

IMPLEMENTAÇÃO JWT E SDK ZOOM ANGULAR

Filipe Nunes Soares
Email: filipe.soares.fn@gmail.com

¹ Velp Tecnologia
R. Antônio Quirino da Silva, 544 - 2º andar - Ingá, Betim - MG, 32604-458

Abstract. *A brief description of how to integrate the zoom API E SDK with the Angular framework.*

Resumo. *Uma breve descrição de como fazer a integração da API E SDK do zoom com o framework Angular.*

1. Introdução

O Zoom possui um Marketplace onde desenvolvedores podem criar seus próprios aplicativos utilizando as API's e SKD's fornecidas pela empresa. Ao criar uma conta como desenvolvedor, podemos dar inicio ao desenvolvimento de aplicativos utilizando suas ferramentas.

2. JWT

O JWT é disponível pela zoom e ele é o responsável por criar as reuniões personalizadas. Ao criar um aplicativo JWT é disponibilizado algumas credenciais como API KEY e API SECRET além de um Token responsável por fazer autenticação do usuário para poder criar uma reunião. Para implementação foi utilizado a Documentação disponibilizada pela Zomm.

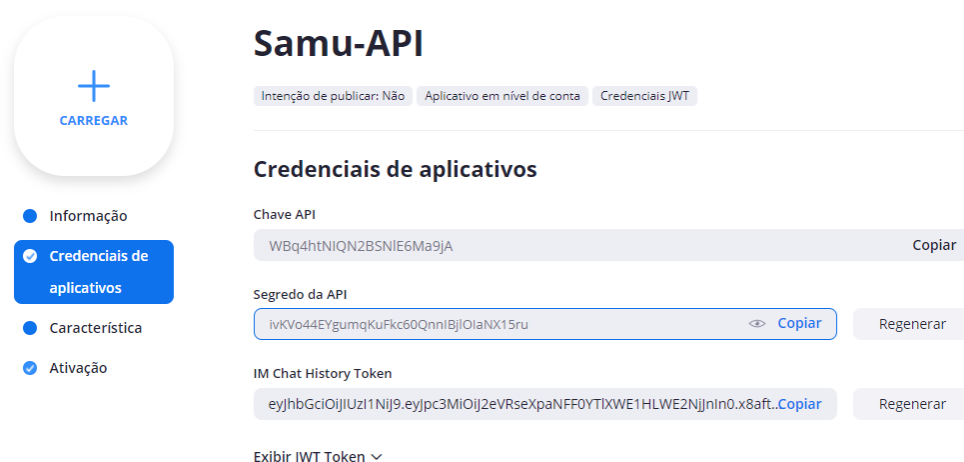


Figura 1. Credenciais Geradas JWT

3. SDK

Um aplicativo SDK é responsável por permitir um usuário entrar em uma video chamada já existente. O SDK usa o número Meeting para acessar a video chamada, e por conta dele podemos fazer o acesso em qualquer plataforma. Ao criar um aplicativo, também é criado chave e segredo responsáveis pela validação. Para implementação foi utilizado a Documentação disponibilizada pela Zoom.

The screenshot shows the Zoom SDK Credentials page for an application named 'SamuVideoConferencia'. On the left, there is a sidebar with a 'CARREGAR' button and a list of options: 'Informação', 'Baixar', 'Credenciais de aplicativos' (selected), 'Escopos', and 'Ativação SDK'. The main content area has tabs for 'Intenção de publicar: Não', 'Aplicativo gerenciado pelo usuário', and 'Credenciais SDK OAuth'. Under the 'Credenciais SDK' section, there are fields for 'Chave SDK' (ptoYElvf7AahV9uY8nXW7Q5BiXT05nEftAU) and 'Segredo SDK' (cLi2vXEyCYdtzNdPX7ypeY4GkCnRu7NoMihC), both with 'Copiar' buttons. Below this is the 'Credenciais OAuth' section, which includes a description, 'ID do cliente' (4pTE9mo4SIKZUSTSXZsgSw), 'Segredo do Cliente' (5xDb2oyA8CYSPOEVQCE8jvKk9Hr3dVI), and a 'Redirecionar URL para OAuth' field with the value 'https://yourcompany.com/redirect'. Each of these fields has a 'Copiar' button.

