

Aula 2

Começando a programar

Unidade

**Página web: utilizando
HTML e CSS para construir
um site criativo**

O que vamos aprender?



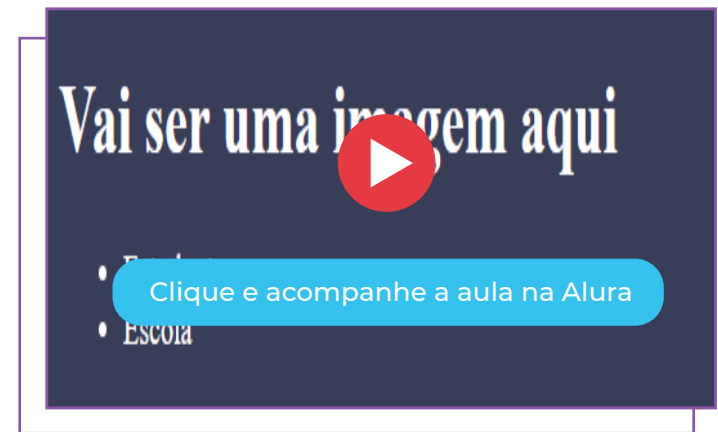
Criar um repositório no GitHub.



Desenvolver o cabeçalho da página utilizando as linguagens HTML e CSS.

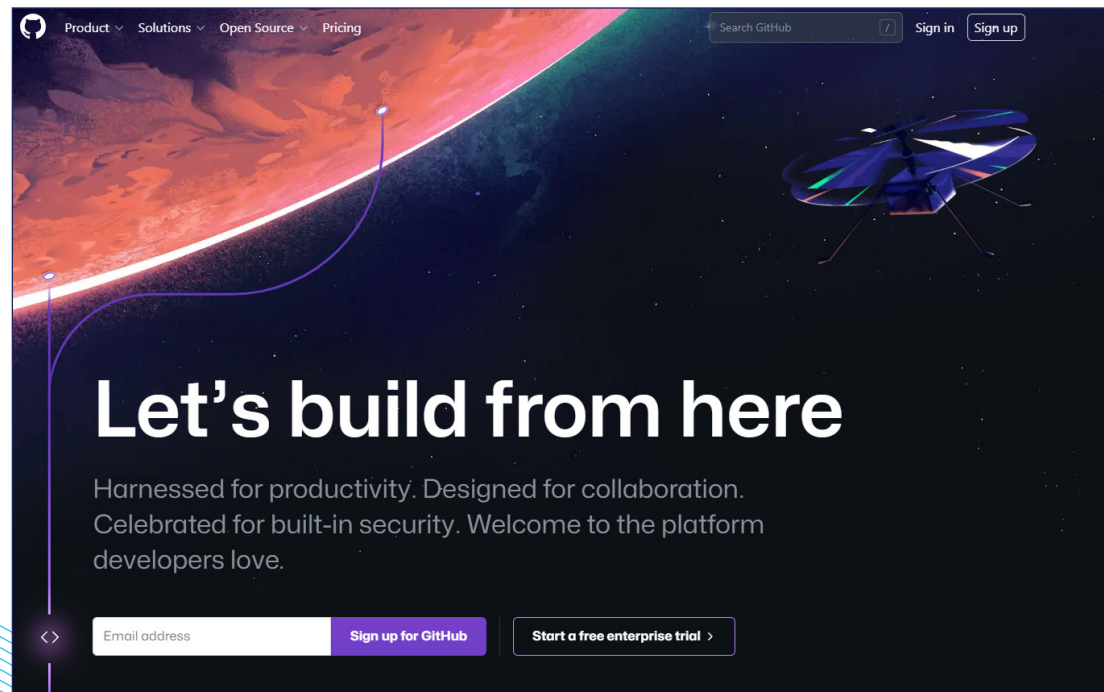


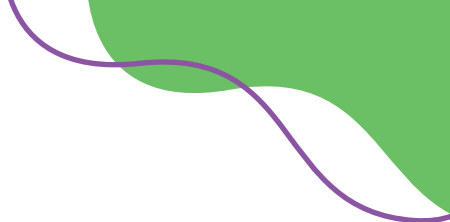
Alterar a cor de fundo da página e a cor do texto.



