



# **Técnico em desenvolvimento de sistemas**

## **Módulo II: Desenvolvimento Web I**

Professor: Lucas Martins de Oliveira



# Sobre o professor:

Formação:

- Bacharel em Ciência da Computação na Universidade Federal da Fronteira Sul.
- Graduação sanduíche em Science Computer na Universitetet i Oslo, Noruega.





## Sobre os alunos:

Agora é a vez de vocês me falarem sobre vocês, quero saber:

- Nome;
- Idade;
- Ano que está estudando;
- Disciplina do módulo I que mais gostou.



# Desenvolvimento Web I

- Arquitetura de Aplicações Web (Back-end e Front-end);
- Linguagem de programação para Back-end;
- Funções próprias da linguagem e aplicações;
- Memória de aplicação web (armazenamento de dados e cookies);
- API;
- Node.js
- Desenvolvimento de API's RESTful e Coleção de Dados;
- CRUD.



## Resumo da história da internet

- A internet surge como uma resposta do governo dos Estados Unidos para o lançamento soviético de satélites, no final da década de cinquenta;
- Iniciou conectando os principais centros de pesquisa do país com o Pentágono;
- A tecnologia utilizada na época se chamava WAN (Wide Area Network), mas era composta de linguagem muito complicada;
- Nos anos 70 começam a ser utilizados os primeiros emails, para comunicação entre universidades;



## Resumo da história da internet

- A internet atual funciona por meio do protocolo TCP/IP (Transmission Control Protocol / Internet Protocol), que lê a informação enviada e entrega para o destino escolhido pelo usuário;
- No final dos anos 80 a internet contava com pouco mais de 100 mil servidores;
- No ano de 1992, com o surgimento do WWW (World Wide Web) o número de servidores já passava de 1 milhão;
- Segundo o relatório Global Digital de 2024, aproximadamente 5,35 bilhões de pessoas tem acesso a internet no mundo, correspondendo a 66% da população mundial.



## 10 maiores empresas do mundo

1. Microsoft
2. Apple
3. Nvidia
4. Alphabet (Inc) (Google, Maps, Android, Chrome...)
5. Amazon
6. Saudi Aramco (Petróleo)
7. Meta Platforms (Facebook)
8. Berkshire Hathaway (Têxtil)
9. TSMC
10. Eli Lilly (medicamentos)



# Front-end

Front-end é a parte visual de um site ou aplicativo, com a qual os usuários interagem. É a camada de apresentação do software.

- Elementos visuais, como botões, caixas de seleção, gráficos e mensagens de texto;
- Interface gráfica desenvolvida com tecnologias como HTML, CSS e JavaScript.

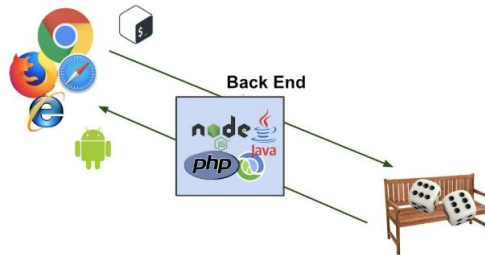


# Back-end

O front-end está mais próximo dos usuários, enquanto o back-end está mais próximo dos dados da aplicação.

- O back-end é responsável pela integração das informações contidas nos bancos de dados com o navegador

Uma pessoa que trabalha tanto com front-end quanto com back-end é chamada de full stack developer

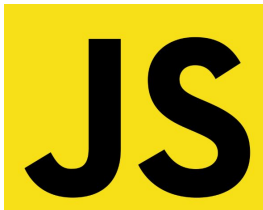




# Linguagem de programação para Back-end

Existem diversas opções para programação web back-end, como: PHP, Ruby, Java, C#... nesse curso usaremos a linguagem JavaScript.

Usaremos o Visual Studio Code e a extensão Live Server.





# Comentários em JavaScript

Para comentar uma linha usamos o comando `//` no início da linha comentada.

