

 tryber / **sd-03-block5-project-pixels-art**

☆ 0 stars 🍴 0 forks

☆ Star

👁 Watch ▼

<> **Code**

! Issues

🔗 Pull requests 63

▶ Actions

📁 Projects

📖 Wiki

! Security

🔗 master ▼

...



albertosca Merge pull request #54 from tryber/adjust-requisite-6 ...

✓ on 3 Mar ⌚ 12

[View code](#)

README.md



🔗 Boas vindas ao repositório do projeto de Arte com Pixels!

Você já usa o GitHub diariamente para desenvolver os exercícios, certo? Agora, para desenvolver os projetos, você deverá seguir as instruções a seguir. Fique atento a cada passo e, se tiver qualquer dúvida, nos envie por *Slack*! #vqv 🚀

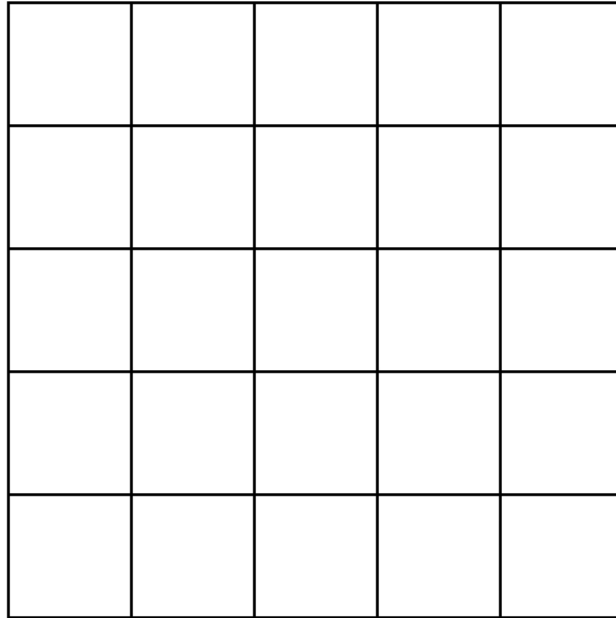
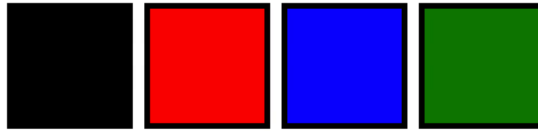
Aqui você vai encontrar os detalhes de como estruturar o desenvolvimento do seu projeto a partir deste repositório, utilizando uma branch específica e um *Pull Request* para colocar seus códigos.

O que deverá ser desenvolvido

Neste projeto, você implementará um editor de arte com pixels. Ou seja, dada uma paleta de cores e um quadro composto por pixels, você vai permitir que quem usa consiga pintar o que quiser no quadro!

Abaixo você pode ver um exemplo de um editor de arte com pixels. Utilize este exemplo como referência. Você poderá estilizar seu projeto da forma que desejar desde que todos os requisitos sejam cumpridos. Use sua imaginação!

Paleta de cores



Entregáveis

Para entregar o seu projeto você deverá criar um *Pull Request* para este repositório no **GitHub**.

Este *Pull Request* deverá conter, necessariamente, 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.

Requisitos do projeto

A seguir, estão listados todos os requisitos do projeto. Leia-os atentamente e siga à risca o que for pedido. Em particular, atente-se para os nomes de classes e ids que alguns elementos de seu projeto devem possuir. **Não troque ids por classes ou vice-versa.** O não cumprimento de um requisito, total ou parcialmente, impactará em sua avaliação.

Os requisitos do seu projeto são avaliados automaticamente, sendo utilizada a resolução `1366 x 768` (1366 pixels de largura por 768 pixels de altura). Logo, recomenda-se desenvolver seu projeto usando a mesma resolução, via instalação [deste plugin](#) do `Chrome` para facilitar a configuração dessa resolução.

Você tem liberdade para adicionar novos comportamentos ao seu projeto, seja na forma de aperfeiçoamentos em requisitos propostos ou novas funcionalidades, **desde que tais comportamentos adicionais não conflitem com os requisitos propostos**. Em outras palavras, você pode fazer mais do que for pedido, mas nunca menos. 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.

1 - A página deve possuir o título "Paleta de Cores"

O título deverá ficar dentro de uma tag `h1` com o id `title`.

2 - A página deve possuir uma paleta de quatro cores

A paleta 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 deve possuir o id `color-palette`, ao passo que cada cor individual da paleta de cores deve possuir a classe `color`.

A cor de fundo de cada elemento da paleta deverá ser a cor que o elemento representa.

3 - A cor preta deve ser a primeira na paleta de cores

As demais cores podem ser escolhidas livremente.

4 - A página deve possuir um quadro de pixels, com 25 pixels

O quadro de pixels deve possuir 5 pixels de largura por 5 pixels de altura. A cor inicial dos pixels, ao abrir a página, deve ser branca.

O quadro de pixels deve possuir o id `pixel-board`, ao passo que cada pixel individual no quadro deve possuir a classe `pixel`.

5 - O quadro de pixels deve aparecer abaixo da paleta de cores

6 - Cada pixel do quadro de pixels deve possuir 40 pixels de largura e 40 pixels de altura e ser delimitado por uma margem

preta de 1 pixel

40 pixels deve ser o tamanho total do elemento, incluindo seu conteúdo e excluindo a margem preta, que deve ser contada à parte.