Iniciando o projeto

Ao criarmos um projeto, devemos decidir onde armazená-lo. No universo da programação, as pessoas desenvolvedoras utilizam **repositórios** em nuvem para guardar seus projetos. Um repositório em nuvem é um espaço alocado em um servidor da internet. Existem vários repositórios que podem ser utilizados, vamos usar o GitHub.





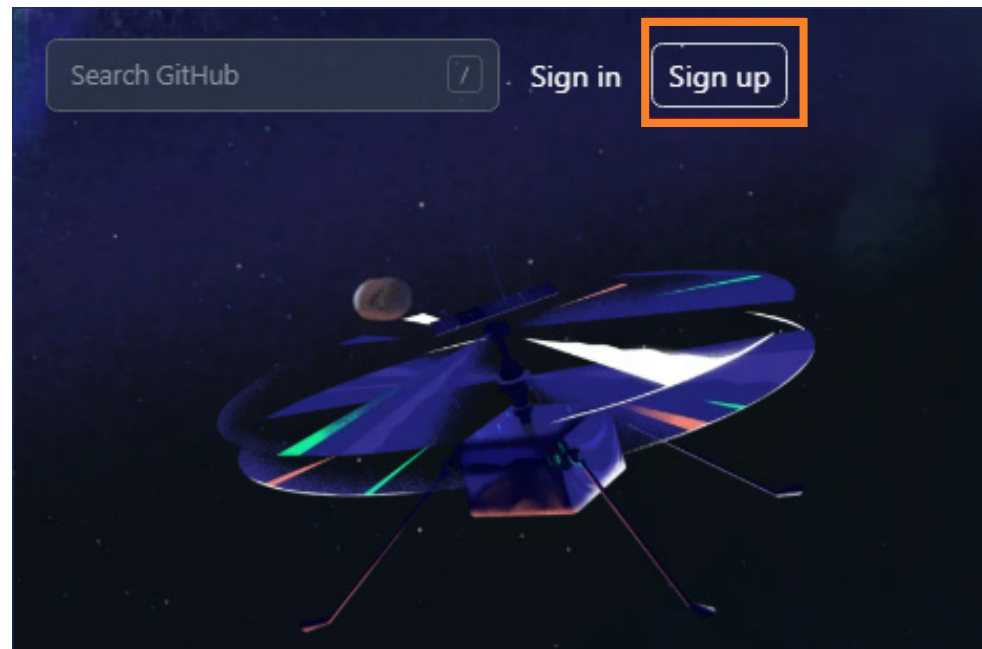
O GitHub é um site mundialmente utilizado por pessoas desenvolvedoras e entusiastas da área de tecnologia. O site permite promover o compartilhamento de códigos de programação. É uma **plataforma em nuvem**, o que significa que os arquivos e dados ficam armazenados e são acessados por meio da internet. Ela foi criada em 2008 e desde 2018 pertence à empresa Microsoft. Por meio do GitHub, podemos compartilhar nossos projetos com toda a comunidade de desenvolvedores ou podemos organizar e armazenar nossos projetos de forma privada, ou seja, apenas você terá acesso ao que foi produzido.

O primeiro passo para começar a utilizar o GitHub é criar uma conta. Caso você já tenha uma conta criada, poderá pular os passos a seguir.

