



# Desenvolvimento em JavaScript

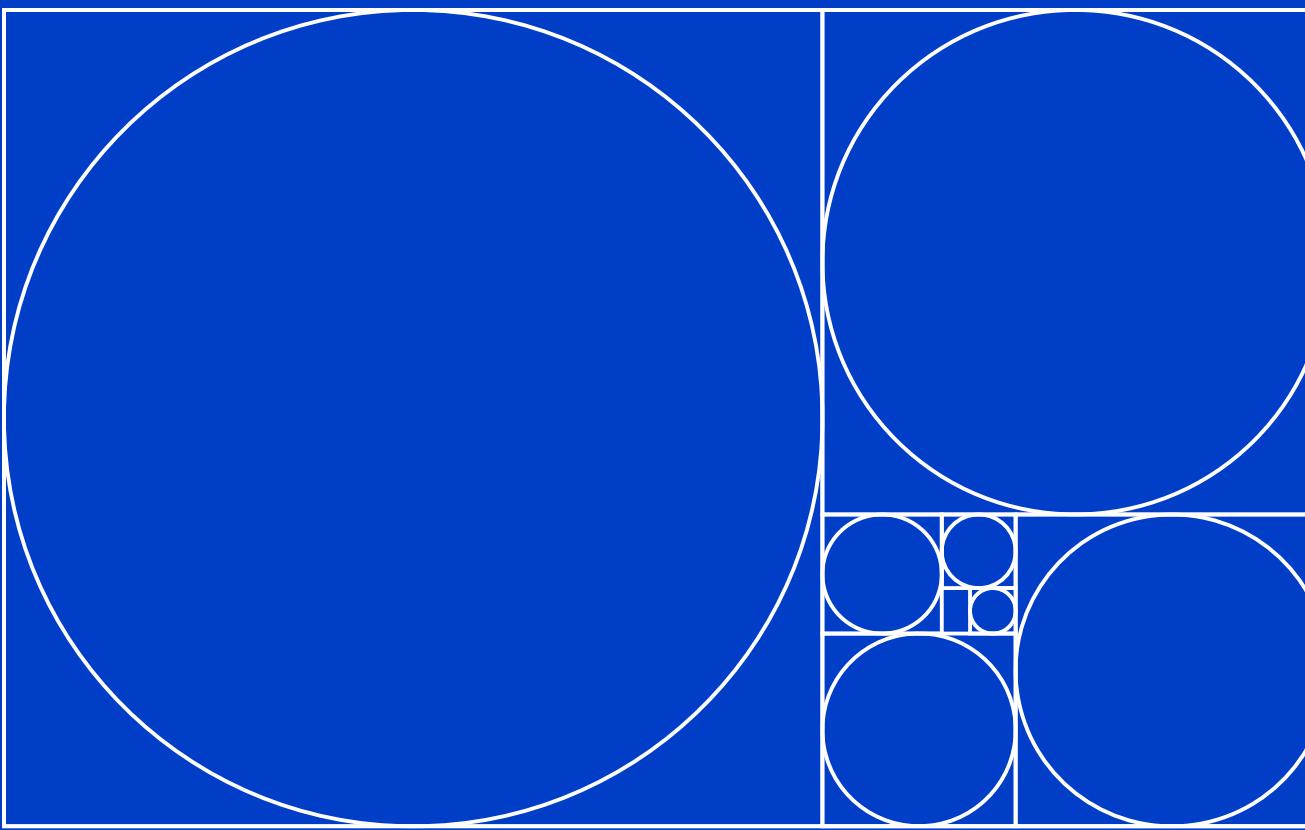
1. Características de uma  
linguagem de scripting

**dezembro de 2025**

# Hoje

- 1. Apresentações
  - Sobre mim
  - Sobre vocês
- 2. O papel do JS na web
- 3. *Scripting Language*
- 4. Integração de JavaScript
- 5. O nosso primeiro script

# 1. Apresentações



# Bruno Brito

**42 anos / Lisboa**

Licenciado em Publicidade e Marketing

Bootcamp de programação

Certificações Cambridge School, Microsoft  
e Google

+15 anos de experiência em Marketing

+10 anos de experiência em Programação

[/in/brunobrito](https://www.linkedin.com/in/brunobrito)



# Experiência Profissional



# Em 2025



Marketing Manager e  
Programador Front-end



Docente de Marketing Digital e  
Web Marketing

# Bruno Brito

Full-stack Marketer e  
Programador Web (Freelance)

A screenshot of a Git commit history interface. The left sidebar shows branches: main (HEAD), feature, admin-dashboard, order-processing, search-functional..., hotfix, and production. The main area displays commits for the 'main' branch in June 2025. The commits are color-coded by author: Alex Johnson (blue), Bruno Martinez (green), and Michal Kowalski (orange).

Author	Date	Commit Hash	Message
Alex Johnson	24/06/2025	8236e1df	Merge security hotfix
Alex Johnson	24/06/2025	85544bf1	Add security headers to prevent c...
Bruno Martinez	24/06/2025	f3ad9faf	Add validation to Part model
Alex Johnson	24/06/2025	dc70f86d	Merge admin dashboard
Alex Johnson	24/06/2025	1f615a3f	Merge order processing system
Alex Johnson	24/06/2025	4067940c	Merge admin dashboard route...
Bruno Martinez	24/06/2025	578dd1f1	Add parts resource routes
Alex Johnson	24/06/2025	67bace04	Merge admin dashboard styl...
Michal Kowalski	24/06/2025	72a16176	Add admin users routes
Alex Johnson	24/06/2025	7498a513	Merge shopping cart funct...

