

# Mobile-first UI com React [24E3\_1]

[Página inicial do site](#) ▶ [Meus cursos](#) ▶ [Mobile-first UI com React \[24E3\\_1\]](#) ▶ [Etapa 6](#) ▶ [Teste de Performance - TP3 \[Obrigatório\]](#)

## TESTE DE PERFORMANCE - TP3 [OBRIGATÓRIO]

---

Olá, Samuel,

Chegamos em mais uma etapa de preparação! A cada Teste de Performance (TP) você terá a oportunidade de praticar os conhecimentos adquiridos e receber feedbacks relevantes para o seu aprendizado.

Esse é o terceiro teste de performance (TP) da disciplina regular **Mobile-First UI com React**, do Bloco de **Desenvolvimento Front-end com Frameworks**. É a partir daqui que você dará seus primeiros passos em sua formação.

Nesse TP, temos como objetivo desenvolver a segunda e terceira competência prevista para nossa disciplina: *Criar páginas web responsivas com CSS3* e *Estilizar componentes React com CSS3* respectivamente. Para tal, você irá desenvolver algumas tarefas que irão passar por todo o conteúdo visto na primeira e segunda etapas do curso.

Tarefa 1

Página com Menu Responsivo

Utilizando a ferramenta **CodeSandbox** e o template **HTML, CSS e JS**, crie uma página simples contendo um menu com, ao menos, cinco elementos: uma imagem com o logo da página, um ícone para o usuário interagir com o meu e outros elementos como opções do mesmo, por exemplo: Produtos, Serviços e Contato.

Implemente o que for necessário para que o menu da página fique responsivo.

Quando não houver espaço suficiente, o menu deve ser no estilo expande/colapsa no topo da página.

Quando houver espaço suficiente para que todos os elementos do menu estejam visíveis para o usuário, o menu deve ser do tipo horizontal. Também no topo da página.

- **Utilize a *tag* própria para uso de imagens na página *HTML*.**
- **Atribua à imagem uma classe referência em *CSS*.**
- **Não implemente códigos *CSS* no arquivo *HTML*.**
- **Pense Mobile-First.**

## Tarefa 2

### Página com Menu Responsivo Lateral

Utilizando a ferramenta **CodeSandbox** e o template **HTML, CSS e JS**, crie uma página simples contendo um menu com, ao menos, 8 elementos: uma imagem com o logo da página, um ícone para o usuário interagir com o meu e outros elementos como opções do mesmo, por exemplo: Perfil, Postagens, Amigos, Fotos, Vídeos e Configurações.

Implemente o que for necessário para que o menu da página fique responsivo.

Quando não houver espaço suficiente, o menu deve ser no estilo expande/colapsa, porém aparecendo na lateral direita da página.

Quando houver espaço suficiente para que todos os elementos do menu estejam visíveis para o usuário, o menu deve ser do tipo horizontal, no topo da página.

- Utilize a **tag** própria para uso de imagens na página **HTML**.
- Atribua à imagem uma classe referência em **CSS**.
- Não implemente códigos **CSS** no arquivo **HTML**.
- Pense **Mobile-First**.

### Tarefa 3

#### Página com Menu Responsivo do tipo Drawer navigation

Utilizando a ferramenta **CodeSandbox** e o template **HTML, CSS e JS**, crie uma página simples contendo um menu com, ao menos, 8 elementos: uma imagem com o logo da página, um ícone para o usuário interagir com o meu e outros elementos como opções do mesmo, por exemplo: Perfil, Postagens, Amigos, Fotos, Vídeos e Configurações.

Implemente o que for necessário para que o menu da página fique responsivo.

Quando não houver espaço suficiente, o menu deve ser no estilo expande/colapsa, no estilo vertical, aparecendo na lateral esquerda da página.

Quando houver espaço suficiente para que todos os elementos do menu estejam visíveis para o usuário, o menu deve ser do tipo vertical, no lado esquerdo da página.

- Utilize a **tag** própria para uso de imagens na página **HTML**.
- Atribua à imagem uma classe referência em **CSS**.

- **Não implemente códigos CSS no arquivo *HTML*.**
- **Pense Mobile-First.**

#### Tarefa 4

#### Página com Menu Responsivo do tipo Bottom Navigation

Utilizando a ferramenta ***CodeSandbox*** e o template **HTML, CSS e JS**, crie uma página simples contendo um menu com, ao menos, 5 elementos: uma imagem com o logo da página, um ícone para o usuário interagir com o meu e outros elementos como opções do mesmo, por exemplo: Perfil, Postagens e Configurações.