SamuVideoConferencia

Intenção de publicar: Não Aplicativo gerenciado pelo usuário Credenciais SDK OAuth

Credenciais SDK

Chave SDK
ptoYElvf7AahV9uY8nXW7Q5BiXT05nEftAU Copiar

Segredo SDK
cLi2vXEyCYdtzNdPX7ypeY4GkCnRu7NoMihC Copiar Regenerar

Credenciais OAuth

Abaixo a credencial permite gerar um token que é utilizado pelo Zoom OAuth, fornecendo-lhe acesso às APIs do Zoom.

ID do cliente
4pTE9mo4SIKZUSTSXZsgSw Copiar

Segredo do Cliente
5xDb2oyA8CYSPOEVQCE8jvKk9Hr3dVI Copiar Regenerar

Redirecionar URL para OAuth
URL de destino onde o Zoom enviará o token de acesso após o usuário concluir a autenticação do OAuth.
https://yourcompany.com/redirect

Figura 2. Credenciais Geradas JWT

4. Implementação

A implementação foi feita utilizando o framework Angular, criando uma simples tela possibilitando a criação de uma nova reunião, deletar uma reunião que já tenha acabado, e entrar em uma reunião existente. Foi utilizado tanto o JWT e o SDK, e para fazer as requisições do JWT ao servidor da Zoom para criar novas reuniões foi desenvolvido um Webservice em Java.

4.1. Problemas Encontrados e Observações

Durante a implementação do JWT para criação de uma nova reunião houve problemas com as requisições feitas pelo próprio Angular (Por parte do cliente), apresentando o erro CORS policy. Com alguns estudos foi constatado que para fazer qualquer requisição ao servidor da Zoom deve ser feito por parte do Web Service (Por parte do servidor).

```

1 @POST
2 @Consumes(MediaType.APPLICATION_JSON)
3 @Path("Meeting/Create")
4 public String sendPost(String body) throws MalformedURLException
   , IOException{
5
6     URL url = new URL( "https://api.zoom.us/v2/users/me/meetings
   " );
7     String response = null;
8
9     HttpURLConnection connection = (HttpURLConnection) url.
        openConnection();
10
11     connection.setRequestMethod("POST");
12     connection.setRequestProperty( "Authorization", "Bearer
        eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
        eyJpc3MiOiJXQnE0aHROSVFOMkJTTmxFNk1hOWpBIiwiaXhwIjoxNjUwNDU3Njg2NTU2LCJ
        .m8PSRC7B-xjEIM7GlHoEe_u5VVvk56Oi7-2n_0wFma4");
13     connection.setRequestProperty( "Content-Type", MediaType.
        APPLICATION_JSON);
14     connection.setDoOutput(true);
15
16     OutputStream os = connection.getOutputStream();
17     BufferedWriter wr = new BufferedWriter(
18         new OutputStreamWriter(os, "UTF-8"));
19     wr.write(body.toString());
20     wr.flush();
21     wr.close();
22
23     //Inicia a coleta da resposta da requisi o
24     BufferedReader in = new BufferedReader(new InputStreamReader
        (connection.getInputStream()));
25     String inputLine = null;
26     response = "";
27
28     while ((inputLine = in.readLine()) != null) {
29         response += inputLine + "\n";
30     }
31     in.close();
32     os.close();
33
34     int responseCode = connection.getResponseCode();
35     System.out.println("Response Code : " + responseCode);
36     System.out.println("Response : " + response);
37
38     return response;
39 }

```

Listing 1. Requisição de Criar Meeting no Web Service

```

1 @POST
2 @Consumes(MediaType.APPLICATION_JSON)
3 @Path("Meeting/Delete")
4 public void sendDelete(String idMeeting) throws
    MalformedURLException, IOException{
5
6     String uri = "https://api.zoom.us/v2/meetings/"+idMeeting;
7     URL url = new URL( uri );
8
9     HttpURLConnection connection = (HttpURLConnection) url.
        openConnection();
10
11     connection.setRequestMethod("DELETE");
12     connection.setRequestProperty( "Authorization", "Bearer
        eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.
        eyJpc3MiOiJXQnE0aHROSVFOMkJTTmxFNk1hOWpBIiwizXhwIjoxNjUwNDU3Njg2NTU2LCJ
        .m8PSRC7B-xjEIM7GlHoEe_u5VVvk56Oi7-2n_0wFma4");
13     connection.setRequestProperty( "Content-Type", MediaType.
        APPLICATION_JSON);
14     connection.setDoOutput(true);
15
16     int responseCode = connection.getResponseCode();
17     System.out.println("Response Code : " + responseCode);
18
19 }