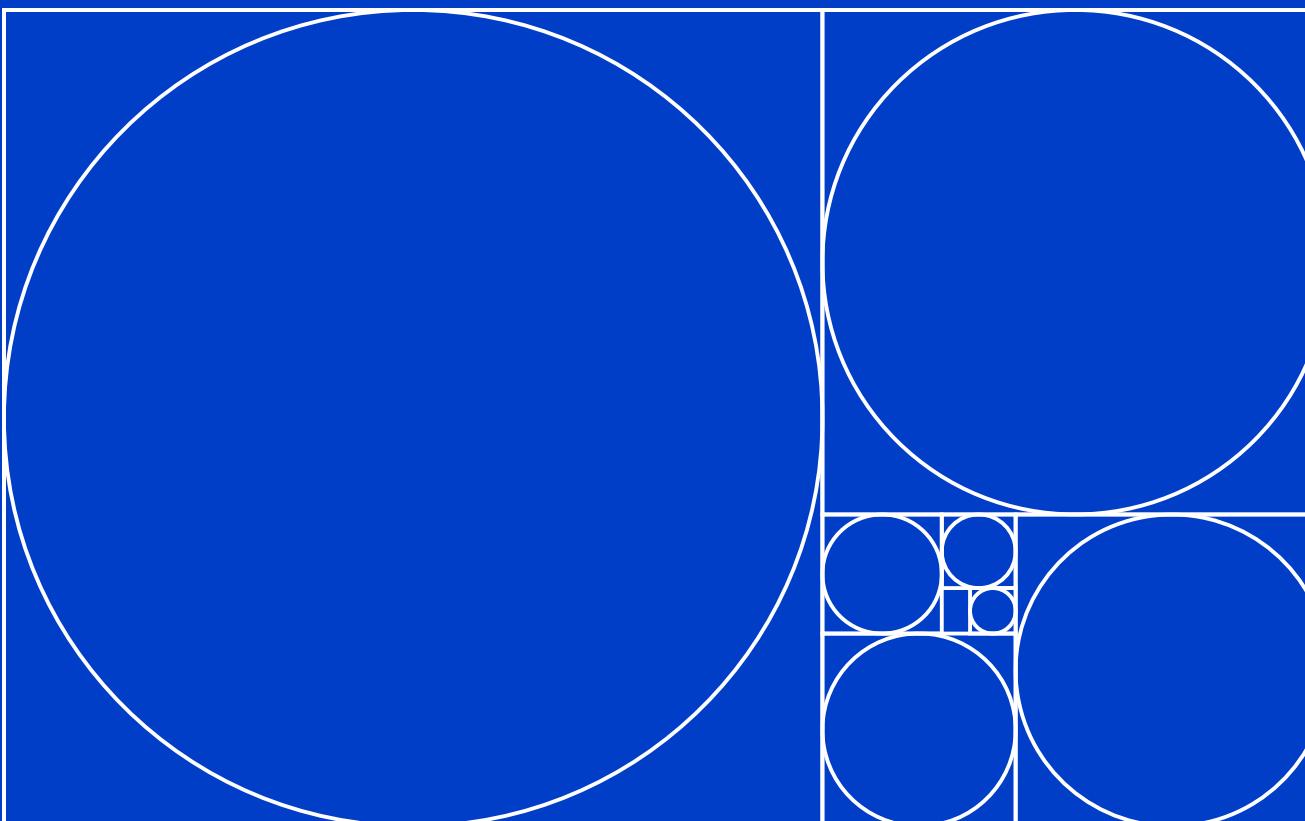
# Hobbies



# Over to you...

- A tua história
- O que te trouxe cá
- Expectativas quanto a este curso
- Uma curiosidade sobre ti!