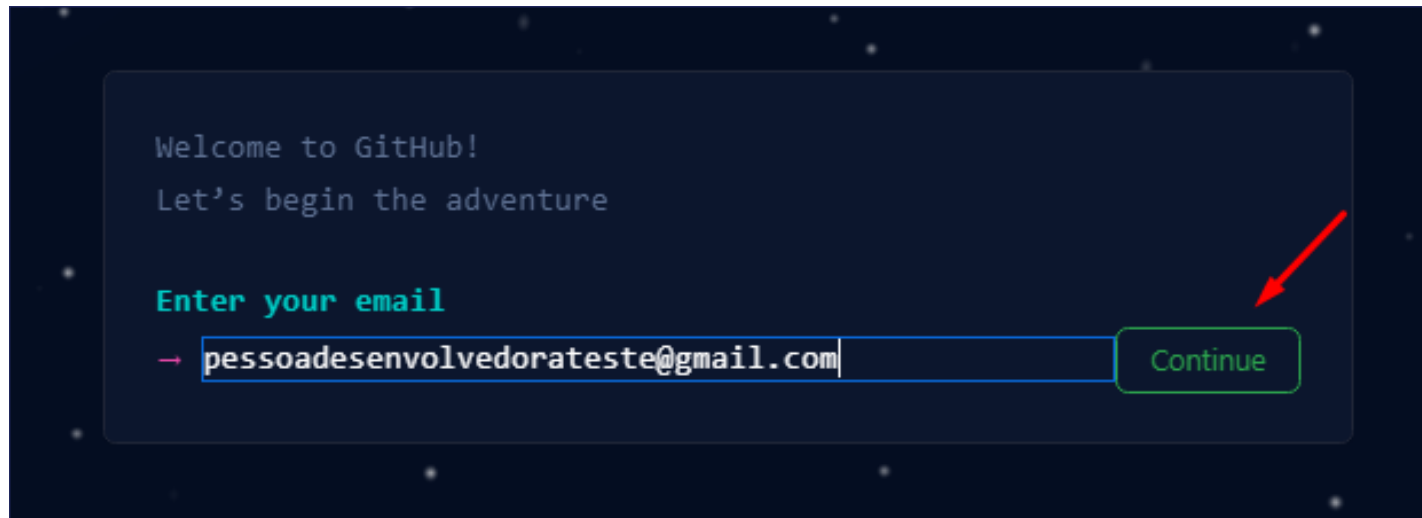
Devemos acessar o site do GitHub pelo endereço <https://github.com/>.



Após acessar o site, uma página inicial irá abrir. No canto superior direito da página, devemos clicar em *Sign up*, que em inglês significa “inscreva-se”, para criar uma conta. Todo o site estará na língua inglesa, por essa razão aprenderemos as traduções das palavras mais importantes.



Em seguida, devemos digitar nosso e-mail pessoal. Pode ser qualquer e-mail, contanto que tenhamos acesso a ele. Após digitar, clique em *Continue*, que em inglês significa “continuar”.



Welcome to GitHub!
Let's begin the adventure

Enter your email

→

A red arrow points to the 'Continue' button.

Após incluir o e-mail, crie uma senha com no mínimo 8 caracteres, entre letras e números.

A screenshot of the GitHub account creation interface. The background is dark blue with white text. At the top, it says "Welcome to GitHub!" and "Let's begin the adventure". Below that, it prompts "Enter your email" and shows a green checkmark next to the email "pessoadesenvolvedorateste@gmail.com". The "Create a password" section is highlighted with an orange border. It contains a password input field with a red arrow on the left and a green eye icon on the right. To the right of the input field is a green "Continue" button. Below the password field, there are three green progress bars, the text "Password is strong", and a note: "Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter." The entire interface is surrounded by colorful abstract shapes like circles, lines, and wavy patterns in green, blue, and purple.

Welcome to GitHub!
Let's begin the adventure

Enter your email
✓ pessoadesenvolvedorateste@gmail.com

Create a password

→ [password field] [Continue]

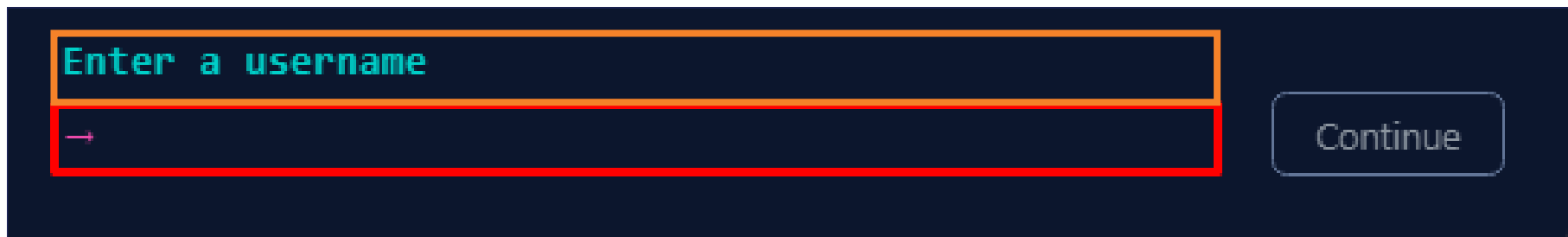
— — —
Password is strong
Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Logo abaixo, o GitHub define se a senha é forte (*strong*). Após digitar a senha, clicaremos em *Continue* novamente.

