

Boas-vindas ao repositório do projeto Arte com Pixels!

Para realizar o projeto, atente-se a cada passo descrito a seguir, e se tiver qualquer dúvida, nos envie por Slack! #vqv 🚀

Aqui você vai encontrar os detalhes de como estruturar o desenvolvimento do seu projeto a partir desse repositório, utilizando uma branch específica e um *Pull Request* para colocar seus códigos.

Termos e acordos

Ao iniciar este projeto, você concorda com as diretrizes do [Código de Conduta e do Manual da Pessoa Estudante da Trybe](#).

Entregáveis

► 🧑 Como entregar

Para entregar o seu projeto você deverá criar um *Pull Request* neste repositório.

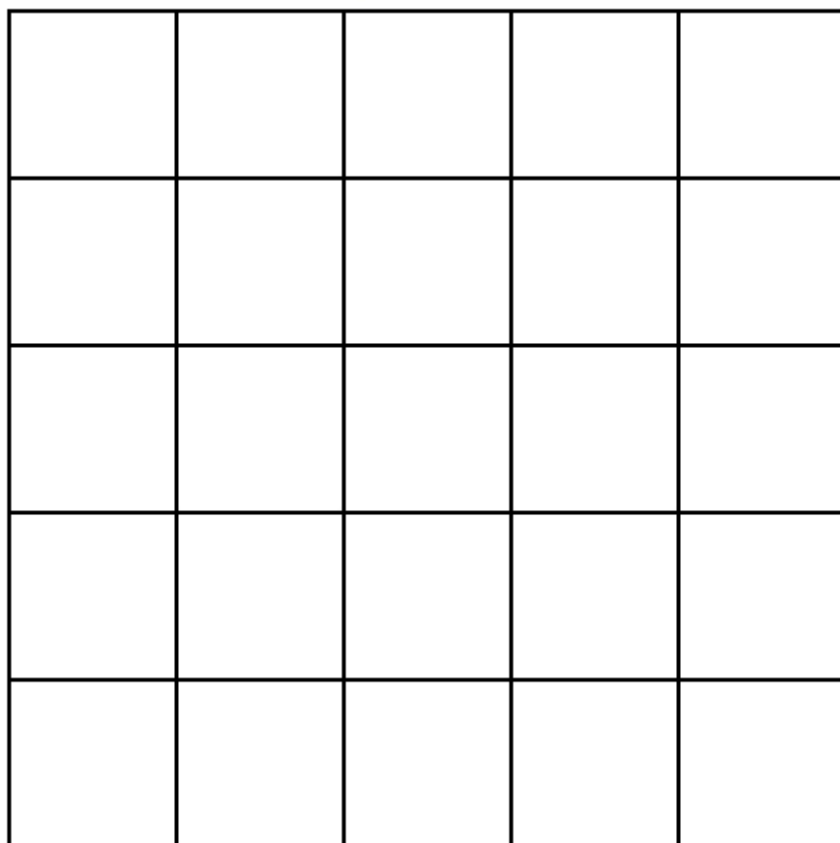
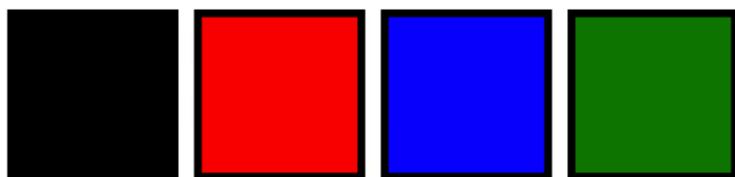
Lembre-se que você pode consultar nosso conteúdo sobre [Git & GitHub](#) e nosso [Blog - Git & GitHub](#) sempre que precisar!

► 🧑 O que deverá ser desenvolvido

Neste projeto, você vai implementar um editor de arte com pixels em que a pessoa usuária poderá escolher uma cor em uma paleta de cores e poderá pintar o que quiser em um quadro branco 🎨 🧑

💡 Veja o exemplo a seguir de como o projeto pode se parecer depois de pronto.

Paleta de cores



Você pode ~~e deve~~ ir além para deixar o projeto com a sua cara e impressionar todas as pessoas, mas não deixe de cumprir os requisitos!

► 📅 Data de Entrega

- Este projeto é individual;
- São **X** dias de projeto;
- Data para entrega final do projeto: **dd/mm/yyyy – 14:00h**.