## 2. O papel do JS na Web



**HTML****CSS****JS**

# HTML



O conteúdo  
da página

# CSS



O “look”  
da página

# JS

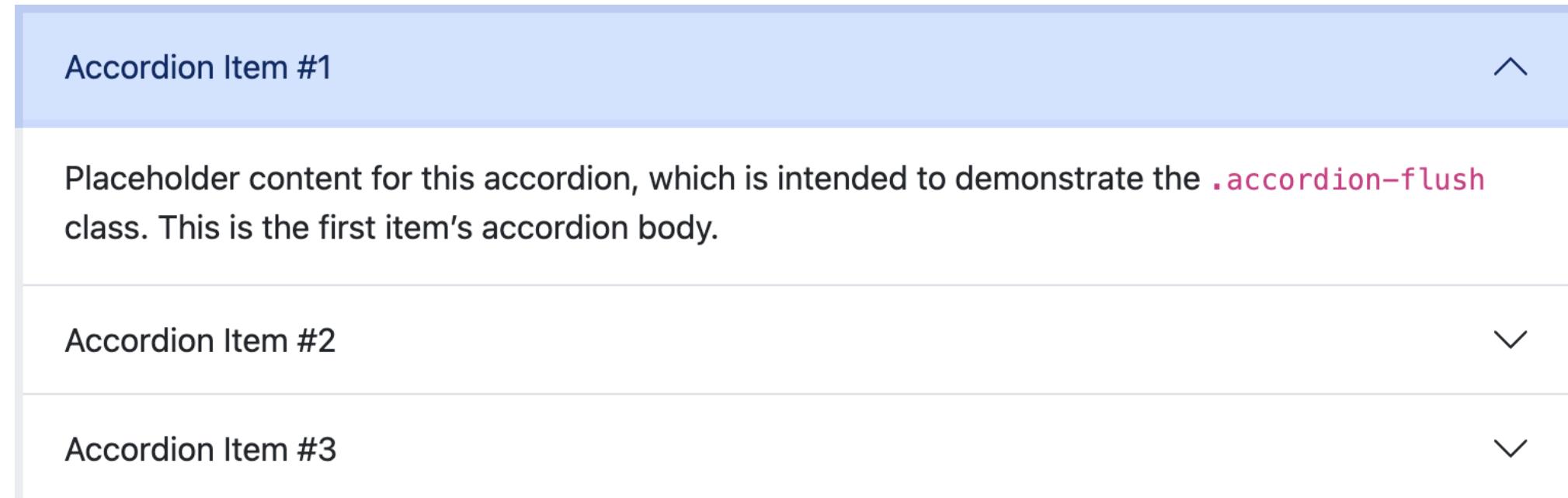


Como  
interagimos  
com a página

# Exemplo: Carousel



# Exemplo: Accordion



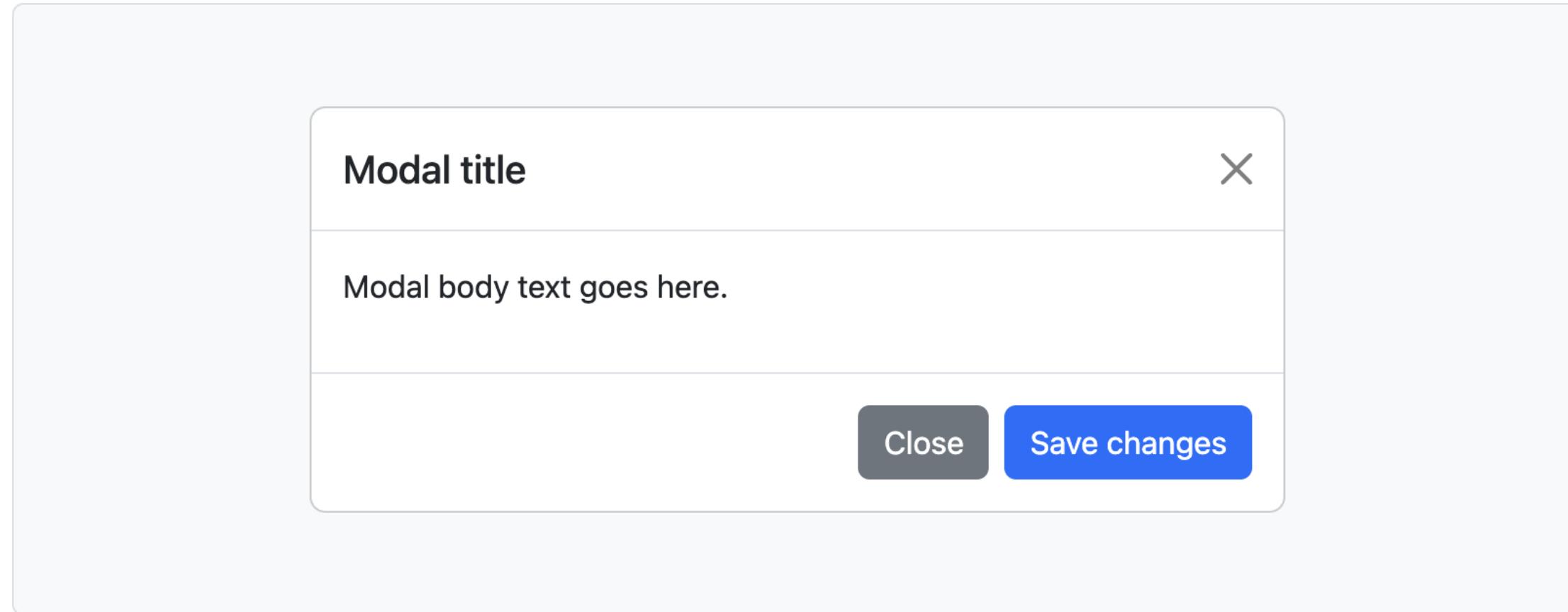
Accordion Item #1

Placeholder content for this accordion, which is intended to demonstrate the `.accordion-flush` class. This is the first item's accordion body.

Accordion Item #2

Accordion Item #3

# Exemplo: Modal



The screenshot shows the homepage of the Worten website. At the top, there is a navigation bar with links for 'Menu', 'Promoções', 'Worten Resolve' (which is highlighted in green), 'Prendas de Natal', 'Cupões para ti', 'Worten Life', 'Blog', and 'App Worten'. A search bar with the placeholder 'O que estás à procura?' is also present. On the right side of the header, there are icons for 'Olá!' and 'Iniciar Sessão' (Logout) along with a shopping cart icon.

