.innerHTML = 'Confirmar Senha *As senhas não
conferem';
        confirmSenha.setAttribute('style', 'border-color: red');
    }
});

```

```

        validConfirmSenha = false;
    } else {
        labelConfirmSenha.setAttribute('style', 'color: green');
        labelConfirmSenha.innerHTML = 'Confirmar Senha';
        confirmSenha.setAttribute('style', 'border-color: green');
        validConfirmSenha = true;
    }
});

senha.addEventListener('keyup', () => {
    if (senha.value !== confirmSenha.value || senha.value.length < 5)
    { //vai verificar se os dois estao iguais em tempo real
        labelConfirmSenha.setAttribute('style', 'color: red');
        labelConfirmSenha.innerHTML = 'Confirmar Senha *As senhas não
conferem';
        confirmSenha.setAttribute('style', 'border-color: red');
    } else {
        labelConfirmSenha.setAttribute('style', 'color: green');
        labelConfirmSenha.innerHTML = 'Confirmar Senha';
        confirmSenha.setAttribute('style', 'border-color: green');
    }
});

function cadastrar() { //cadastro
    if (validNome && validUsuario && validSenha && validConfirmSenha)
    {

        let listaUser = JSON.parse(localStorage.getItem('listaUser') ||
'[]')
    }
}

```

```

    listaUser.push(
    {
        nomeCad: nome.value,
        userCad: usuario.value,
        senhaCad: senha.value
    }
    )

    localStorage.setItem('listaUser', JSON.stringify(listaUser))

    msgSuccess.setAttribute('style', 'display: block')
    msgSuccess.innerHTML = '<strong>Cadastrado com sucesso..</strong>'
    msgError.setAttribute('style', 'display: none')
    msgError.innerHTML = ''

    setTimeout(()=>{

window.location.href = './index.html'

    }, 3000)
    } else {
        msgError.setAttribute('style', 'display: block')
        msgError.innerHTML = '<strong>Preencha todos os campos corretamente antes de cadastrar</strong>'
        msgSuccess.setAttribute('style', 'display: none')
        msgSuccess.innerHTML = ''
    }
}

```