Implemente o que for necessário para que o menu da página fique responsivo.

Quando não houver espaço suficiente, o menu deve ser no estilo expande/colapsa, também no estilo horizontal, aparecendo na base da página.

Quando houver espaço suficiente para que todos os elementos do menu estejam visíveis para o usuário, o menu deve ser do tipo horizontal, na base da página.

- **Utilize a *tag* própria para uso de imagens na página *HTML*.**
- **Atribua à imagem uma classe referência em CSS.**
- **Não implemente códigos CSS no arquivo *HTML*.**
- **Pense Mobile-First.**

#### Tarefa 5

#### Componente React com Estilização por Tags

Utilizando a ferramenta **CodeSandbox** e o template **React**, crie uma página simples.

Esta página deve conter:

- Título de segundo nível com o texto: “Desenvolvimento Front-End com Frameworks”.
- Título de terceiro nível com o texto: “Mobile-First UI com React”.
- Título de quarto nível com o texto: “Diferença entre responsividade e adaptabilidade”.
- Um texto, devendo ser separado em vários parágrafos, explicando a diferença apontada no texto anterior.

Estilize a página para que:

- O *h2* fique com a fonte maior que o *h3*, que terá a fonte maior que o *h4*, que terá a fonte maior que o *p*. Cada um desses componentes deve ter um tamanho de fonte mínimo, ideal e máximo.
- Todos os cabeçalhos devem ter o texto centralizado.
- Todos os parágrafos devem ter o texto justificado e a primeira linha deve estar avançada para a direita. Além disso, deve-se haver um espaçamento interno (padding) na vertical de 0 e na horizontal de 20px.

*Faça isso utilizando apenas estilização direta de tags, exemplo:*

Arquivo CSS:

```
div { /* estilo */ }
```

- **Pense Mobile-First.**

## Tarefa 6

### Componente React com Estilização por Class

Utilizando a ferramenta **CodeSandbox** e o template **React**, crie uma página simples.

Esta página deve conter:

- Título de segundo nível com o texto: “Desenvolvimento Front-End com Frameworks”.
- Título de terceiro nível com o texto: “Mobile-First UI com React”.
- Título de quarto nível com o texto: “Conceitos de Usabilidade”.
- Um texto, devendo ser separado em vários parágrafos, explicando a diferença apontada no texto anterior.

Estilize a página para que:

- O *h2* fique com a fonte maior que o *h3*, que terá a fonte maior que o *h4*, que terá a fonte maior que o *p*. Cada um desses componentes deve ter um tamanho de fonte mínimo, ideal e máximo.
- Todos os cabeçalhos devem ter o texto centralizado.
- Todos os parágrafos devem ter o texto justificado e a primeira linha deve estar avançada para a direita. Além disso, deve-se haver um espaçamento interno (padding) na vertical de 0 e na horizontal de 20px.

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

- **Pense Mobile-First.**

## Tarefa 7

### Componente Card React com Estilização por Class

Utilizando a ferramenta **CodeSandbox** e o template **React**, crie um componente com estilo *card*, com bordas arredondadas e com efeito de elevação (sombra).

Para ele (o componente), o programador deve passar uma imagem, um título, uma descrição e um rodapé (ou seja, este componente deve esperar que lhe sejam passados parâmetros para que ele possa ser construído).

Este *card* deve exibir a imagem na lateral esquerda dele.

O **título**, a **descrição** e o **rodapé** devem aparecer à direita do card, alinhados verticalmente.

**Um exemplo deste card pode ser visto na imagem a seguir:**



*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

- **Pense Mobile-First.**

Utilizando a ferramenta **CodeSandbox** e o template **React**, crie um componente com estilo *card*, com bordas arredondadas e com efeito de elevação (sombra).

Para ele (o componente), o programador deve passar uma imagem, um título, uma descrição e um rodapé (ou seja, este componente deve esperar que lhe sejam passados parâmetros para que ele possa ser construído).

Este *card* deve exibir a imagem no topo dele.

O **título**, a **descrição** e o **rodapé** devem aparecer na parte de baixo do card, alinhados verticalmente.

**Um exemplo deste card pode ser visto na imagem a seguir:**





### **Instituto Infnet**

O Infnet é um Instituto de Tecnologia fundado em 1994 que tornou-se referência na formação de profissionais em Tecnologia da Informação, Negócios e para a Indústria Criativa.