A prominent white cookie consent dialog box is centered over the page. It features a blue circular icon with three white dots. The text inside the dialog reads:

**Utilizamos cookies**

Utilizamos cookies pois queremos que tenhas uma experiência de compra personalizada, com anúncios e promoções adequados aos teus interesses, a possibilidade de consultar as avaliações de produtos e também para que possamos analisar o tráfego do nosso site.

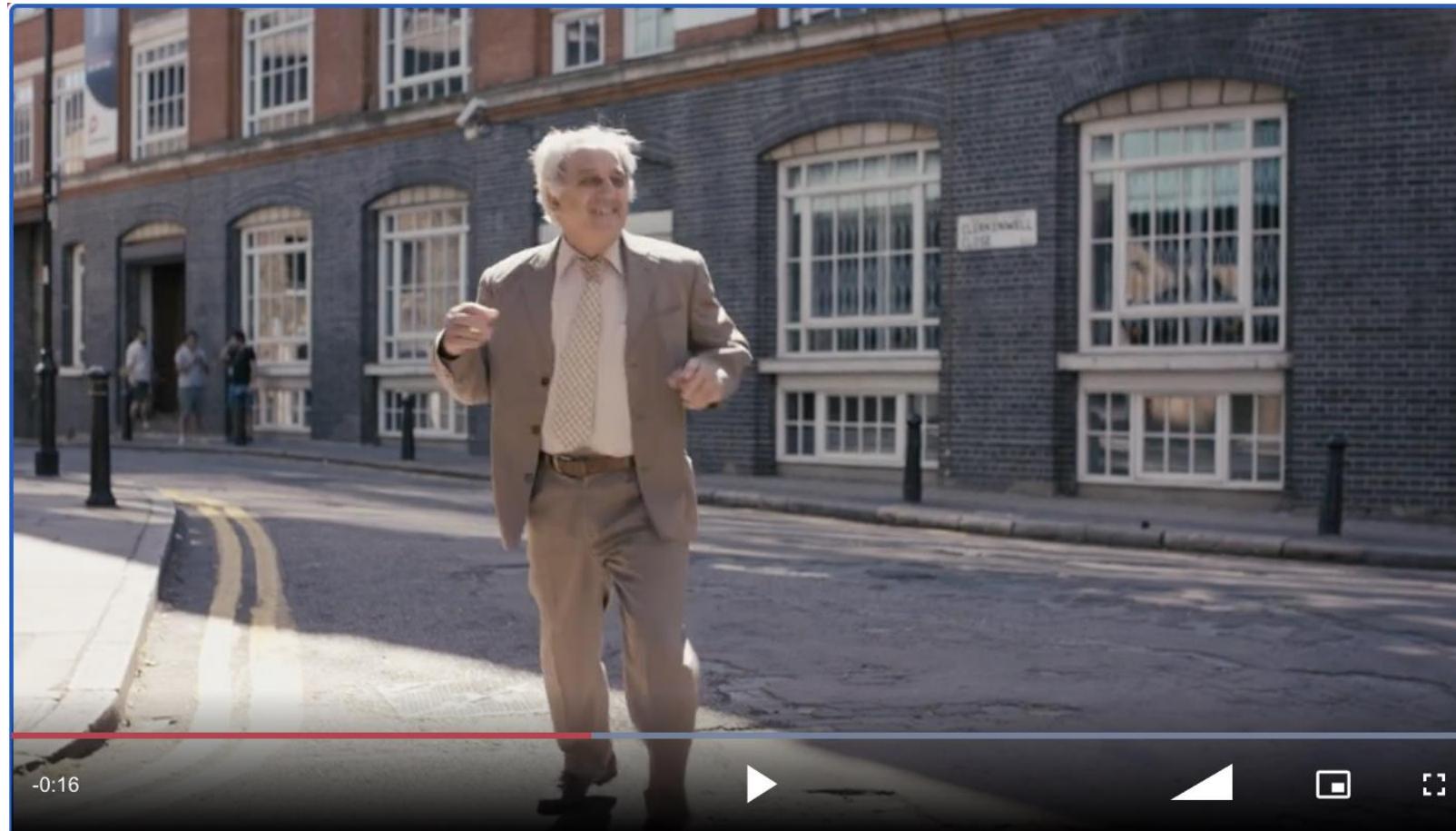
Podes alterar as tuas definições de cookies a qualquer altura.

At the bottom of the dialog, there are two buttons: 'Rejeitar cookies' (Reject cookies) and 'ACEITAR COOKIES' (Accept cookies). The 'ACEITAR COOKIES' button is highlighted with a dark background.