Orientações

► !! Antes de começar a desenvolver

1. Clone o repositório

- Use o comando: `git clone git@github.com:tryber/sd-0x-project-pixels-art.git`
 - Entre na pasta do repositório que você acabou de clonar:
 - `cd sd-0x-project-pixels-art`
2. Instale as dependências
- `npm install`
3. Crie uma branch a partir da branch `main`
- Verifique que você está na branch `main`
 - Exemplo: `git branch`
 - Se não estiver, mude para a branch `main`
 - Exemplo: `git checkout main`
 - Agora crie uma branch à qual você vai submeter os `commits` de seu projeto
 - Você deve criar uma branch no seguinte formato: `nome-sobrenome-nome-do-projeto`
 - Exemplo: `git checkout -b mariazinha-project-pixels-art`
4. Adicione as mudanças ao `stage` do Git e faça um `commit`
- Verifique que as mudanças ainda não estão no `stage`
 - Exemplo: `git status` (devem aparecer listados os novos arquivos em vermelho)
 - Adicione o novo arquivo ao `stage` do Git
 - Exemplo:
 - `git add .` (adicionando todas as mudanças - *que estavam em vermelho* - ao stage do Git)
 - `git status` (devem aparecer listados os arquivos em verde)
 - Faça o `commit` inicial
 - Exemplo:
 - `git commit -m 'iniciando o projeto pixels art'` (fazendo o primeiro commit)
 - `git status` (deve aparecer uma mensagem tipo *nothing to commit*)
5. Adicione a sua branch com o novo `commit` ao repositório remoto
- Usando o exemplo anterior: `git push -u origin mariazinha-project-pixels-art`
6. Crie um novo `Pull Request (PR)`
- Vá até a página de `Pull Requests` do [repositório no GitHub](#)
 - Clique no botão verde `"New pull request"`
 - Clique na caixa de seleção `"Compare"` e escolha a sua branch **com atenção**
 - Coloque um título para a sua Pull Request
 - Exemplo: `"Cria tela de busca"`
 - Clique no botão verde `"Create pull request"`
 - Adicione uma descrição para o `Pull Request`, e clique no botão verde `"Create pull request"`
 - **Não se preocupe em preencher mais nada por enquanto!**
 - Volte até a [página de Pull Requests do repositório](#) e confira que o seu `Pull Request` está criado

- Faça **commits** das alterações que você fizer no código regularmente
- Lembre-se de sempre após um (ou alguns) **commits** atualizar o repositório remoto
- Os comandos que você utilizará com mais frequência são:
 1. **git status** (para verificar o que está em vermelho - fora do stage - e o que está em verde - no stage);
 2. **git add** (para adicionar arquivos ao stage do Git);
 3. **git commit** (para criar um commit com os arquivos que estão no stage do Git);
 4. **git push** (para enviar o commit para o repositório remoto após o passo anterior);
 5. **git push -u origin nome-da-branch** (para enviar o commit para o repositório remoto na primeira vez que fizer o **push** de uma nova branch).

► 📁 Depois de terminar o desenvolvimento (opcional)

Para sinalizar que o seu projeto está pronto para o "Code Review", faça o seguinte:

- Vá até a página **DO SEU Pull Request**, adicione a label de "code-review" e marque seus colegas:
 - No menu à direita, clique no link "**Labels**" e escolha a label **code-review**;
 - No menu à direita, clique no link "**Assignees**" e escolha **o seu usuário**;
 - No menu à direita, clique no link "**Reviewers**" e digite **students**, selecione o time **tryber/students-sd-0x**.

Caso tenha alguma dúvida, veja este [video explicativo](#).

► 🧑 Revisando um pull request

Use o conteúdo sobre [Code Review](#) para te ajudar a revisar os *Pull Requests*.

► 🏠 Como estruturar seu projeto

- Implemente uma paleta de cores usando **javascript**, **css** e **html**;
- Crie os arquivos **index.html**, **style.css** e **script.js**, que conterão seu código HTML, CSS e JavaScript, respectivamente;

⚠️ É importante que seus arquivos tenham exatamente estes nomes! ⚠️

- Você pode adicionar outros arquivos se julgar necessário. Caso tenha alguma dúvida, poste no *Slack*;

⚠️ ** Recomenda-se que você desenvolva seu projeto com a resolução de tela de **1366 x 768**, a mesma que será utilizada pelo avaliador. Para facilitar a configuração da resolução, use este [plugin do Chrome](#).** ⚠️

- Caso seu projeto contenha imagens, ⚠️ **Atenção**:
 - Não utilize arquivos maiores que **500Kb**;
 - Utilize uma ferramenta como [esta](#) para redimensionar as imagens.
- Você tem liberdade para implementar novos comportamentos ao seu projeto, seja na forma de aperfeiçoamentos em requisitos propostos ou novas funcionalidades, mas atente-se para **não conflitar com os requisitos propostos**.

