



Boas-vindas! Caro aspirante a dev!

### Sobre o Desafio:

Você já deve estar familiarizado com o conceito de Kanban visto em aulas de soft skills. Aqui, não vamos nos ater ao conceito, mas à ferramenta de suporte Kaban. Um quadro Kanban é uma ferramenta de gerenciamento de projeto ágil que auxilia na visualização de trabalho, limitação de trabalho em andamento e maximização de eficiência (ou fluxo). Ele pode ajudar equipes ágeis e de DevOps a estabelecer ordem no trabalho diário. Por isso, o seu desafio será o desenvolvimento de uma aplicação em React que ajude o gerenciamento de tarefas do dia a dia de uma equipe de desenvolvimento de software.

Protótipo da Aplicação: [Acessar protótipo](#).

### Orientações e Regras do Desafio.

1 - O desafio ficará limitado aos conceitos das 6 (seis) primeiras aulas. Ou seja, você não poderá usar conceitos avançados que não foram apresentados em aulas, neste momento. Antes que você pergunte: “o por quê?”. Queremos entender como você se sairá das limitações pertinentes a estas primeiras aulas.

2 - O que vimos até agora:

#### 1 - Aula - Conhecendo o React

- O que é o React
- Conceitos de SPA
- Bibliotecas mais usadas
- Introduzindo Babel e Webpack
- Instalação e configuração do ambiente manualmente

#### 2 - Aula - Conceitos Essenciais - Parte I

- Extensões produtivas no VS Code
- Instalação e configuração do ambiente automatizada
- Primeiro componente Funcional
- Introdução a JSX
- Renderizando Elementos

#### 3 - Aula - Conceitos Essenciais - Parte II

- Componentes e Props
- Introdução a Estados e Ciclo de Vida de componentes

#### 4 - Aula - Conceitos Essenciais - Parte III

- Introdução aos Hooks
- Primeiro Hook: useState
- Manipulação de eventos
- Renderização condicional
- Listas e Chaves



### 5 - Aula - Conceitos Essenciais - Parte IV

- Segundo Hook: UseEffect
- Formulários - Formik
- Elevando nível do Estado
- Composição e Herança

### 6 - Aula - Conceitos Essenciais - Parte V

- Estilização de Componentes
- Um pouco de CSS vs SASS
- Styled Components
- Conhecendo a MUI

### 3 - Uso obrigatório:

- Conceito de SPA
- JSX
- Componentização
- Props
- useStates
- Listas
- useEffect
- Formik
- Styled Components

### 4 - Acerca do desenvolvimento:

O desafio deverá ser desenvolvido de forma individual. Em caso de dúvidas você poderá consultar seu monitor, em segundo caso o seu colega de turma, não estando ainda com a dúvida esclarecida, você poderá nos chamar. O desafio deverá cumprir todos os requisitos listados e terá como principal ponto a entrega obrigatória. Para tanto, o aspirante a dev deverá inserir [nesta lista](#) o seu nome e o repositório do projeto.

Obs.: Serão analisados pontos como: interação com git e github.

### 5 - O que a aplicação deve fazer?

- a) O usuário poderá cadastrar uma tarefa e colocar uma data, que será o prazo de término da determinada tarefa.
- b) Se a tarefa por exemplo estiver expirada, deverá aparecer a seguinte mensagem na frente da data e hora: "Tarefa expirada".
- c) Ao cadastrar uma tarefa o usuário poderá escolher um dos três status: "A fazer", "Fazendo" ou "Concluído".
- d) Ao inserir a tarefa, esta deverá aparecer em formato de CARD na coluna correspondente.
- e) Uma vez inserida a tarefa, o usuário poderá excluí-la ou editá-la.



- f) Ao editar uma tarefa, esta deverá sofrer a alteração solicitada.
- g) Exemplo: se o usuário mudar o estado da tarefa que está na coluna “A FAZER” para “FAZENDO”. O card deverá desaparecer do “A FAZER” e aparecer no “FAZENDO”.

**Da Entrega:**

A entrega ocorrerá até às 12:00 do dia 23/05 e poderá ser solicitada a apresentação do trabalho em aula.