

Documentação do arquivo SendMessageAtendechat.cls

Introdução

O arquivo `SendMessageAtendechat.cls` contém uma classe Apex que permite enviar mensagens para o serviço Atendechat utilizando uma API HTTP. Ele é projetado para ser usado em sistemas Salesforce que precisam integrar-se com o serviço de mensagens Atendechat.

Descrição

A classe `SendMessageAtendechat` é uma classe pública com compartilhamento habilitado (`with sharing`) que contém um método estático chamado `send`. Este método é responsável por enviar uma mensagem para um número de telefone específico utilizando a API do Atendechat. Ele constrói uma requisição HTTP POST com um corpo JSON contendo o número de telefone e o texto da mensagem, e envia essa requisição para o endpoint da API.

Estrutura

A estrutura do arquivo é simples e consiste em:

- Uma classe pública chamada `SendMessageAtendechat`.
- Um método estático chamado `send` que realiza a operação de envio de mensagens.

Dependências

A classe depende das seguintes funcionalidades:

- `HttpRequest` e `HttpResponse` para realizar requisições HTTP.
- `JSON.serialize` para serializar o corpo da requisição em formato JSON.
- `System.debug` para registrar mensagens de depuração.
- A API externa do Atendechat.

Imports

A classe utiliza as seguintes classes e métodos nativos do Salesforce:

- `HttpRequest`
- `HttpResponse`
- `Http`
- `JSON`
- `System`

Variáveis

A classe utiliza as seguintes variáveis locais:

- `req`: Instância de `HttpRequest` usada para configurar e enviar a requisição HTTP.
- `payload`: Um mapa (`Map<String, Object>`) que contém os dados da mensagem (número de telefone e corpo da mensagem).
- `http`: Instância de `Http` usada para enviar a requisição.
- `res`: Instância de `HttpResponse` que armazena a resposta da API.
- `ex`: Instância de `Exception` usada para capturar erros durante o envio da mensagem.

Métodos

Método: `send`

Descrição:
Este método estático é responsável por enviar uma mensagem para um número de telefone específico utilizando a API do AtendeChat.

Parâmetros:

- `phoneNumber` (String): O número de telefone para o qual a mensagem será enviada.
- `messageBody` (String): O texto da mensagem que será enviada.

Funcionamento:

1. Cria uma instância de `HttpRequest` e configura o endpoint, método HTTP, cabeçalhos e corpo da requisição.
2. Serializa o corpo da requisição em formato JSON.
3. Envia a requisição utilizando uma instância de `Http`.
4. Captura e registra a resposta da API ou qualquer erro que ocorra durante o envio.

Exceções:

- Captura qualquer exceção que ocorra durante o envio da mensagem e registra o erro usando `System.debug`.

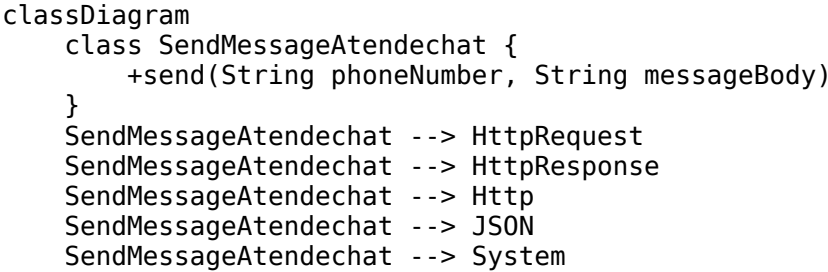
Exemplo

Aqui está um exemplo de como usar a classe `SendMessageAtendeChat`:

```
String phoneNumber = '+5511999999999';
String messageBody = 'Olá, esta é uma mensagem de teste!';
SendMessageAtendeChat.send(phoneNumber, messageBody);
```

Diagrama de Dependência

O diagrama abaixo ilustra as dependências da classe `SendMessageAtendeChat`:



Notas

- Certifique-se de que o token de autorização (`Bearer 123`) seja atualizado para um valor válido antes de usar a classe em produção.
- O endpoint da API (`https://chatapi.sticonsulting.net/api/messages/send`) deve estar acessível e configurado corretamente.
- A classe não realiza validação do número de telefone ou do corpo da mensagem. Certifique-se de fornecer valores válidos.

Vulnerabilidades

- **Token de autorização hardcoded:** O token de autorização está codificado diretamente no código. Isso pode representar um risco de segurança. Recomenda-se armazenar o token em um local seguro, como o Salesforce Custom Settings ou o Salesforce Protected Custom Metadata.
- **Ausência de tratamento detalhado de erros:** O método captura exceções genéricas, mas não fornece informações detalhadas sobre o tipo de erro ou como lidar com ele.
- **Dependência de um endpoint externo:** A funcionalidade da classe depende de um serviço externo. Se o serviço estiver indisponível, a classe não funcionará corretamente.