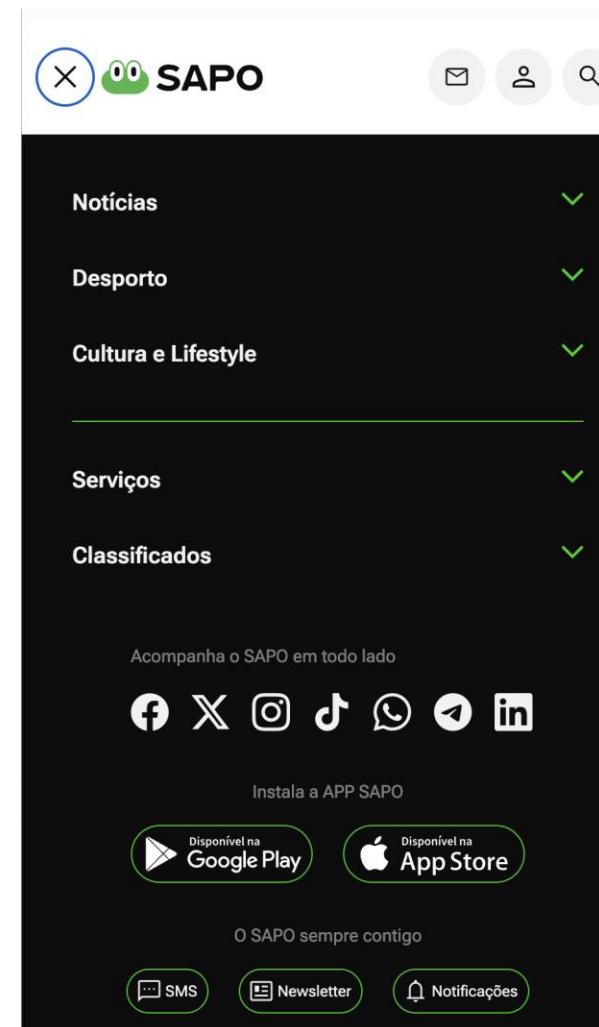
In the background, the website's main content is visible, including a banner for 'Até 30% desconto direto em Aquecedores e Ar Condicionado' (Up to 30% discount directly on heaters and air conditioners) and another for 'Chromebook Days! O portátil com a IA da Google desde 249€' (Chromebook Days! The portable with Google's AI from 249€).

Below the main content, there are several categories represented by icons and labels: Casa e Decoração (Household & Decoration), Desporto e Outdoor (Sports & Outdoor), Jardim (Garden), Perfumaria e Cosmética (Cosmetics), Jogos e Brinquedos (Games & Toys), Bebé (Baby), Moda (Fashion), Acessórios para Carros (Car accessories), Animais de estimação (Pet animals), and Supermercado (Supermarket). There is also a small 'APP' icon in the bottom right corner.