Password is strong

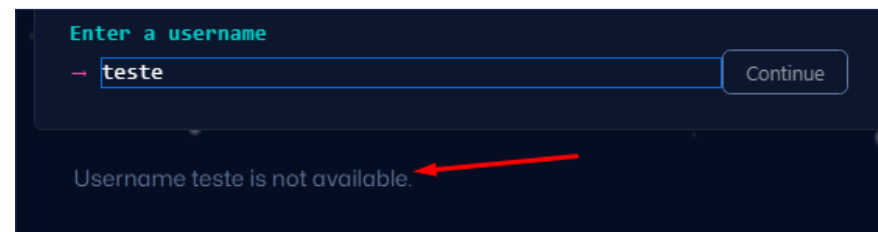
Make sure it's at least 15 characters OR at least 8 characters including a number and a lowercase letter.

Devemos também definir um *username*, que em inglês significa “nome do usuário”. Esse *username* também poderá ser utilizado para entrar no GitHub. Como sugestão, coloque seu nome e sobrenome.



A screenshot of a dark-themed interface with a text input field labeled "Enter a username" in teal. The input field has an orange border on top and a red border on the bottom. A small pink arrow points to the input field. To the right of the input field is a "Continue" button.

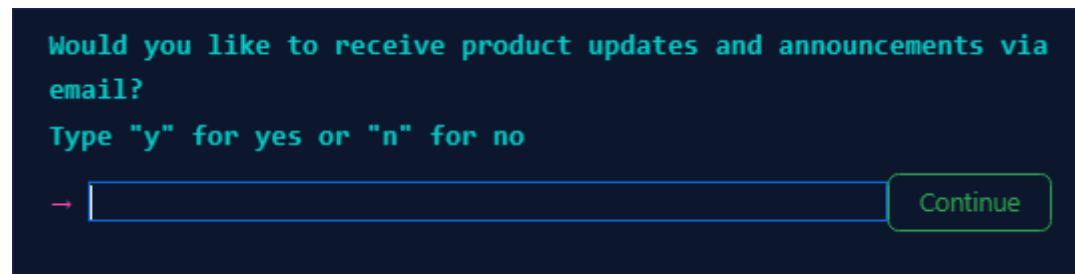
Caso o nome do usuário escolhido não esteja disponível, aparecerá a seguinte mensagem: *not available*. Essa mensagem aparece quando outras pessoas utilizam um username idêntico.



A screenshot of the same dark-themed interface. The input field now contains the text "teste" and has a blue border. The "Continue" button is still present. Below the input field, the message "Username teste is not available." is displayed in a light gray font. A red arrow points to this message.

Após a escolha do nome, clique novamente em *Continue*.

A próxima tela exibirá a seguinte pergunta: você gostaria de receber atualizações de produtos e anúncios por e-mail? Digite y para sim ou n para não. Caso seja do seu interesse receber atualizações sobre o GitHub, você pode digitar y; caso contrário, você deve digitar n.



A terminal window with a dark background. The text inside is as follows:

```
Would you like to receive product updates and announcements via  
email?  
Type "y" for yes or "n" for no  
→  Continue
```

Após a escolha, clique novamente em *Continue*.

O próximo passo é verificar a conta. Para isso, um enigma será exibido. Clique em *Verificar*.

