



INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
RIO GRANDE DO NORTE

AULA:

Acessando API que necessita de autenticação (Estudo de caso API SUAP)

Programação Orientada a Serviços

Alba Lopes, Profa.

<http://docentes.ifrn.edu.br/albalopes>
alba.lopes@ifrn.edu.br

Exemplo de uso com Javascript e JQuery

- ▶ O primeiro passo é conseguir o token de acesso.
- ▶ Isso é feito através do fornecimento das credenciais autorizadas
 - ▶ No exemplo do SUAP, deve-se passar o login e a senha do usuário
- ▶ No nosso exemplo, vamos utilizar um formulário de login para que o próprio usuário forneça seu login e senha
- ▶ Ao fornecer as credenciais, a API devolve o token



Exemplo de uso com Javascript e JQuery

```
//Monitorar a função de clique no botão de login no arquivo login.html
$("#botao-login").click(function(e){
    e.preventDefault();
    //Recupera os dados digitados pelo usuário
    var username = $("#login").val();
    var password = $("#senha").val();
    //Converte no JSON esperado pela API (nesse caso, do SUAP, a API pede username e password)
    var dadosjson = JSON.stringify({"username": username, "password":password});
    //Efetiva a requisição AJAX
    $.ajax({
        url: "https://suap.ifrn.edu.br/api/v2/autenticacao/token/", /*Endereço do serviço de autenticação*/
        dataType: 'json', /*Tipo dos dados enviados na requisição*/
        data: dadosjson, /*Dados enviados na solicitação*/
        type: 'POST', /*Método da requisição*/
        contentType: 'application/json', /*Tipo de dado da resposta*/
        success: function (data) { /*Função que será executada caso a API tenha retornado uma resposta da classe 200*/
            alert(data.token); /*Como esperamos um dado do tipo JSON, informado no campo contentType, acessamos um desses campos*/
        },
        error: function(data){ /*Função que será executada caso a resposta não seja da classe 200*/
            alert("Impossível recuperar dados");
        }
    });
});
```



Armanezando o token de acesso

- ▶ Como a API exige a autenticação, e essa autenticação será confirmada através do token de acesso, é importante que o token seja armazenado para que não seja preciso solicitá-lo toda vez.
- ▶ Para armazená-lo, podemos utilizar o recurso do session storage do Javascript.
- ▶ Enquanto não fecharmos o navegador e o token estiver válido, podemos acessar o session storage e recuperar o token.
- ▶ Para tanto, vamos adicionar a funcionalidade `Session.storage` na função de `success`
 - ▶ Vamos ainda efetivar um redirecionamento para a página que teremos nossas funcionalidades.



Armanezando o token de acesso

```
$("#botao-login").click(function(e){
    e.preventDefault();

    var username = $("#login").val();
    var password = $("#senha").val();

    var dadosjson = JSON.stringify({"username": username, "password":password});

    $.ajax({
        url: "https://suap.ifrn.edu.br/api/v2/autenticacao/token/",
        dataType: 'json',
        data: dadosjson,
        type: 'POST',
        contentType: 'application/json',
        success: function (data) {

            sessionStorage.setItem("token", data.token);
            window.location.href="dados.html";

        },
        error: function(data){
            alert("Impossível recuperar dados");
        }
    });
});
```



Requisitando recursos que necessitam do token

- ▶ Uma vez recuperado e armazenado o token, podemos agora fazer requisições a API passando o token necessário
- ▶ O token precisa ser enviado no cabeçalho HTTP e é preciso indicar o formato de segurança utilizado
- ▶ No caso da API que estamos usando, o token de autorização é no formato JWT (JSON Web Tokens)



Requisitando recursos que necessitam do token

```
//Assim que a página dados.html for completamente carregada, executa a funcionalidade abaixo
$(document).ready(function() {
    $.ajax({
        //Incluir no cabeçalho HTTP a forma de autenticação passando o token
        headers: {
            "Authorization" : "JWT "+sessionStorage.getItem("token")
        },
        //Informar a URL do recurso desejado
        url: "https://suap.ifrn.edu.br/api/v2/minhas-informacoes/meus-dados/",
        contentType: 'application/json',
        dataType: 'json',
        //Informar o método da requisição, no caso, GET
        type: 'GET',
        //Ação, caso sucesso
        success: function (data) {
            $("#usuario-nome_usual").html(data.nome_usual);
            $("#usuario-tipo_vinculo").html(data.tipo_vinculo);
            $("#usuario-email").html(data.email);
        },
        //Ação, caso erro
        error: function(data){
            alert("Impossível recuperar dados. Você deve fazer login!");
            window.location.href = "login.html";
        }
    });
});
```



Removendo o token da sessão do usuário

- ▶ É possível remover o token da sessão através do uso da função `sessionStorage.removeItem`. Ex:

```
$("#botao-sair").click(function(e) {  
    sessionStorage.removeItem("token");  
    window.location.href="login.html";  
});
```



Exercícios

1. Inclua uma nova linha na tabela da página dados.html para exibir a matrícula do usuário.
2. Exiba ainda a foto do usuário.
3. Como você pode fazer para exibir o nome completo do usuário logado ao invés do nome usual? Faça a modificação necessária no seu código para implementar essa funcionalidade.
4. Crie dois novos botões na página dados.html. Um chamado Disciplinas e outro chamado Próximas Avaliações.
5. Ao clicar no botão Disciplinas, o usuário deve visualizar a lista de disciplinas que ele está matriculado no semestre atual
6. Ao clicar no botão Próximas Avaliações, o usuário deve visualizar a lista das suas próximas avaliações

