

Documentação do arquivo GetUsersAPI.cls

Introdução

Este arquivo contém a classe GetUsersAPI, que é uma API RESTful desenvolvida em Apex para o Salesforce. Sua principal funcionalidade é permitir a busca de usuários no sistema com base em um nome fornecido como parâmetro.

Descrição

A classe GetUsersAPI é uma classe global com compartilhamento de registros habilitado (with sharing). Ela expõe um endpoint RESTful que permite a busca de usuários no Salesforce com base em um nome parcial. A busca é realizada utilizando a cláusula LIKE no SOQL (Salesforce Object Query Language), permitindo encontrar usuários cujo nome contenha o texto fornecido.

A classe possui um único método, getUsersByName, que é acessível via requisições HTTP POST. Este método recebe um nome como parâmetro, realiza a consulta no banco de dados e retorna uma lista de usuários que correspondem ao critério de busca.

Estrutura

A estrutura do arquivo é simples e consiste em:

- Uma anotação @RestResource para definir o mapeamento de URL do endpoint.
- Uma classe global chamada GetUsersAPI.
- Um método estático global chamado getUsersByName, que é exposto como um endpoint HTTP POST.

Dependências

A classe depende dos seguintes elementos:

- Objeto padrão User do Salesforce.
- Linguagem SOQL para realizar consultas no banco de dados.

Imports

Não há imports explícitos, pois o Salesforce Apex não utiliza declarações de importação como em outras linguagens. No entanto, a classe utiliza o objeto padrão User.

Variáveis

A classe utiliza as seguintes variáveis locais:

- userName: Uma string que armazena o nome do usuário formatado com caracteres curinga (%) para a busca.
- lUsers: Uma lista de objetos User que armazena os resultados da consulta SOQL.

Métodos

getUsersByName(String name)

Descrição:

Este método realiza uma busca no banco de dados Salesforce para encontrar usuários cujo nome contenha o texto fornecido como parâmetro.

Parâmetros:

- name (String): O nome ou parte do nome do usuário a ser buscado.

Retorno:

- `List<User>`: Uma lista de objetos `User` que correspondem ao critério de busca.

Funcionamento:

1. O método recebe o parâmetro `name`.
2. Adiciona caracteres curinga (%) antes e depois do nome para realizar uma busca parcial.
3. Executa uma consulta SOQL no objeto `User` para buscar registros cujo nome corresponda ao critério.
4. Retorna a lista de usuários encontrados.

Anotações:

- `@HttpPost`: Define que o método será acessado via requisições HTTP POST.
- `global static`: Torna o método acessível globalmente e estático, ou seja, pode ser chamado sem instanciar a classe.

Exemplo

Abaixo está um exemplo de como utilizar o endpoint exposto pela classe `GetUsersAPI`:

Requisição HTTP

POST /services/apexrest/getUsers/
Content-Type: application/json

```
{
  "name": "John"
}
```

Resposta HTTP

```
[
  {
    "Id": "0051U00000078abcQAA",
    "Name": "John Doe"
  },
  {
    "Id": "0051U00000078xyzQAA",
    "Name": "Johnny Appleseed"
  }
]
```

Diagrama de Dependência

O diagrama abaixo ilustra a relação entre a classe `GetUsersAPI` e o objeto `User`:

```
classDiagram
    class GetUsersAPI {
        +List~User~ getUsersByName(String name)
    }
    class User {
        +Id
        +Name
    }
    GetUsersAPI --> User : Consulta SOQL
```

Notas

- A classe utiliza a anotação `with sharing`, o que significa que respeita as permissões de compartilhamento do Salesforce. Apenas os registros que o usuário atual tem permissão para visualizar serão retornados.
- Certifique-se de que o endpoint está habilitado e acessível para os usuários ou sistemas que precisam consumi-lo.

Vulnerabilidades

- **Injeção SOQL:** Embora o uso de variáveis bind (:userName) proteja contra injeção SOQL, é importante validar o parâmetro name antes de utilizá-lo.
- **Exposição de Dados Sensíveis:** Certifique-se de que os campos retornados na consulta não contenham informações sensíveis ou desnecessárias.
- **Controle de Acesso:** Verifique se o endpoint está protegido contra acessos não autorizados, utilizando perfis, permissões ou autenticação adequada.