7 - Ao carregar a página, a cor preta da paleta já deve estar selecionada para pintar os pixels

O elemento da cor preta deve possuir, inicialmente, a classe `selected`.

Note que o elemento que deverá receber a classe `selected` deve ser um dos elementos que possuem a classe `color`, como especificado no requisito 2.

8 - Ao clicar em uma das cores da paleta, a cor selecionada na paleta é que vai ser usada para preencher os pixels

A classe `selected` deve ser adicionada à cor selecionada, ao mesmo tempo em que é removida da cor anteriormente selecionada.

Note que os elementos que deverão receber a classe `selected` devem ser os mesmos elementos que possuem a classe `color`, como especificado no requisito 2.

9 - Ao clicar em um pixel com uma cor selecionada, o pixel deve ser preenchido com esta cor

10 - Ao clicar em um pixel com uma cor selecionada, somente esse pixel deverá ser preenchido, sem influenciar na cor dos demais pixels

BÔNUS

11 - Crie um botão que, ao ser clicado, limpa o quadro, preenchendo a cor de todos seus pixels com branco

O botão deve ter o id `clear-board`.

12 - Faça o quadro de pixels ter seu tamanho definido pelo usuário

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, onde **N** é o número inserido no input.

Ou seja, se o valor passado para o input for 7, ao clicar no botão, vai ser gerado um quadro de 49 pixels (7 pixels de largura x 7 pixels de altura).

O novo quadro deve ter todos os pixels preenchidos com a cor branca.

O input e o botão deve ter os **ids** `board-size` e `generate-board`, respectivamente.

Caso o valor digitado fuja do intervalo de 5 a 50, faça:

- Valor menor que 5, considerar 5 como padrão;
- Valor maior que 50, considerar 50 como padrão.

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

Lembre-se de que a cor preta ainda precisa estar presente e deve ser a primeira na sua paleta de cores.

Instruções para entregar seu projeto:

ANTES DE COMEÇAR A DESENVOLVER:

1. Clone o repositório

- `git clone https://github.com/tryber/sd-03-block5-project-pixels-art.git`.

2. Crie uma branch a partir da branch `master`





- Verifique que você está na branch `master`
 - Exemplo: `git branch`
- Se não estiver, mude para a branch `master`
 - Exemplo: `git checkout master`
- Agora, crie uma branch onde você vai guardar os `commits` do seu projeto
 - Você deve criar uma branch no seguinte formato: `nome-de-usuario-nome-do-projeto`
 - Exemplo: `git checkout -b joaozinho-pixels-art-project`

3. Crie na raiz do projeto os arquivos que você precisará desenvolver:

- Verifique que você está na raiz do projeto

- Exemplo: `pwd` -> o retorno vai ser algo tipo `/Users/joaozinho/code/sd-01-week4-5-project-pixels-art`
 - Crie os arquivos `index.html`, `style.css` e `script.js`
 - Exemplo:
 - `touch index.html style.css script.js`
 - 4. Adicione as mudanças ao *stage* do Git e faça um `commit`
 - Verifique que as mudanças ainda não estão no *stage*
 - Exemplo: `git status` (deve aparecer listada a pasta *joaozinho* 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` (deve aparecer listado o arquivo *joaozinho/README.md* em verde)
 - Faça o `commit` inicial
 - Exemplo:
 - `git commit -m 'iniciando o projeto pixel 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 joaozinho-pixels-art-project`
 - 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**
 - 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
-

DURANTE O DESENVOLVIMENTO



-  **LEMBRE-SE DE CRIAR TODOS OS ARQUIVOS DENTRO DA PASTA COM O SEU NOME** 
-  **PULL REQUESTS COM ISSUES NO CODE CLIMATE NÃO SERÃO AVALIADAS, ATENTE-SE PARA RESOLVÊ-LAS ANTES DE FINALIZAR O DESENVOLVIMENTO!** 
- 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:
 - i. `git status` (para verificar o que está em vermelho - fora do stage - e o que está em verde - no stage)
 - ii. `git add` (para adicionar arquivos ao stage do Git)
 - iii. `git commit` (para criar um commit com os arquivos que estão no stage do Git)
 - iv. `git push -u nome-da-branch` (para enviar o commit para o repositório remoto na primeira vez que fizer o `push` de uma nova branch)
 - v. `git push` (para enviar o commit para o repositório remoto após o passo anterior)

DEPOIS DE TERMINAR O DESENVOLVIMENTO

Para "**entregar**" seu projeto, siga os passos a seguir:

- 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`

Se ainda houver alguma dúvida sobre como entregar seu projeto, [aqui tem um video explicativo](#).

 Lembre-se que garantir que todas as *issues* comentadas pelo CodeClimate estão resolvidas! 

REVISANDO UM PULL REQUEST



À medida que você e os outros alunos forem entregando os projetos, vocês serão alertados **via Slack** para também fazer a revisão dos *Pull Requests* dos seus colegas. Fiquem atentos às mensagens do "*Pull Reminders*" no *Slack*!

Use o material que você já viu sobre [Code Review](#) para te ajudar a revisar os projetos que chegaram para você.

Releases

No releases published

[Create a new release](#)

Packages

No packages published

[Publish your first package](#)

Contributors 4



LeandroLM



inaC



albertosca



LucasCFerraz Lucas Cassiano Ferraz Paolillo