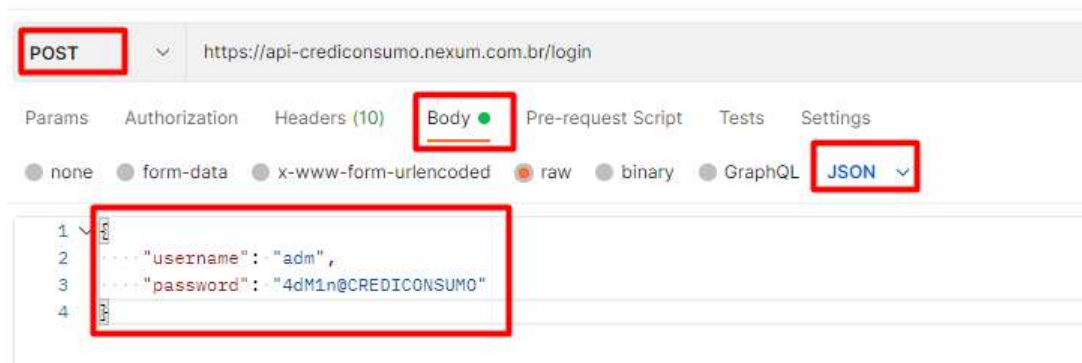


Api Nxbr

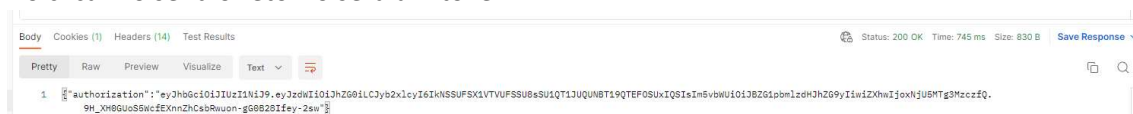
A ideia aqui é retirar as chamadas de user/senha do front e jogarmos para o back, dessa forma faremos inicialmente 2 chamadas no Postman para obtermos um token que será único, e ele será utilizado nos header das chamadas das apis do nxbr.

Porém importante aqui que vocês mudarão as chamadas dos js, e chamarão controller do Lecom que fara as chamadas das apis do nxbr, tudo isso para que não fique exposto esses códigos de segurança de autenticação.

1. Primeira chamada no Postman, será a de login, para obtermos o token de acesso: Colocando a url da api do nxbr do cliente, e incluindo os dados de user e senha



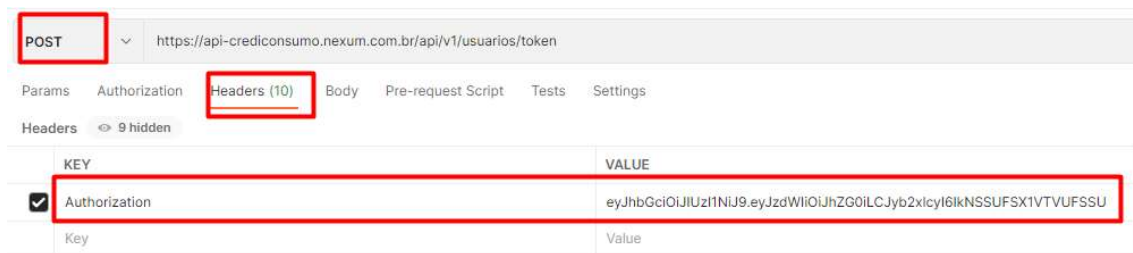
Ao clicar no send o retorno será um token:



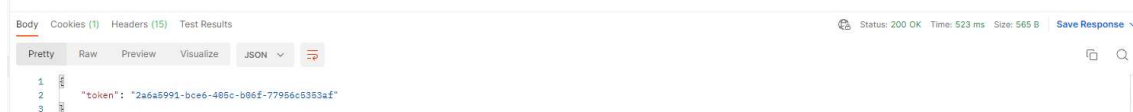
Copiaremos esse retorno para utilizarmos na próxima chamada.