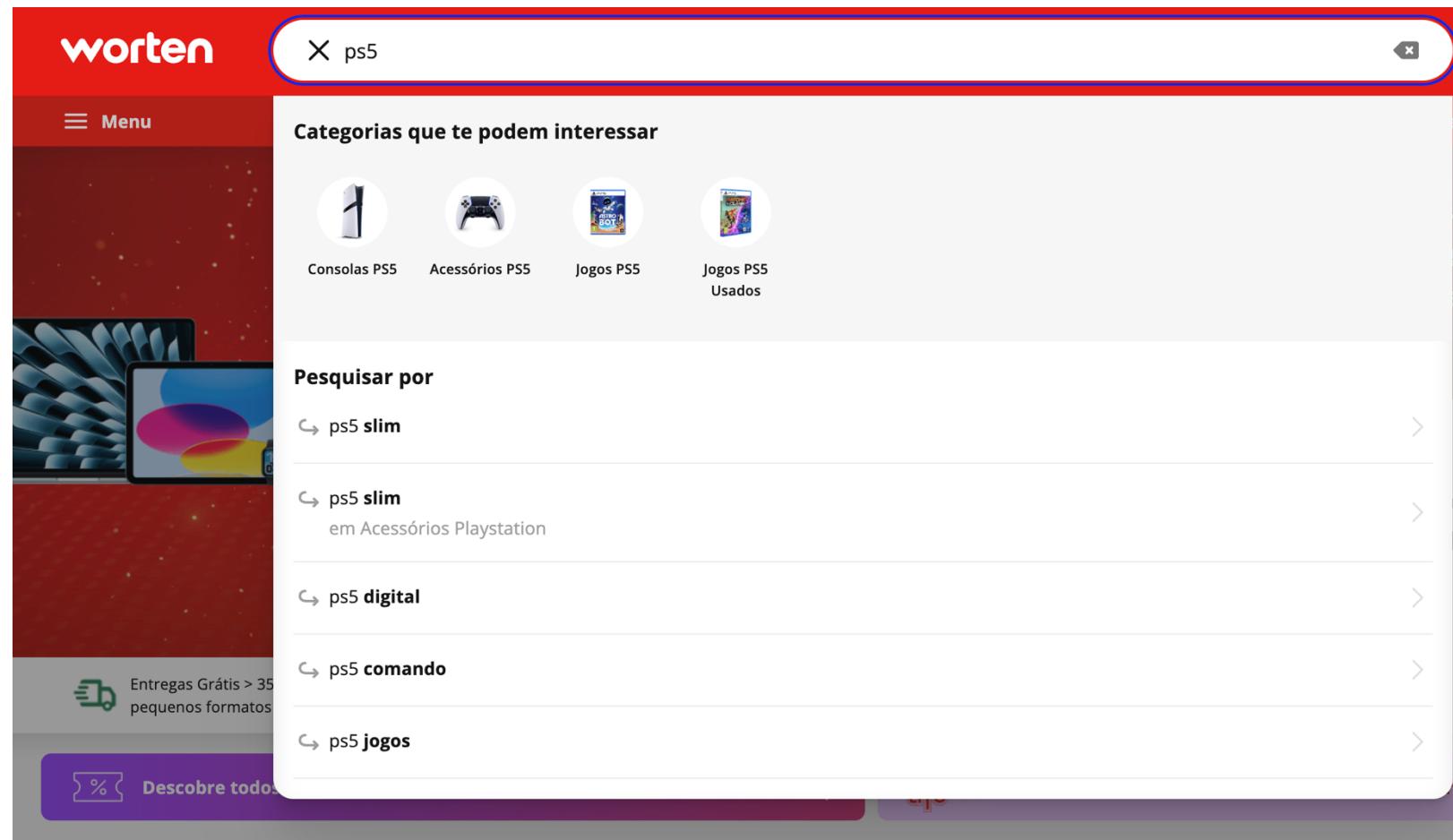
# Exemplo: Leitor de vídeo



# Exemplo: Hamburguer menu



# Exemplo: Filtragem de resultados



# Exemplo: Aplicações Web

December 2025



**8 Dec**

Mon

Hover to complete

**9 Dec**

Tue

Choose from colors

Click to edit

Hope you like 😊

Drag to other day

**10 Dec**

Wed

That's all!

**11 Dec**

Thu

**12 Dec**

Fri

Remember to save ✎

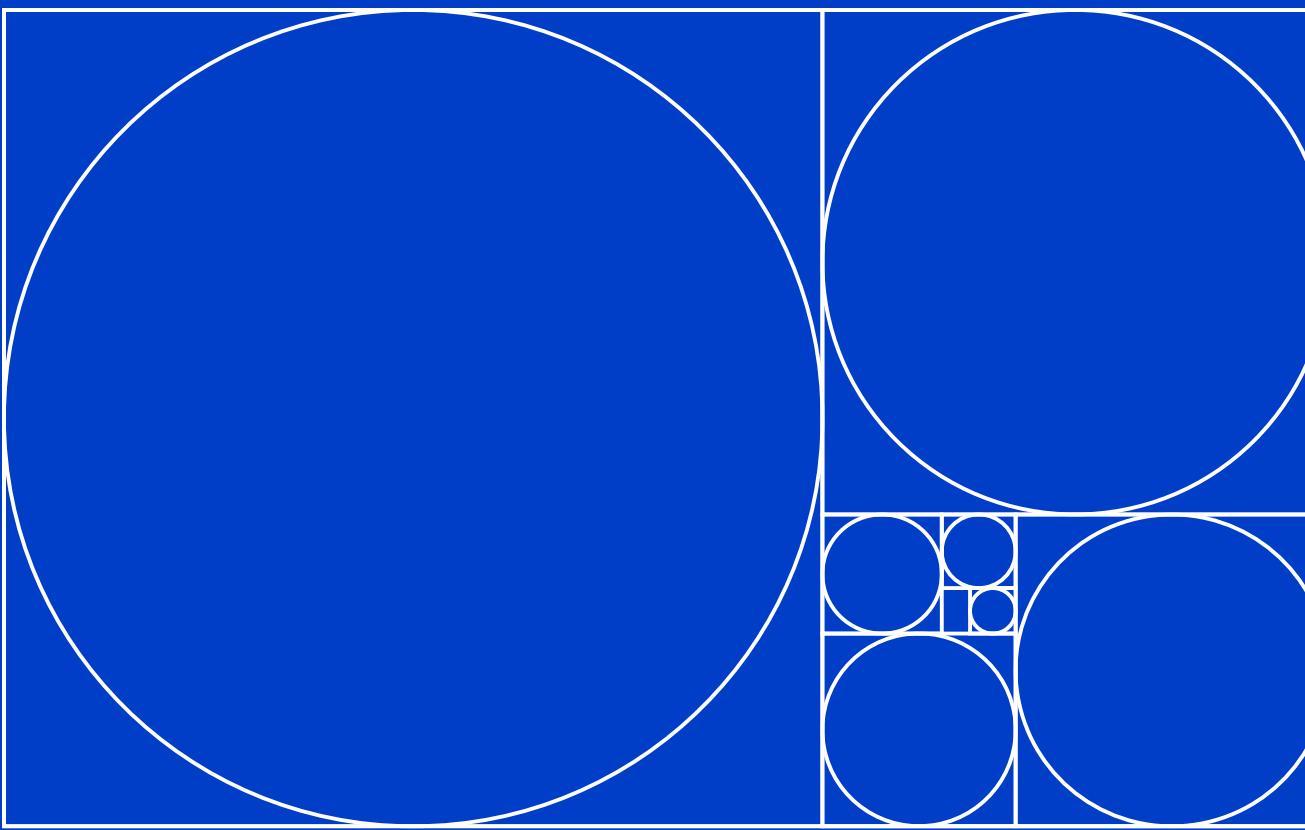
**13 Dec**

Sat

# Fora do browser...

- Aplicações para Android e iOS (ex: [React Native](#))
- Aplicações Desktop (ex: [Electron](#))
- Jogos (ex: [Three.js](#))
- Servidor (ex: [Node.js](#))
- Inteligência Artificial (ex: [TensorFlow.js](#))

# 3. Scripting Language



# Scripting Language

- **1. Executa instruções passo a passo, em tempo real.**

Quando o browser encontra um <script>, ele:

- Lê linha a linha,
- Interpreta,
- Executa no momento.

# Scripting Language

- **2. Não precisa de compilação prévia (interpretação) como, por exemplo, C, C++, Java ou Rust.**

Não existe um processo de “build” e não é necessário gerar um ficheiro executável primeiro.

O browser faz tudo automaticamente enquanto corre, tornando a linguagem ótima para rapidamente prototipar e testar.