► 👁 De olho nas dicas

- Não recomenda-se a utilização de `table`, pois o sentido semântico de construir uma tabela no HTML não tem relação com a construção de uma grade de pixels para serem coloridos. Nesse caso, fazer uso de `table` representa uma má prática;
- Que tal usar um `loop` para adicionar o mesmo evento em vários elementos? Ou então a técnica de `event bubbling` combinada com `classList`?
- Se precisar consultar os valores do CSS de um elemento a partir do *JavaScript*, dê uma olhada aqui;
- Para alterar alguma propriedade do CSS de um elemento por meio do *JavaScript*, dê uma olhada no atributo `style` do elemento.
- Caso a avaliação falhe com alguma mensagem de erro do tipo `[409:0326/130838.878602:FATAL:memory.cc(22)] Out of memory. size=4194304` é provável que as imagens que está utilizando são muito grandes. Tente redimensioná-las para um tamanho menor.

► 🐛 Linter

Para garantir a qualidade do código, vamos utilizar neste projeto os linters `ESLint` e `StyleLint`. Desta forma, o código estará alinhado com as boas práticas de desenvolvimento, sendo mais legível e de fácil manutenção! Para rodar o `ESLint` e o `StyleLint` localmente no projeto, execute os comandos abaixo:

1. Para avaliar se os arquivos com a extensão `CSS` estão com o padrão correto

```
npm run lint:styles
```

2. Para avaliar se os arquivos com a extensão `JS` estão com o padrão correto

```
npm run lint
```

⚠ **Atenção** O `ESLint` e o `StyleLint` não serão avaliados neste projeto. Você pode rodar os testes localmente e fazer as correções se desejar. ⚠

► 🛠 Testes e Cypress

O Cypress é uma ferramenta de teste de front-end desenvolvida para a web. Antes de utilizá-lo, certifique-se de ter executado o comando `npm install` dentro do projeto.

Você pode rodar o Cypress localmente para verificar se seus requisitos estão passando, para isso execute um dos seguintes comandos:

1. Para executar os testes apenas no terminal:

```
npm test
```

2. Para executar os testes e vê-los rodando em uma janela de navegador:

```
npm run cypress:open
```

ou

```
npx cypress open
```

Após executar um dos comandos acima, uma janela de navegador será aberta e, então, você poderá escolher o teste a ser executado (project.spec.js) ou escolher **Run all specs** para executar todos os testes

Veja [este vídeo](#) para ver como rodar o Cypress localmente 🤖 📺

- Siga este passo a passo para verificar os **detalhes da execução do avaliador**:
 - Na página do seu *Pull Request*, acima do "botão de merge", procure por **"Evaluator job"** e clique no link **"Details"**;
 - Na página que se abrirá, clique na linha **"Cypress evaluator step"**;
 - Analise os resultados a partir da mensagem **"(Run Starting)"**;
 - Caso tenha dúvidas, consulte [este vídeo](#) ou poste a sua dúvida no *Slack*.

⚠ **O avaliador automático não necessariamente avalia seu projeto na ordem em que os requisitos aparecem no readme. Isso acontece para deixar o processo de avaliação mais rápido. Então, não se assuste se isso acontecer, ok?**

- Contudo, tenha em mente que **nada além do que for pedido nos requisitos será avaliado**. *Esta é uma oportunidade de você exercitar sua criatividade e experimentar com os conhecimentos adquiridos.*

O não cumprimento de um requisito, total ou parcialmente, impactará em sua avaliação.

▶ 🔗 **Links auxiliares para o desenvolvimento do projeto**

- Como pessoa desenvolvedora você deve fazer pesquisas para auxiliar o seu entendimento do assunto. Assim, para solucionar os requisitos do projeto é inevitável e estimulado que pesquisas sejam feitas nas mais variadas fontes (plataforma da Trybe, google, youtube, etc) sempre tomando cuidado para utilizar fontes "confiáveis" nas pesquisas da Internet, como por exemplo:
 - [Javascript.com](#)
 - [W3Schools](#)
 - [MDN](#)

- [StackOverflow](#)

► **Nos dê feedbacks sobre o projeto!**

Ao finalizar e submeter o projeto, não se esqueça de avaliar sua experiência preenchendo o formulário.

Leva menos de 3 minutos!



Link: [Formulário de avaliação do projeto](#)

► **Compartilhe seu portfólio!**

Você sabia que o **LinkedIn** é a principal rede social profissional e compartilhar o seu aprendizado lá é muito importante para quem deseja construir uma carreira de sucesso?