2. Segunda chamada no Postman, será a de pegar o token único que usaremos em todas as outras chamadas.

Nessa parte do header é adicionar exatamente o retorno da chamada anterior:



Ao clicar no send eles nos retornarão um token:



E é esse token que será único, vejam só farão essas 2 chamadas apenas uma única x, e esse token ficará reservado em um properties do Lecom.

3. Criaremos um properties, que contemplará seguintes dados:

```
dadosNxbr.properties ×
1 token = bfcbf497-328f-4900-82c1-7d42a7222606
2 url = https://api-crediconsumo.nexum.com.br
```

Esse properties será colocado no diretório upload/cadastros/config no servidor.

4. Foi criado um controller, que vocês replicarão para cada cliente e colocando os métodos que precisão, inicialmente criei o controller baseado nas chamadas da crediconsumo, que utiliza dados do associado e saldo médio.

```
nxbr.controller
> ApisNxBr.java

28
29 @WebServlet("/app/public/apisNxBr")
30 public class ApisNxBr extends ControllerServlet{
31
32     /**
33      *
34      */
35     private static final long serialVersionUID = 1L;
36     private static final Logger logger = Logger.getLogger(ApisNxBr.class);
37     private static final String CAMINHOF = Funcoes.getWFFooterDir()+File.separator+"upload"+File.separator+"cadastros"+File.separator+"confi
38
39     public void getAssociado(HttpServletRequest request, HttpServletResponse resp) throws ServletException, IOException {}
40
41     public void getSaldoMedio(HttpServletRequest request, HttpServletResponse resp) throws ServletException, IOException {}
42
43 }
```

E nessa classe poderia ter todos os métodos necessários do nexbr, pois na chamada dele utilizará o properties acima e enviando chamada de “back end to back end” ficando escondido no back o token gerado:

```
logger.info(" inicio api getassociado ");
String cpfAssociado = request.getParameter("cpf");
logger.info(" cpfAssociado = "+cpfAssociado);

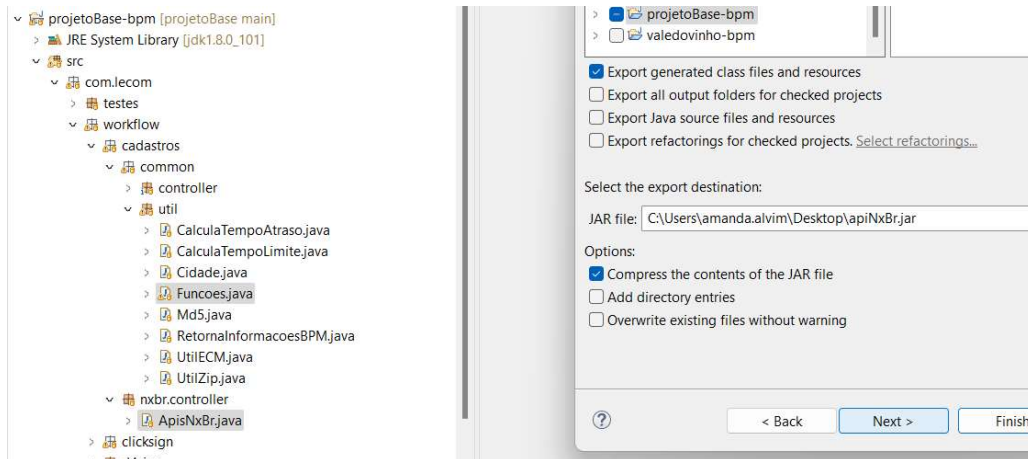
Map<String,String> dadosNxBr = Funcoes.getParametrosIntegracao(CAMINHOF + "dadosNxbr");

HttpClient client = HttpClientBuilder.create().build();
HttpGet get = new HttpGet(dadosNxBr.get("url")+"/api/v1/associados/?document="+cpfAssociado );
get.addHeader("Authorization",dadosNxBr.get("token"));
HttpResponse response = client.execute(get);
if(response.getStatusLine().getStatusCode() == HttpURLConnection.HTTP_OK) {
    StringBuilder result = new StringBuilder();
    try(BufferedReader rd = new BufferedReader(new InputStreamReader(response.getEntity().getContent()))){
        String line = "";
        while ((line = rd.readLine()) != null) {
            result.append(line);
        }
    }
    org.json.JSONObject json = new org.json.JSONObject(result.toString());
    org.json.JSONArray jsonArray = (JSONArray) json.get("content");
    org.json.JSONObject jsonObjeto = new JSONObject();
    if(jsonArray.length() > 0) {
        jsonObjeto = (JSONObject) jsonArray.get(0);
    }

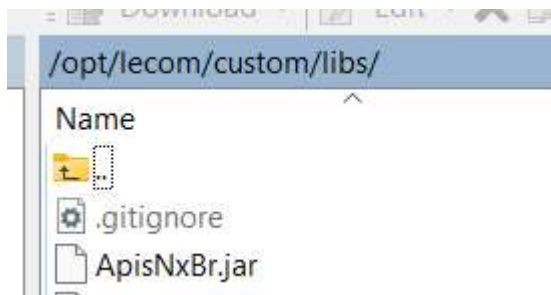
    retorno.put("associado",jsonObjeto);
}
}
```

