

INSTITUTO FEDERAL

Paraná

Campus Paranaguá

JSON e AJAX

Desenvolvimento Web

Prof. Diego Stiehl

Comunicação Síncrona

- Na web “clássica” precisamos recarregar as páginas para recebermos novos conteúdos
 - O famoso **F5**
- O que acontece?
 - A página antiga morre
 - Uma nova é fornecida
 - Dados atualizados

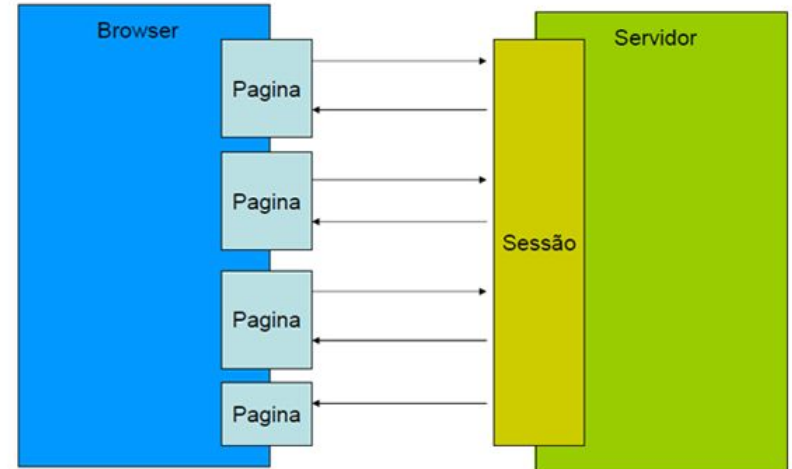


Figura 1. Comunicação síncrona cliente-servidor

<https://pt.slideshare.net/volverine22/aula-04-ferramentas-para-autoria-de-produtos-multimidia-i>

Comunicação Assíncrona

- Podemos usar JavaScript para fazer requisições menores (parciais) dentro da mesma página
 - Sem recarregar
 - Dinamicidade
 - Economia de dados
 - Sem servidor (nosso)

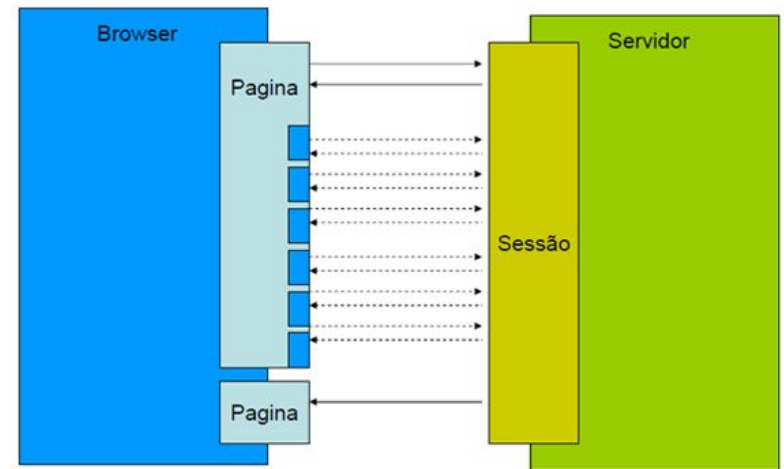
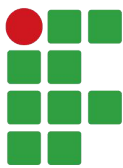


Figura 2. Comunicação assíncrona cliente-servidor

<https://pt.slideshare.net/volverine22/aula-04-ferramentas-para-autoria-de-produtos-multimidia-i>



AJAX

- Aynchronous JavaScript and XML
- Técnica que utiliza JavaScript para carregar dados externos (web) em **segundo plano**
 - O JavaScript faz uma requisição HTTP
 - A página atual não é recarregada no browser
 - Os dados retornados são consumidos pelo próprio JavaScript
 - Podem ser renderizados no HTML
 - Novamente sem recarregar a página

Processamento Assíncrono

- A palavra assíncrono significa que algo irá acontecer em paralelo ao programa principal
- Quando este processamento terminar, o código principal é avisado
 - O JavaScript usa *callback functions* para isso
- O **programa principal** não fica travado durante este processamento paralelo

Exemplo de Função Assíncrona

- Função `setTimeout(callback, tempo)`
 - Executa callback somente após o tempo estipulado
 - Ações do usuário no site não ficam bloqueadas
 - JavaScript não fica bloqueado

```
console.log('ANTES');
```

```
setTimeout(function() {  
    console.log('Olá! (+2 segundos)');  
}, 2000);
```

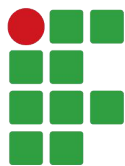
```
console.log('DEPOIS');
```

```
// Saída no terminal:  
//  
// ANTES  
// DEPOIS  
// Olá! (+2 segundos)
```

Exemplo de Função Assíncrona

- Função `setInterval(callback, tempo)`
 - Executa callback a cada intervalo estipulado
 - Ações do usuário no site não ficam bloqueadas
 - JavaScript não fica bloqueado

```
const div = document.querySelector('.interval');
let interval = 10;
div.textContent = interval;
let it = setInterval(function() {
    interval--;
    div.textContent = interval;
    if (interval <= 0) {
        div.textContent = 'Acabou!';
        clearInterval(it); // Pára repetição
    }
}, 1000);
```

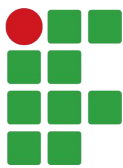


APIs

- A URL que fornece nossos dados é chamada de API (*Application Programming Interface*)
- Pode ser nossa
 - Do mesmo site que hospeda a página atual
- Pode ser de terceiros
 - Google Maps, YouTube, Facebook, Amazon, uma empresa parceira, algum site de receitas, outro site de previsão do tempo, ...