Compartilhe esse projeto no seu **LinkedIn**, marque o perfil da Trybe (**@trybe**) e mostre para a sua rede toda a sua evolução.

Requisitos Obrigatórios

 **Leia todos os requisitos atentamente e siga à risca o que for pedido. Em particular, atente-se para os nomes de *ids* e *classes* que alguns elementos de seu projeto devem possuir. Não troque *ids* por *classes* ou vice-versa** 

1 - Adicione à página o título "Paleta de Cores".

- Sua página deve conter o título "Paleta de Cores"
- O título deverá ficar dentro de uma tag **h1** com o **id** denominado **title**;
- O texto do título deve ser **exatamente** "Paleta de Cores".

O que será verificado



- Verifica se contém um elemento **h1**;
- Verifica se está com id **title**;
- Verifica se existe o título **Paleta de Cores**.

2 - Adicione à página uma paleta contendo quatro cores distintas.

- Sua página deve conter uma paleta com quatro cores
- A paleta de cores deve ser um elemento com **id** denominado **color-palette**, e cada cor individual contida na paleta de cores deve possuir a **classe** chamada **color**;
- A cor de fundo de cada elemento da paleta deverá ser a cor que o elemento representa. **A única cor não permitida na paleta é a cor branca**;
- Cada elemento da paleta de cores deverá ter uma borda preta, sólida e com 1 pixel de largura;
- A paleta de cores deverá listar todas as cores disponíveis para utilização lado a lado, e deverá ser posicionada abaixo do título **Paleta de Cores**;

- A paleta de cores não deve conter cores repetidas.

O que será verificado

- A paleta de cores deve ser um elemento com `id` denominado `color-palette`;
- Cada cor individual da paleta de cores possui a `classe` chamada `color`;
- A cor de fundo de cada elemento da paleta é a cor que o elemento representa  **A única cor não permitida na paleta é a cor branca **;
- Verifica se cada elemento da paleta de cores tem uma borda preta, sólida e com 1 pixel de largura;
- A paleta lista todas as cores disponíveis para utilização, lado a lado;
- A paleta de cores está posicionada abaixo do título `Paleta de Cores`;
- A paleta de cores não contém cores repetidas.

3 - Adicione a cor **preta** como a primeira cor da paleta de cores.

- ▶ A primeira cor na paleta criada no requisito 2 deve ser preta

O que será verificado

- Verifica se a primeira cor da paleta tem `background-color` preto;
- As demais cores podem ser escolhidas livremente.

4 - Adicione à página um quadro contendo 25 pixels.

- ▶ Sua página deve conter um quadro de pixels 5x5

- O quadro de "pixels" deve ter 5 elementos de largura e 5 elementos de comprimento;
- O quadro de "pixels" deve possuir o `id` denominado `pixel-board`, ao passo que cada "pixel" individual dentro do quadro deve possuir a `classe` denominada `pixel`;
- Ao abrir a página, a cor inicial dos "pixels" que compõem o quadro de pixels deve ser branca;
- O quadro de "pixels" deve aparecer abaixo da paleta de cores.

O que será verificado

- O quadro de "pixels" possui o `id` denominado `pixel-board`;
- Cada pixel individual dentro do quadro possui a `classe` denominada `pixel`;
- A cor inicial dos "pixels" dentro do quadro, ao abrir a página, é branca;
- O quadro de "pixels" aparece abaixo da paleta de cores.

5 - Faça com que cada pixel do quadro tenha largura e altura de 40 pixels e borda preta de 1 pixel de espessura.

- ▶ Cada **pixel** do quadro deve possuir 40px de largura e 40px de altura e uma borda preta solida de 1px de espessura

O que será verificado

- O quadro de "pixels" tem altura e comprimento de 5 elementos;
- Verifica se 40 pixels de altura e 40 pixels de largura é o tamanho total do elemento, incluindo seu conteúdo e excluindo a borda preta, que deve ser criada à parte.

6 - Defina a cor preta como cor inicial da paleta de cores

- ▶ Ao carregar a página, a cor preta já deve estar selecionada na paleta para pintar os pixels do quadro.
- O elemento da cor preta deve possuir inicialmente a classe **selected**;
- O elemento que deverá receber a classe **selected** deve ser um dos elementos que possuem a classe **color**, como especificado no **requisito 2**.

O que será verificado

- O elemento da cor preta possui, inicialmente, a classe **selected**;
- Nenhuma outra cor da paleta tem a classe **selected** ao carregar a página.

7 - Selecione uma cor na paleta de cores e preencha os pixels no quadro.