Rua São José, 90 – Centro, Rio de Janeiro – RJ, 20010-020

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

- **Pense Mobile-First.**

Utilizando a ferramenta **CodeSandbox** e o template **React**, crie um componente que é um menu.

Este menu deve ser do tipo *expande/colapsa*, possuir um elemento sempre visível que é o logo ou a marca do App e um outro elemento, também sempre visível, que é o botão para expandir/colapsar o menu.

Os outros itens do menu, a saber: **Home**, **Serviços**, **Produtos**, **Sobre nós** e **Contatos**, ficarão escondidos no menu.

Caso o dispositivo seja capaz de exibir todos os itens do menu na página (ou seja, caso ele tenha uma largura suficiente para isso), então o menu deve se adaptar a esta condição e se apresentar em um modelo de menu horizontal, exibindo todos os itens, sem a mecânica de expandir/colapsar. *Neste caso, o elemento que permite expandir/colapsar o menu, deve estar invisível.*

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

- **Pense Mobile-First.**

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma página que represente a **sua rede social**.

Esta página deve conter:

- Um layout com grid que contenha um cabeçalho, um menu principal, um feed de postagens de amigos, uma área com a relação de amigos e uma área com a relação de sugestões de amizade.
- As postagens de amigos devem conter uma imagem, um título, um texto, uma data de publicação, o autor da postagem, a quantidade de curtidas, a quantidade de compartilhamentos e uma lista de comentários.
- A relação de amigos deve apresentar a foto, o nome e a quantidade de amigos em comum.
- A relação de sugestões deve apresentar a foto, o nome e um amigo em comum entre você e a sugestão apresentada.
- O layout e o estilo da página ficam a seu critério.

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

Talvez você já tenha feito este trabalho anteriormente, porém agora você tem uma outra condição: todos os elementos da sua página devem ser componentes React e todos os elementos que se repetem na mesma, como a relação de amigos, devem ser criados pela reutilização de um componente genérico/personalizável.

**Combine *flexbox* e *grid layout*.**

- **Pense Mobile-First.**

## Tarefa 11

### Página Responsiva Detalhes do Produto

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma página apresentando os detalhes de um produto em uma plataforma de e-commerce.

Nesta página você deve apresentar:

- Uma imagem do produto, com opção para visualizar outras imagens do mesmo produto;
- O nome, o valor, a descrição, e as especificações do produto;
- Uma lista de produtos relacionados apresentando uma imagem, o nome e o valor dos produtos;
- Uma lista de comentários de outros usuários apresentando o nome, a data e o comentário.

Esta página deve ser responsiva, de modo que, quando o usuário estiver em um dispositivo maior, o layout deve ser reorganizado para apresentar os dados de uma melhor forma - podendo até apresentar mais informações.

Esta página deve ter um menu com, no mínimo, 4 itens, que também é responsivo. Em caso de não haver espaço suficiente, o menu deve funcionar em modo *expande - colapsa*.

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

Talvez você já tenha feito este trabalho anteriormente, porém agora você tem uma outra condição: todos os elementos da sua página devem ser componentes React e todos os elementos que se repetem na mesma, como a relação de amigos, devem ser criados pela reutilização de um componente genérico/personalizável.

- **O menu não precisa estar funcional.**
- **Implemente estilos diferentes para o esquema de cores *light* e *dark*.**
- **Pense Mobile-first!**

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma página apresentando um vídeo de uma plataforma de vídeo.

Nesta página você deve apresentar:

- Um vídeo em destaque;
- O título e o autor do vídeo;
- um menu com opções para curtir, compartilhar, salvar e denunciar o vídeo (essas opções não precisam estar funcionais);
- A descrição do vídeo;
- Uma área com propaganda;
- Uma lista de vídeos relacionados apresentando uma capa, o nome e o autor do vídeo;
- Uma lista de comentários de outros usuários apresentando o nome, a data e o comentário.

Esta página deve ser responsiva, de modo que, quando o usuário estiver em um dispositivo maior, o layout deve ser reorganizado para apresentar os dados de uma melhor forma - podendo até apresentar mais informações.

Esta página deve ter um menu com, no mínimo, 4 itens, que também é responsivo. Em caso de não haver espaço suficiente, o menu deve funcionar em modo *expande - colapsa*.

*Faça isso utilizando apenas estilização por classes CSS, exemplo:*

Arquivo CSS:

```
.container { /* estilo */ }
```

Talvez você já tenha feito este trabalho anteriormente, porém agora você tem uma outra condição: todos os elementos da sua página devem ser componentes React e todos os elementos que se repetem na mesma, como a relação de amigos, devem ser criados pela reutilização de um componente genérico/personalizável.