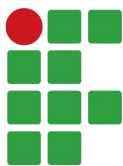
Dados de APIs

- URLs comuns normalmente devolvem recursos para o browser renderizar
 - HTML, imagens, CSS, ...
- A URL de uma API retorna dados brutos (crus)
 - XML ← formato clássico
 - JSON ← **formato moderno atualmente mais aceito**



JSON

- JavaScript Object Notation
- Formato que nasceu do JavaScript
- É um objeto JavaScript visto como String
- Fácil criar JSON com JavaScript
- Fácil criar objetos JavaScript com base em um documento JSON







JSON

```
{
  "informacoes": {
    "alunos": [
      {
        "nome": "Fritz",
        "notas": [10, 0, 7]
      },
      {
        "nome": "Franz",
        "notas": [5, 6, 8]
      },
      {
        "nome": "Berlin",
        "notas": [10, 10, 9]
      }
    ]
  }
}
```

Exemplos de APIs JSON

- MetaWeather: Clima
 - <https://www.metaweather.com/api/location/44418/>
- Open Trivia Database: Perguntas para quiz
 - <https://opentdb.com/api.php?amount=10&category=18&difficulty=hard>
- News API: Agregador de notícias
 - <https://newsapi.org/v2/top-headlines?country=br&category=business&apiKey=a8e9258d73934726966be7f4f51c16c3>
- Observação
 - Várias APIs exigem registro e autenticação

Fazendo Chamadas

- É possível fazer chamadas HTTP com JavaScript manualmente
 - Objeto [XMLHttpRequest](#)
- Existem bibliotecas que facilitam o serviço
 - A axios é uma delas
 - <https://github.com/axios/axios>
 - Muito utilizada  1,373,137  1,101  60,813  4,917
 - Facilita chamada
 - Lê e transforma JSON automaticamente



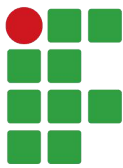
axios

- Estrutura da chamada
 - `axios.get(url, opções)`
 - Requisição e parâmetros
 - `then(função)`
 - Sucesso
 - `catch(função)`
 - Caso dê erro cai aqui
 - `finally(função)`
 - Executa ao final
 - Opcional

```
axios.get('/user', {  
  params: {  
    ID: 12345  
  }  
})  
  .then(function (response) {  
    console.log(response);  
  })  
  .catch(function (error) {  
    console.log(error);  
  })  
  .finally(function () {  
    // always executed  
  });
```

Chamando API de Clima

```
axios.get('https://www.metaweather.com/api/location/455827/')  
  .then(function(json) {  
    const weather = json.data;  
    console.log(`Próximas temperaturas em ${weather.title}`);  
    for (const day of weather.consolidated_weather) {  
      console.log(` em ${day.applicable_date}: ${day.the_temp.toFixed(0)}°C`);  
    }  
  })  
  .catch(function(erro) {  
    console.log(`Erro: ${erro.message}`);  
  });
```



Erro?

- Recebeu o erro abaixo?
 - Failed to load <https://www.metaweather.com/api/location/455827/>: No 'Access-Control-Allow-Origin' header is present on the requested resource. Origin 'null' is therefore not allowed access.
- É uma questão de segurança do browser
 - CORS (*Cross-Origin Resource Sharing*)
- Porque nosso HTML foi aberto localmente
 - <file:///home/usuário/Desktop/ajax/index.html>
- Solução (muito) temporária (e feia)
 - Concatenar um proxy
 - Usar: <https://cors-anywhere.herokuapp.com/>

Chamando API de Clima

- Atualizando a chamada

```
const proxy = 'https://cors-anywhere.herokuapp.com/';
axios.get(`${proxy}https://www.metaweather.com/api/location/455827/`)
  .then(function(json) {
    const weather = json.data;
    console.log(`Próximas temperaturas em ${weather.title}`);
    for (const day of weather.consolidated_weather) {
      console.log(`  em ${day.applicable_date}: ${day.the_temp.toFixed(0)}°C`);
    }
  })
  .catch(function(erro) {
    console.log(`Erro: ${erro.message}`);
  });
```

Chamando API de Notícias (+ parâmetros e DOM)

```
// https://newsapi.org/v2/top-headlines?country=br&category=business& apiKey=a8e9258d73934726966be7f4f51c16c3
const proxy = 'https://cors-anywhere.herokuapp.com/';
axios.get(`${proxy}https://newsapi.org/v2/top-headlines`, {
  params: {
    country: 'br',
    category: 'business',
    apiKey: 'a8e9258d73934726966be7f4f51c16c3'
  }
})
.then(function(json) {
  const noticias = json.data.articles;
  let div = document.querySelector('.noticias');
  div.textContent = '';
  for (const noticia of noticias) {
    div.innerHTML += `<div>
      
      <h2><a href="${noticia.url}">${noticia.title}</a></h2>
    </div>`;
  }
})
.catch(function(erro) {
  console.log(`Erro: ${erro.message}`);
});
```