- ▶ Ao clicar em outras cores da paleta de cores, a cor deve ser selecionada e poderá ser utilizada para preencher os quadros
- A **classe selected** deve ser adicionada à cor selecionada na paleta, ao mesmo tempo em que é removida da cor anteriormente selecionada;
- Somente uma das cores da paleta deve ter a classe **selected** de cada vez;
- Os elementos que deverão receber a **classe selected** devem ser os mesmos elementos que possuem a classe **color**, como especificado no **requisito 2**.

O que será verificado

- Verifica se somente uma cor da paleta de cores tem a classe **selected** de cada vez;
- Verifica se os pixels dentro do quadro não têm a classe **selected** quando são clicados.

8 - Preencha um pixel do quadro com a cor selecionada na paleta de cores.

- ▶ Ao clicar em algum **pixel** do quadro a sua cor deve ser alterada para a cor selecionada na paleta de cores

O que será verificado

- Ao carregar a página deve ser possível pintar os pixels do quadro de preto;
- Após selecionar outra cor na paleta de cores, é possível pintar os pixels do quadro com essa cor;

- Somente o pixel que foi clicado foi preenchido com a cor selecionada, sem influenciar na cor dos demais pixels.

9 - Crie um botão que retorne a cor do quadro para a cor inicial.

- Sua página deve ter um botão que, ao ser clicado, deixe todos os **pixels** do quadro com a cor branca

O que será verificado

- O botão tem o **id** denominado **clear-board**;
- O botão está posicionado entre a paleta de cores e o quadro de pixels;
- O texto do botão é **Limpar**;
- Ao clicar no botão, o quadro de pixels é totalmente preenchido de branco.

Requisitos Bônus

10 - Faça o quadro de pixels ter seu tamanho definido pela pessoa usuária.

- Sua página deve conter um input para que a pessoa usuária possa definir o tamanho do quadro de pixels
 - Crie um input e um botão que permitam definir um quadro de pixels com tamanho entre 5 e 50. Ao clicar no botão, deve ser gerado um quadro de **N** pixels de largura e **N** pixels de altura, em que **N** é o número inserido no input. Ou seja, se o valor passado para o input for igual a 7, ao clicar no botão, será gerado um quadro de 49 pixels (7 pixels de largura x 7 pixels de altura);
 - O input deve ter o **id** denominado **board-size** e o botão deve ter o **id** denominado **generate-board**;
 - O input só deve aceitar número maiores que zero. Essa restrição **deve** ser feita usando os atributos do elemento **input**;
 - O botão deve conter o texto "VQV";
 - O input deve estar posicionado entre a paleta de cores e o quadro de pixels;
 - O botão deve estar posicionado ao lado do input;
 - Se nenhum valor for colocado no input ao clicar no botão, mostre um **alert** com o texto: "Board inválido!";
 - O novo quadro deve ter todos os pixels preenchidos com a cor branca.

O que será verificado

- Existe um input com o id **board-size**;
- Existe um botão com o id **generate-board**;

- O input só aceita número maiores que zero. Essa restrição deve ser feita usando os atributos do elemento `input`;
- O botão contém o texto `VQV`;
- O input está posicionado entre a paleta de cores e o quadro de pixels;
- O botão está posicionado ao lado direito do input;
- Caso nenhum valor for colocado no input ao clicar no botão, um `alert` é exibido com o texto: `Board inválido!`;
- Ao clicar no botão com um valor preenchido, o tamanho do board muda;
- O novo quadro tem todos os pixels preenchidos com a cor branca.

11 - Limite o tamanho mínimo e máximo do quadro de pixels.

- ▶ A pessoa usuária não pode definir um quadro com menos que 5 e mais que 50 `pixels`
 - Caso o valor digitado no input `board-size` esteja fora do intervalo de 5 a 50, faça:
 1. Se o valor de `board-size` for menor que 5, considere 5 `pixels` como padrão;
 2. Se o valor de `board-size` for maior que 50, considere 50 `pixels` como padrão.

O que será verificado

- A altura máxima do board é 50;
- A altura do board é 5 pixels quando um valor menor que 5 é colocado no input;
- A altura do board é 50 pixels quando um valor maior que 50 é colocado no input.

12 - Faça com que as cores da paleta sejam geradas aleatoriamente ao carregar a página.

- ▶ A cor preta ainda precisa estar presente e deve ser a primeira na sua paleta de cores.

O que será verificado

- As cores geradas na paleta são diferentes a cada carregamento da página;
- A cor preta ainda está presente e é a primeira na sua paleta de cores.