5. Subindo controller em produção.

Pode-se subir o controller mais a classe funções juntas em um jar. Para isso gera-se o arquivo selecionando o controller e a funções:



E esse jar colocar na pasta libs:



Dar o update no servidor e pronto só utilizar no js conforme abaixo:

6. No js do modelo chamaria da seguinte maneira esse controller:

```

96
97 function buscaDadosAssociado(doc) {
98     doc = doc.replace(/[\^d]+/g, "");
99     setTimeout(1000);
100     $.ajax({
101         url: "https://app-hom.crediconsumo.lecom.com.br/bpm/app/public/apisNxBr?action=getAssociado&cpf=" + doc,
102         method: "GET",
103     }).done(function (response) {
104         if (response.error != undefined) {
105             alert(reponse.error);
106         } else if (response.msg != undefined) {
107             Form.fields('LT_TELEFONE').value("");
108             Form.fields('LT_EMAIL').value("");
109             Form.fields('LT_NOME_ASSOCIADO').value("");
110             Form.fields('DT_ASSOCIACAO').value("");
111             Form.fields('VLR_RENDA_ASSOCIADO').value("");
112             Form.apply();
113
114             alert("Documento " + Form.fields("LT_CPF_CNPJ").value() + " não encontrado no sistema!");
115         } else {

```

Exemplo do retorno de associado encontrado e não encontrado

```

{"associado":{"tipoPessoa":"PF","telefone":[{"telefone":"(15) 997351660","tipoContato":"CELULAR","id":1}, {"telefone":"(984132888","tipoContato":"CELULAR","id":62250}], "observacao":null, "endereco":{"uf":"SP", "complemento":"BL D AP21", "end VIEIRA PEDRICO", "bairro":"VILA GUILHERME", "municipio":"VOTORANTIM", "id":2, "tipoLogradouro":"AVENIDA", "cep":"18115390"}] 15T00:00:00.000+00:00", "descricaoProfissao":"NAO SE APLICA", "rendaBrutaMensal":2262.51, "nomeAssociado":"ADRIANO DE ALME RIO", "statusContaCapital":"ATIVO", "vinculoEmpregaticio":"EMPREGADOS DE EMPRESAS PRIVADAS APLICA", "contaCorrente":"-2", "numSisbr":82406, "id":2, "contaCapital":314528, "clusterizacaoDTO":null, "ultimaRenovacao": FISICA", "dataNascimento":"1979-01-19T00:00:00.000+00:00", "email":"MENDYYZ.AM@GMAIL.COM", "gerente":{"pa":{"numero":0, "id SUL"}, "nome":"JANINE GABRIELA MAIA", "id":25}}}

```

```

{"msg":"Associado não encontrado"}

```

7. Métodos já feitos:

- a. GetAssociado
Retorno `associado`
- b. GetSaldoMedio
Retorno `dadosSaldoMedio`
- c. GetSaldoMedioCCO
Retorno `dadosSaldoMedioCCO`
- d. GetEndividamento
Retorno `dadosEndividamento`
- e. GetMovimentacaoDiaria
Retorno `dadosAssociado`
- f. GetGrupoEconomico
Retorno `dadosGrupoEconomico`