# Scripting Language

- **3. Permite automatizar tarefas, reagir a eventos e manipular conteúdo dinâmico.**

No browser, podemos:

- manipular HTML acedendo ao DOM (Document Object Model);
- reagir a eventos (cliques, scrolls, movimentos do rato, teclas pressionadas...);
- controlar ficheiros multimédia (áudio ou vídeo);
- comunicar com APIs;
- gravar informação localmente.

# Scripting Language

- **4. Corre dentro de um ambiente (*browser, Node.js*).**

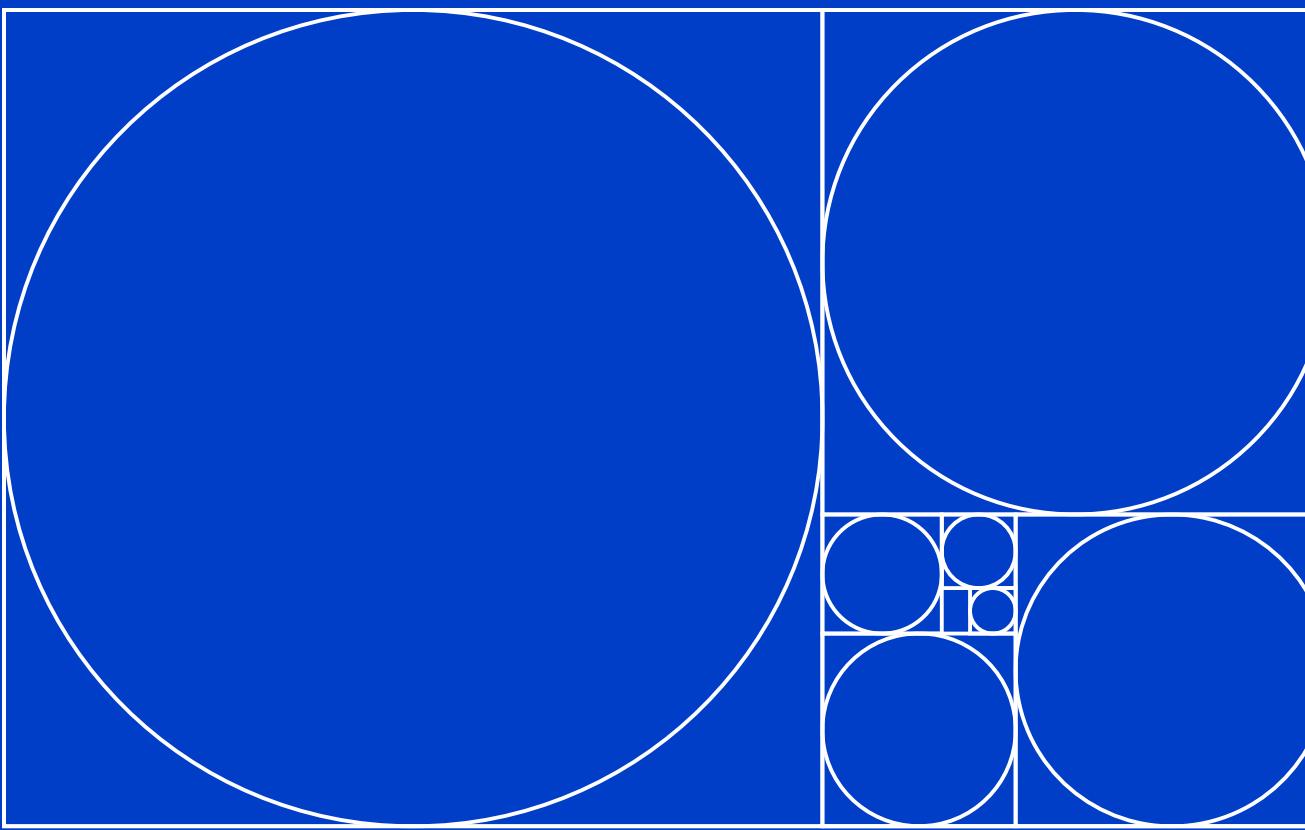
JavaScript precisa sempre de um ambiente que o acolha e execute.

Com Node.js, podemos aceder a ficheiros do nosso computador, comunicar com servidores, e gerir processos de sistema.

# O que JS faz que HTML/CSS não fazem

- **Manipula o DOM:** cria, apaga ou altera elementos em tempo real.
- **Responde a eventos:** cliques, teclas, scroll, formulários.
- **Toma decisões:** condições, ciclos, variáveis, funções.
- **Comunica com servidores (APIs).**
- **Guarda dados localmente (localStorage).**
- **Controla multimédia e animações lógicas.**
- **Permite aplicações completas:** SPAs, jogos, dashboards...