```

Listing 2. Requisição de Deletar Meeting no Web Service

5. Como rodar o sistema

Para que o sistema funcione, você primeiramente tem que ter algumas coisas instaladas e na versão determinada. A aplicação pode ser encontrada no link: <https://github.com/FilipeLipe/API-Zoom>

- Angular CLI: 13.1.2
- Node: 14.17.0
- Package Manager: npm 6.14.13
- NetBeans: 8.2
- JDK: 1.8.0_291

Assim que tiver todas as dependências instaladas você deve clonar o projeto e abri-lo. A aplicação e o servidor de resposta devem ser abertos no VsCode, e o WebService deve ser aberto no NetBeans.

Vamos começar com a Aplicação, você deve entrar em seu repositório e utilizar o comando "npm i" para instalar todas as dependências necessárias para o projeto, mesma coisa com o repositório do servidor de resposta. Assim que terminado, em sua aplicação deve usar o comando "ng serve --open" e para o servidor de resposta, em um outro terminal utilizar o comando "node index.js".

Agora no NetBeans com o WebService aberto deve executá-lo, com isso seu projeto está pronto para funcionar.

6. Funcionalidade

Para entender o sistema, vamos repassar o passo a passo de como é a aplicação. Após os passos a cima terem sido seguidos e a aplicação estiver 100%, teremos a pagina principal que consiste em apenas um botão para a criação de uma nova reunião.

Aplicação JWT e SDK

Criar nova Meeting

Id Meeting -

Start Url -

Join Url -

Password -

Ao clicar no botão "Criar nova Meeting", será aberto um formulário para a criação de uma nova meeting com nome, duração e senha desejados.

Aplicação JWT e SDK

Criar nova Meeting

Id Meeting -

Start Url -

Join Url -

Password -

Criar Meeting

Meeting

Duração

Senha

Criar

Cancelar

Após escolher o nome, tempo de duração e senha, e criar, os botões mudarão para que seja possível deletar a meeting criada, e para entrar na mesma. Além de mostrar o ID da Meeting, URL de Start que é o do Host, URL de Join que é o dos participantes e a senha da meeting.

Aplicação JWT e SDK

Deletar Meeting

Entrar em Meeting

Id Meeting - 81324090401

Start Url - <https://us05web.zoom.us/j/81324090401?pwd=MFRSMUNHSjJ4LzR1UHJMU2FBdFRZz09>

zak=eyJ0eXAiOiJKV1QiLCJpdil6IjAwMDAwMSIsInptX3NrbSI6InptX28ybSIsImFsZyI6IkhTMjU2In0.eyJh

Join Url - <https://us05web.zoom.us/j/81324090401?pwd=MFRSMUNHSjJ4LzR1UHJMU2FBdFRZz09>

Password - 123

Caso deseje entrar na meeting, clique no botão, ele abrirá um formulário que já vem carregado com o id e senha da meeting criada. Mas caso queria entrar em outra meeting é só alterar os dados e acessar, sendo direcionado para clicar em join. E caso em vez de entrar, queira deletar a meeting, é só clicar em seu botão sendo redirecionado para a primeira view

Aplicação JWT e SDK

Deletar Meeting

Entrar em Meeting

Id Meeting - 84962916987

Start Url - <https://us05web.zoom.us/j/84962916987?pwd=TzNEVzJDRTJZU2tYN1RMd0hHc0INZz09>

zak=eyJ0eXAiOiJKV1QiLCJpdil6IjAwMDAwMSIsInptX3NrbSI6InptX28ybSIsImFsZyI6IkhTMjU2In0.eyJhdWQiO

Join Url - <https://us05web.zoom.us/j/84962916987?pwd=TzNEVzJDRTJZU2tYN1RMd0hHc0INZz09>

Password - 123

Entrar Meeting

Number Meeting

84962916987

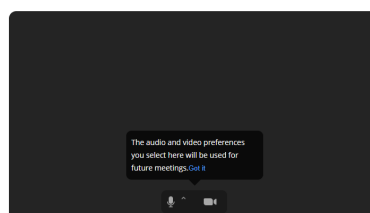
Senha

123

Entrar

Cancelar

Join Meeting



Join