

Documentação do arquivo

AtendechatToCaseWebhook.cls

Introdução

Este arquivo contém a classe `AtendechatToCaseWebhook`, que é uma classe Apex exposta como um serviço RESTful. Ela é projetada para lidar com mensagens recebidas de um sistema de atendimento via WhatsApp, criando ou atualizando casos no Salesforce com base nos dados recebidos.

Descrição

A classe `AtendechatToCaseWebhook` é responsável por processar mensagens recebidas de um webhook. Quando uma mensagem é recebida, ela verifica se já existe um caso associado ao `sessionId` fornecido. Se não existir, um novo caso é criado. Caso contrário, um comentário é adicionado ao caso existente. Essa funcionalidade é útil para integrar sistemas de atendimento ao cliente, como o WhatsApp, ao Salesforce.

Estrutura

A classe possui uma única anotação RESTful (`@RestResource`) e um método principal (`handleMessage`) que processa as mensagens recebidas.

Dependências

- Salesforce REST API
- Objeto padrão `Case`
- Objeto padrão `CaseComment`
- Classe `RestRequest` e `RestContext` para manipulação de requisições REST
- Classe `JSON` para desserialização de dados JSON

Imports

Não há imports explícitos, pois a classe utiliza recursos nativos do Salesforce Apex.

Variáveis

A classe não possui variáveis globais ou de instância. Todas as variáveis são locais ao método `handleMessage`.

Variáveis locais no método `handleMessage`:

- `req`: Representa a requisição REST recebida.
- `body`: Contém o corpo da requisição em formato de string.
- `data`: Um mapa que armazena os dados desserializados do JSON recebido.
- `sessionId`: Identificador único da sessão, extraído do JSON.
- `phone`: Número de telefone do remetente, extraído do JSON.
- `msg`: Mensagem enviada pelo remetente, extraída do JSON.
- `existing`: Lista de casos existentes que correspondem ao `sessionId`.

Métodos

`handleMessage`

Descrição: Método principal que processa a mensagem recebida via webhook.

Anotação: @HttpPost - Indica que este método será chamado quando uma requisição HTTP POST for feita ao endpoint REST.

Passos executados:

1. Obtém a requisição REST atual usando `RestContext.request`.
2. Lê o corpo da requisição e o converte para uma string.
3. Desserializa o JSON recebido em um mapa (`Map<String, Object>`).
4. Extrai os valores de `sessionId`, `from` (telefone) e `message` do mapa.
5. Verifica se já existe um caso no Salesforce com o `sessionId` fornecido.
6. Se não existir:
 - Cria um novo caso (`Case`) com os dados fornecidos.
 - Define os campos `Subject`, `Description`, `Origin`, `Status`, `SuppliedPhone` e `Session_Id__c`.
 - Insere o caso no Salesforce.
7. Se existir:
 - Cria um comentário (`CaseComment`) associado ao caso existente.
 - Define os campos `ParentId`, `CommentBody` e `IsPublished`.
 - Insere o comentário no Salesforce.

Parâmetros: Nenhum.

Retorno: Nenhum (void).

Exemplo

Requisição HTTP POST

POST /services/apexrest/atendechat/inbound HTTP/1.1
Content-Type: application/json

```
{
  "sessionId": "12345",
  "from": "+5511999999999",
  "message": "Olá, preciso de ajuda com meu pedido."
}
```

Resultado esperado

1. Se não existir um caso com `Session_Id__c = "12345"`, será criado um novo caso com os seguintes campos:
 - `Subject`: "Atendimento WhatsApp"
 - `Description`: "Olá, preciso de ajuda com meu pedido."
 - `Origin`: "WhatsApp"
 - `Status`: "Novo"
 - `SuppliedPhone`: "+5511999999999"
 - `Session_Id__c`: "12345"
2. Se já existir um caso com `Session_Id__c = "12345"`, será adicionado um comentário ao caso com o texto: "Olá, preciso de ajuda com meu pedido."

Diagrama de Dependência

O diagrama abaixo ilustra as dependências e interações da classe `AtendechatToCaseWebhook`:

```
classDiagram
    class AtendechatToCaseWebhook {
        +handleMessage() void
    }
    class RestRequest {
        +requestBody : Blob
    }
    class RestContext {
        +request: RestRequest
    }
```

```

class JSON {
    +deserializeUntyped(String) : Object
}
class Case {
    +Id : String
    +Subject : String
    +Description : String
    +Origin : String
    +Status : String
    +SuppliedPhone : String
    +Session_Id__c : String
}
class CaseComment {
    +ParentId : String
    +CommentBody : String
    +IsPublished : Boolean
}

AtendechatToCaseWebhook --> RestContext
RestContext --> RestRequest
AtendechatToCaseWebhook --> JSON
AtendechatToCaseWebhook --> Case
AtendechatToCaseWebhook --> CaseComment

```

Notas

- Certifique-se de que o campo personalizado Session_Id__c exista no objeto Case no Salesforce.
- O método handleMessage não retorna nenhuma resposta HTTP. Caso seja necessário, você pode adicionar lógica para retornar mensagens de sucesso ou erro.

Vulnerabilidades

- **Validação de entrada:** O código não valida os dados recebidos no JSON. É importante adicionar validações para evitar erros ou ataques de injeção.
- **Autenticação e autorização:** Certifique-se de que apenas sistemas autorizados possam acessar este endpoint REST.
- **Limitação de requisições:** Considere implementar limites de taxa para evitar abusos do endpoint.