Para comentar diversas linhas usamos o comando `/*` para iniciar o comentário e `*/` para finalizar o comentário.

Ex.:

```
// comentário
```

```
/* comentário
```

```
comentário
```

```
comentário */
```



# Variáveis em JavaScript

Para declarar uma variável usamos o identificador **var** + nome da variável;

Ex.:

```
var x;
```

```
var x = 0;
```

```
var x = var y = 1;
```



# Tipos de variáveis em JavaScript

- **Inteiros:**

```
var x = 1; var y = 2;
```

- **Flutuantes:**

```
var x = 0.1; var y = 0.02;
```



# Tipos de variáveis em JavaScript

- **Strings:**

```
var nome1 = "eu"; var nome2 = 'você';
```

- **Booleano:**

```
var x = true; var y = false;
```



# Comparadores

Usamos == para comparar variáveis iguais.

Ex.: var x = 1; var y = 1;

x == y; (verdadeiro)

Usamos != para comparar variáveis desiguais.

x != y; (falso)



# Arrays

Trabalhamos com arrays usando os indicadores [i], onde 'i' corresponde ao número de elementos -1;

Ex.:

```
var lista = [2, 3, 7, 5, 4];
```

```
lista[5] = 8;
```





# if e else

Usamos **if** para utilizar a expressão SE então faça:

```
if (x >= 1){  
    código;  
}
```

Usamos **else** para utilizar a expressão SENÃO então faça:

```
else{  
    código;  
}
```



# While e for

A expressão de laço **while** ocorre enquanto a condição for verdadeira ou até que uma interrupção seja necessária.

```
var x = 0;
```

```
Ex.: while (x < 10) {
```

```
    console.log(x);
```

```
    x++;
```

```
}
```



# While e for

A expressão de laço **for** ocorre enquanto a condição for verdadeira ou até que uma interrupção seja necessária.

```
var x;
```

```
for (x = 0; x < 10; x++){
```

```
    console.log(x);
```

```
}
```



## Concatenar strings

A concatenação de strings ocorre por meio do operador “+”.

Ex. `var string1 = “Olá”;`

`var string2 = “Mundo”;`

`console.log(string1 + “ “ + string2);`

-> Saída == Olá Mundo



## Operadores lógicos (E, OU)

No caso do operador E a saída só é verdadeira quando todos os valores forem verdadeiros. Caso contrário, a saída será falsa. Em javascript é representado pelo símbolo &&.

No caso do operador OU a saída é verdadeira sempre que um dos valores forem verdadeiros. Caso contrário, a saída será falsa. Em javascript é representado pelo símbolo ||.



# Tipos de saída em JavaScript

Existem quatro tipos de saída em JavaScript:

- `console.log();`
- `alert();`
- `document.write();`
- `document.getElementById("").innerHTML = "";`



## **document.log()**

O uso da função “document.log();” é feita quando queremos que o valor seja mostrado apenas no console do nosso navegador.

Comumente para acessar o console do navegador usamos a tecla de atalho F12.

Ex. `console.log(“olá mundo”);`



## **alert();**

A função `alert()` é chamada quando queremos mostrar uma informação em uma janela pop-up, no meio superior da tela do navegador. As janelas da função `alert` aparecerão de maneira sequencial, enquanto houver algo para ser mostrado.

Ex.: `alert("olá mundo");`





## **document.write();**

Usamos a função `document.write()` para escrever diretamente no navegador.

Ex.: `document.write(x);`

-> imprime o valor de x.



## **document.getElementById()**

Usamos a função `document.getElementById()` quando queremos que nossa saída seja mostrada em um local específico do documento HTML. A função deve ter o Id colocado no arquivo HTML e a chamada da função no documento `javaScript`.



## **document.getElementById()**

No documento HTML:

```
<body>
```

```
<p id="texto"></p>
```

```
</body>
```

No documento javascript:

```
document.getElementById("texto").innerHTML = "Olá Mundo";
```