# 4. Integração de JavaScript



# Opção 1: “inline”



```
<button onclick="alert('Hello World!')>Clica aqui</button>
```

# Opção 2: com a tag <script>



```
<!DOCTYPE html>
<html>
<body>

<script>
    alert("Hello World!");
</script>

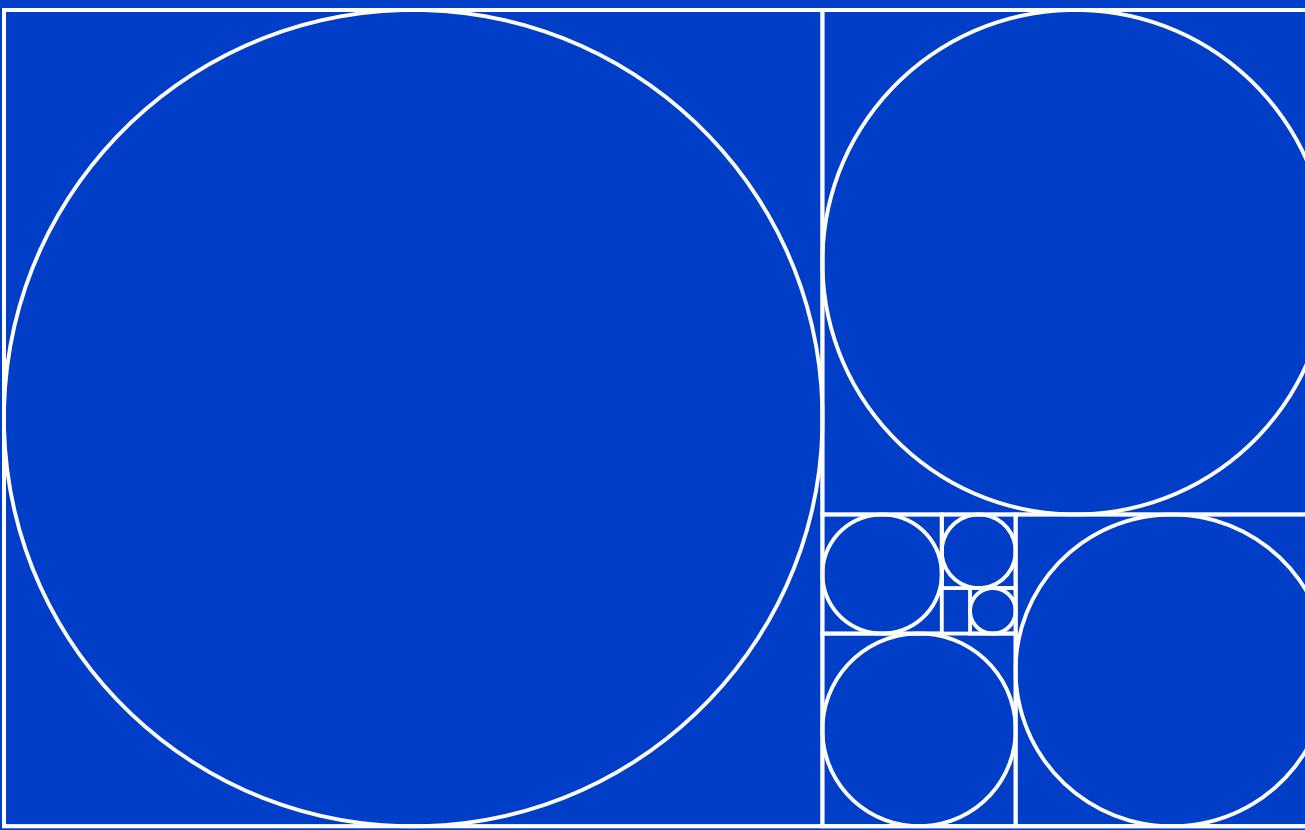
</body>
</html>
```

# Opção 3: ficheiro externo

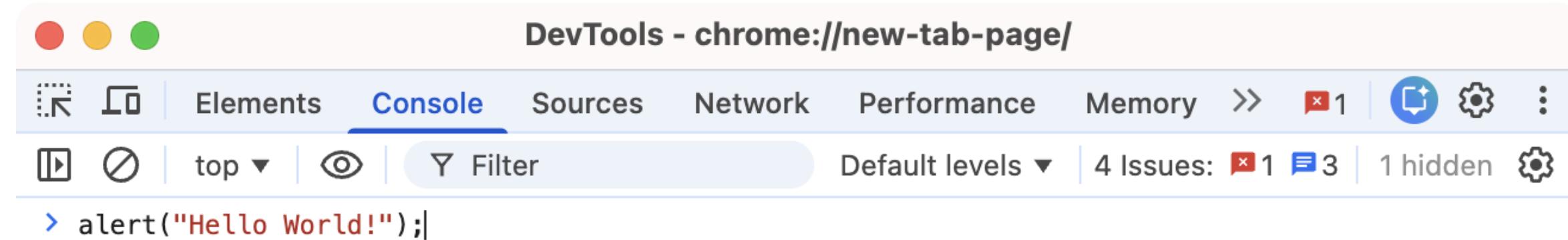
```
● ● ●  
<!DOCTYPE html>  
<html>  
<body>  
  
<script src="script.js"></script>  
  
</body>  
</html>
```

```
● ● ●  
//script.js  
alert("Hello World!");
```

# 5. O nosso primeiro script



# Um olá no browser



# Um título modificado

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8">
5  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  <title>Um olá com JavaScript</title>
7  </head>
8  <body>
9  <h1></h1>
10 <script>
11  var name = "Bruno";
12  var greeting = "Hello " + name + "!";
13  var title = document.querySelector('h1');
14  title.innerHTML = greeting;
15 </script>
16 </body>
17 </html>
```



Faculdade de Design,  
Tecnologia e Comunicação  
ue Universidade Europeia

# Obrigado!

Até à próxima aula! 🙌