

 tryber / sd-03-block5-project-color-guess

☆ 0 stars 🍴 0 forks

☆ Star

👁 Watch ▼

<> Code

! Issues

🔗 Pull requests 15

▶ Actions

📁 Projects

📖 Wiki

🛡 Security

🔗 master ▼

...



jeanpsv Merge pull request #2 from tryber/feature/remove-bonus-requirem...

...

on 3 Mar

🕒 3

[View code](#)

README.md



Boas vindas ao repositório do projeto de Adivinhe a Cor!

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 desse 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 jogo de adivinhar a cor. Utilize este exemplo como referência. Você poderá estilizar seu projeto da forma que desejar desde que todos os requisitos sejam cumpridos. Use sua imaginação!



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 um título centralizado com o nome do seu jogo

O título deverá conter o id `title`.

2 - A página deve conter um texto centralizado com o *RGB* a ser adivinhado

O texto deverá conter os três números das cores RGB a ser adivinhada, separados por vírgula e entre parênteses, conforme ilustrado na imagem acima. Exemplo: `(168, 34, 248)`.

O elemento que contém o texto deverá conter o id `rgb-color`.

3 - A página deve conter 6 bolas como opção de cor de adivinhação

Cada bola deverá conter a classe `ball`.

4 - Ao clicar em uma bola, deve ser mostrado um texto que indica se a cor selecionada foi a correta

Se a bola clicada for a correta, deve ser exibido o texto "Acertou!". Caso contrário, o texto deve ser "Errou! Tente novamente!".

Quando o jogo é iniciado, o texto exibido deve ser "Escolha uma cor".

O elemento que conterá o texto deve ter o `id` `answer` .

5 - As cores das bolas devem ser geradas aleatoriamente via JavaScript

Isso vale tanto quando a página é carregada, quanto ao clicar no botão para reiniciar o jogo.

6 - Deve haver um botão para reiniciar o jogo

Ao clicar no botão, deve ser sorteada uma nova cor RGB e novas cores para as bolas, além de voltar os textos para a forma inicial.

Este botão também poderá ser utilizado para iniciar um novo jogo.

BÔNUS

7 - Crie um placar que incremente 3 pontos para cada acerto no jogo

No início do jogo, o placar deve ser zero. A cada acerto, o placar deve ser incrementado em 3 pontos.

Ao clicar no botão reiniciar, o placar **NÃO** deve ser resetado para zero.

O elemento que conterá o placar deve ter o `id` `score` .

Instruções para entregar seu projeto:

ANTES DE COMEÇAR A DESENVOLVER:

1. Clone o repositório

- `git clone https://github.com/tryber/sd-03-week5-project-color-guess.git` .
- Entre na pasta do repositório que você acabou de clonar:
 - `cd sd-03-week5-project-color-guess`

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-color-guess-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-03-week5-project-color-guess`
- 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 color guess'` (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-color-guess-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-sd-02`

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](#)