

Objetivo

Crie um site multi-página responsivo usando Flexbox e/ou Bootstrap 5

Especificação técnica

Estrutura básica

- Crie uma página **portfólio**, uma página de **contato** e páginas de **login e registro**. As páginas “portfólio” e “contato” devem ter **cabeçalho** e **navegação** responsivos com links para essas duas páginas e um logo da marca que ao clicar nele, leva para a página de **portfólio**
- Inclua um **menu dropdown** em "Portfólio" com as categorias de produtos que o site oferece e links para a página de login e registro
- Ao redimensionar a página, o menu deve se tornar **vertical**
- As páginas devem ter um **rodapé** responsivo com links para a suas redes sociais

Página de Portfólio (portfolio.html):

- Crie uma seção de portfólio para exibir pelo menos 6 projetos envoltos com *flex-wrap* e *media-query*
- Cada projeto deve ser apresentado como um cartão com imagem, título, descrição e um botão "Contratar Projeto" que deve levar para a página de contato. Estilize esses cards para ficarem bem visíveis

Página de Contato (contact.html):

- Crie um formulário de contato com campos para nome, e-mail, projeto a ser contratado e mensagem
- O “projeto a ser contratado” deve vir a partir de uma lista pré-definida
- Implemente validação no lado do cliente usando JavaScript:
- Valide se todos os campos estão preenchidos.
- Certifique-se de que o campo de e-mail está corretamente formatado.
- Estilize a seção de formulário para ser visualmente coesa com o resto do site

Formulário das páginas de login e registro

O formulário da página de **login** deve conter:

- Email
- Senha
- Botão para validar o formulário e redirecionar para a página principal (usar button type “button”)
- Link para redirecionar para a página de registro

O formulário da página de **registro** deve conter:

- Nome
- Data de nascimento
- Email
- Senha
- Input para repetir a senha
- Botão para validar o formulário e redirecionar para a página principal
- Link para redirecionar para a página de login

Validação dos formulários de login e registro

O formulário de **login** deve:

- Ter o valor do endereço de email ser igual a admin@admin.com
- Ter o valor da senha como “senha”
- Após ter esses valores validados, redirecione para a página principal

O formulário de **registro** deve:

- Ter o valor do nome como sendo maior do que 3 caracteres
- Ter o valor da senha e do repetir senha iguais
- Após ter esses valores validados, redirecione para a página principal

Estilização

Siga os padrões de estilos estudados em sala e aplique em todas as páginas, como por exemplo:

- Responsividade
- Pseudo-classes
- Botões grandes
- Bordas ao redor de cards, inputs, etc

Envio

- Ao finalizar, envie os arquivos para henriquedelegreggo@hotmail.com

Observações:

- A prova é individual
- Permitido pesquisar no material enviado e nos seus códigos
- O prazo para enviar a prova termina às 09:45

Questões de múltipla escolha:

Coloque as respostas como comentário do email

- 1) O que acontecerá com os links de navegação ao serem visualizados em um dispositivo com largura de tela de 450px?

```
<nav class="navbar">
  <ul class="nav-links">
    <li><a href="#">Home</a></li>
    <li><a href="#">Sobre</a></li>
    <li><a href="#">Serviços</a></li>
    <li><a href="#">Contato</a></li>
  </ul>
</nav>
<style>
  .nav-links {
    display: flex;
    flex-direction: row;
    list-style-type: none;
  }

  @media (max-width: 600px) {
    .nav-links {
      flex-direction: column;
    }
  }

  @media (max-width: 500px) {
    .nav-links {
      display: none;
    }
  }
</style>
```

- a) Os links serão exibidos verticalmente, pois a flex-direction mudará para column em telas menores
- b) Os links desaparecerão, pois não há estilo específico aplicado para telas abaixo de 600px.
- c) Os links continuarão a ser exibidos horizontalmente, já que a flex-direction é row por padrão.
- d) Os links desaparecerão, pois é aplicado display: none

2) Onde a caixa 2 aparecerá em relação à caixa 1 e ao container?

```
<div class="container">
  <div class="box1">Caixa 1</div>
  <div class="box2">Caixa 2</div>
</div>

<style>
  .container {
    position: relative;
    width: 200px;
    height: 200px;
    background-color: lightgrey;
  }

  .box1 {
    position: absolute;
    top: 20px;
    left: 20px;
    background-color: red;
    width: 80px;
    height: 80px;
  }

  .box2 {
    position: absolute;
    bottom: 10px;
    right: 10px;
    background-color: blue;
    width: 80px;
    height: 80px;
  }
</style>
```

- a) No canto superior esquerdo do container, sobrepondo a caixa 1
- b) No canto inferior direito do container, a 10px de ambas as bordas
- c) No canto superior direito do container, ao lado da caixa 1
- d) Diretamente abaixo da caixa1, alinhada ao lado esquerdo do container

3) Qual caixa será exibida sobre as outras, e por quê?

```
<div class="container">
  <div class="box1">Box 1</div>
  <div class="box2">Box 2</div>
  <div class="box3">Box 3</div>
</div>

<style>
  .container {
    position: relative;
    width: 300px;
    height: 200px;
  }

  .box1 {
    position: absolute;
    top: 10px;
    left: 10px;
    width: 100px;
    height: 100px;
    z-index: 1;
  }

  .box2 {
    position: absolute;
    top: 30px;
    left: 30px;
    width: 100px;
    height: 100px;
    z-index: 2;
  }

  .box3 {
    position: absolute;
    top: 50px;
    left: 50px;
    width: 100px;
    height: 100px;
    z-index: 0;
  }
</style>
```

- a) Caixa 3 porque tem o maior valor de top
- b) Caixa 2 porque tem um z-index maior que as outras
- c) Caixa 3 porque tem o menor z-index, então será sobreposta
- d) Caixa 1 porque aparece primeiro na estrutura HTML