- O menu não precisa estar funcional.
- Implemente estilos diferentes para o esquema de cores *light* e *dark*.
- **Pense Mobile-first!**

## Tarefa 13 App Lista de Tarefas com Categorias

Utilizando a plataforma **CodeSandbox** e o template **React**, implemente uma aplicação que simule uma lista de tarefas. O usuário pode inserir e remover novas tarefas.

A interface inicial apresenta uma relação de categorias de tarefas (por exemplo: mercado, farmácia, estudos).

Ao selecionar uma das categorias, a aplicação apresenta as tarefas que pertencem àquela categoria.

- **Use o layout que achar melhor!**
- **A aplicação deve estar componentizada.**
- Implemente estilos diferentes para o esquema de cores *light* e *dark*.
- **Pense Mobile-first!**

## Tarefa 14 App Quiz

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma aplicação que simule um Quiz em que há várias questões. Cada questão possui um enunciado e quatro opções. A aplicação deve conter ao menos 10 questões. Implemente um mecanismo de pontuação e exiba a pontuação do usuário ao final do Quiz.

- Use o layout que achar melhor!
- Não implemente navegação, use apenas gerenciamento de estado.
- Implemente estilos diferentes para o esquema de cores *light* e *dark*.
- Pense Mobile-first!

## Tarefa 15

### App Match

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma aplicação que apresente para o usuário, um item por vez, de uma relação de itens, por exemplo: títulos de filmes, títulos de músicas, títulos de jogos.

Para cada um dos itens, o usuário deverá informar se gosta ou não gosta do que lhe foi apresentado.

No momento em que ele desejar, ele pode ir para outra interface em que lhe será apresentado uma relação de outros usuários que tenham gostos parecidos com os seus, ou seja, que gostam e não das mesmas coisas e ele. Esta relação deve estar organizada por nível de afinidade.

- Use a técnica que desejar para calcular o nível de afinidade.
- Use o layout que achar melhor!
- A aplicação deve estar componentizada.
- Implemente estilos diferentes para o esquema de cores *light* e *dark*.
- Pense Mobile-first!

Utilizando a ferramenta **CodeSandbox** e o template **React**, implemente uma aplicação que relacione os pontos turísticos da sua cidade, região, estado ou país.

Nesta mesma interface, o usuário pode selecionar um desses pontos turísticos, exibindo informações sobre o mesmo: **nome, valor, informações, meios de transporte.**

Ainda na mesma interface, é apresentada a relação de pontos turísticos que o usuário selecionou para visitar, juntamente com o valor (que pode variar conforme a quantidade de pessoas que irá acompanhá-lo) e o valor total, ao final da lista.

- **Implemente estilos diferentes para o esquema de cores *light* e *dark*.**
- **Implementação adaptabilidade quando identificar que há recurso de apontamento.**
- **Pense Mobile-first!**

Assim que terminar, salve seu trabalho em PDF nomeando o arquivo conforme a regra “nome\_sobrenome\_DR1\_TP3.PDF” e poste como resposta a este TP.

## Status da entrega

Status da entrega	Nenhuma tentativa
Status da avaliação	Não avaliado



Data de entrega	segunda, 16 set 2024, 23:59		
Tempo restante	7 dias 6 horas		
Rubrica	Template de Rubrica para ser utilizado com a extensão Rubricator		
	<b>3. Estilizar componentes React com CSS3</b> <b>O aluno executou corretamente 25% dos exercícios da lista proposta?</b>	Não demonstrou o item de rubrica	Demonstrou o item de rubrica
	<b>3. Estilizar componentes React com CSS3</b> <b>O aluno executou corretamente 50% dos exercícios da lista proposta?</b>	Não demonstrou o item de rubrica	Demonstrou o item de rubrica
	<b>3. Estilizar componentes React com CSS3</b> <b>O aluno executou corretamente 75% dos exercícios da lista proposta?</b>	Não demonstrou o item de rubrica	Demonstrou o item de rubrica
	<b>3. Estilizar componentes React com CSS3</b> <b>O aluno executou corretamente 100% dos exercícios da lista proposta?</b>	Não demonstrou o item de rubrica	Demonstrou o item de rubrica
Última modificação	-		
Comentários da entrega	▶ <a href="#">Comentários (0)</a> .		