

Implementação do componente **escola-form** no frontend

Considerações iniciais

Esta versão do sistema tem apenas a entidade pessoa e, para mais detalhes sobre como implementar os componentes de uma nova entidade, acesse a documentação do projeto.

Esta versão do sistema utiliza o banco de dados **H2** que cria um arquivo de banco de dados no diretório home do sistema operacional. No windows, o diretório home deve ser algo como:

C:\Users\{usuario}

Onde {usuario} corresponde ao seu nome de usuário.

As tabelas são criadas automaticamente após a primeira inicialização do servidor, isto é, com a inicialização do servidor, o spring boot é iniciado e, o mesmo, cria as tabelas na sua primeira inicialização.

Então vamos lá!

Criação do componente: escola-form

Criação dos arquivos

Conforme a explicação contida na documentação crie os seguintes arquivos:

`componentes/app/escola/form/escola-form.html`

Insira nesse arquivo o seguinte conteúdo:

```
<div>
  <h1>Formulário de escolas</h1>
  <div id="mensagem-el"></div>
</div>
```

`componentes/app/escola/form/EscolaForm.js`

Insira o seguinte conteúdo:

```
class EscolaForm {
  onCarregado( jsonObj, params ) {
    sistema.mostraMensagem( 'mensagem-el', 'info', 'Pagina carregada com sucesso!' );
  }
}
```

Vinculação HTML e JS do componente

Configure no arquivo `js/principal.js`, a instanciação do objeto e a vinculação da página ao objeto instanciado.

```
let escolaForm = new EscolaForm();

let componentes = {
  { 'escola-form', { pagina : 'componentes/app/escola/form/escola-form.html', jsonObj :
escolaForm }
}
```

Adição do script JS criado ao index.html

Após isto, adicione o script criado ao arquivo `index.html`. Isto é, antes do script `principal.js` mas após os demais.

```
<head>
...
<script src="componentes/app/escola/form/EscolaForm.js"></script>
</head>
```

Alteração da página inicial

Para especificar qual página deve ser carregada junto com a página index, altere o arquivo `componentes/app/layout/AppLayout.js` conforme abaixo:

```
onCarregado( jsonObj, params ) {  
  ...  
  sistema.carregaPagina( 'escola-form' );  
}
```

Rodando o projeto

Para testar o que já foi feito até aqui, basta executar o servidor e digitar `localhost:8080` no navegador.

Se houver algum problema na exibição da página, utilize o **console de execução do javascript** para informação sobre o erro e tente consertá-lo revisando o conteúdo explicado até aqui.

Após tudo funcionando, podemos alterar o componente de nome: `escola-tela`.

Alterando o componente criado

Após criado o componente, adicione no arquivo `escola-form.html` criado o seguinte conteúdo:

```
<form name="escola_form">  
  <label for="nome">Nome:</label>  
  <br />  
  <input type="text" name="nome" />  
  <br />  
  <input type="submit" value="Salvar" onclick="escolaForm.salvar()" />  
</form>
```

Agora acesse o **swagger** através da seguinte url: <http://localhost:8080/swagger-ui>

Procure pelo endpoint: `registraEscola` e verifique o método e url do endpoint e, após isto, verifique também o formato do corpo da requisição em: `Request Body`.

Agora que você já sabe como utilizar o serviço de cadastro, adicione a classe `EscolaForm` o método `salvar`, conforme definido no atributo `onclick` do botão `salvar` no formulário **HTML**. Implemente também a função `limpaForm` que deve ser chamada em caso de sucesso no registro.

```
salvar() {  
    let instance = this;  
    sistema.ajax( 'POST', '/escola/registra', {  
        cabecalhos : {  
            'Content-Type' : 'application/json; charset=UTF-8'  
        },  
        corpo : JSON.stringify( {  
            nome : document.escola_form.nome.value  
        } ),  
        sucesso : function( resposta ) {  
            instance.limpaForm();  
            sistema.mostraMensagem( 'mensagem-el', 'info', 'Escola salva!' );  
        },  
        erro : function( msg ) {  
            sistema.mostraMensagem( 'mensagem-el', 'erro', msg );  
        }  
    } );  
}  
limpaForm() {  
    document.escola_form.nome.value = "";  
}
```

Feitas estas alterações, você já pode recarregar a página para visualizar o formulário **HTML** de registro de escolas.

Acessando o console do H2

Faça alguns cadastros no formulário de escolas criado e, após isto, acesse o console do banco de dados **H2** no navegador **WEB** conforme abaixo:

<http://localhost:8080/h2-console>

Entre com os dados:

url: `jdbc:h2:file:~/sisescolar`

username: `sa`

password:

Após informar os dados acima e clicar em `connect`, você pode visualizar as tabelas criadas para o banco de dados `sisescolar`. São elas: `PESSOA` e `ESCOLA`.

Agora, execute um `select` para verificar se os registros através do formulário de escolas realmente foram feitos no banco de dados. Escreva um simples `select` conforme abaixo:

```
select * from escola;
```

Com a execução dessa instrução **SQL**, o console deve lhe mostrar as escolas que você registrou!