Verify your account

Verificação

Resolva este enigma para sabermos que você é uma pessoa de verdade

Verificar

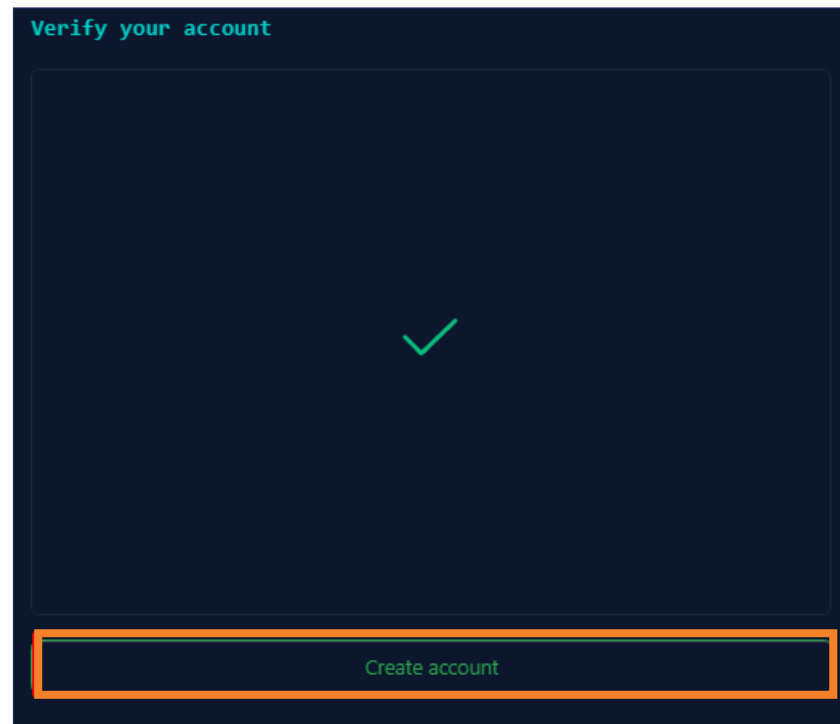
⌂

🔊

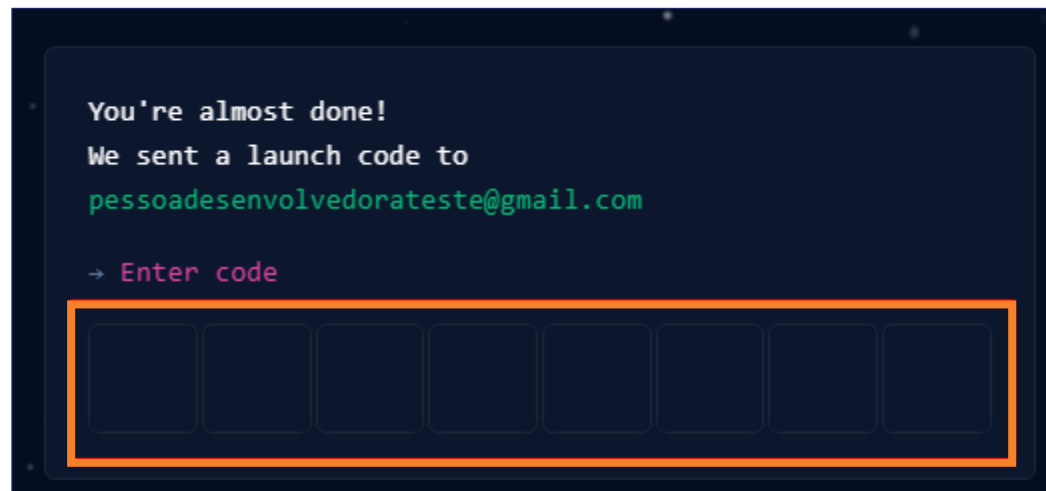
Os enigmas são resolvidos interpretando o enunciado e clicando na opção correta. Por exemplo, o enigma abaixo solicita que escolha a galáxia em espiral.



Após clicar na opção correta, uma tela semelhante à imagem abaixo aparecerá. Então, clique em *Create account*, que em inglês significa “criar a conta”.



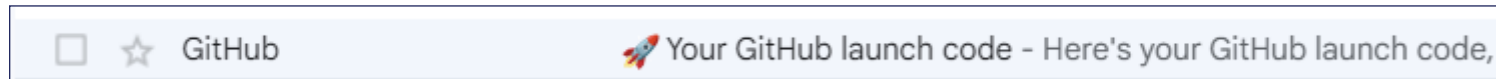
Uma tela semelhante à imagem a seguir será carregada com a mensagem informando que um código numérico foi enviado para o e-mail cadastrado. No espaço destacado devemos digitar o código numérico.



The image shows a dark-themed user interface for a verification step. It contains the following text and elements:

- Text: "You're almost done!"
- Text: "We sent a launch code to"
- Email address: `pessoadesenvolvedorateste@gmail.com`
- Text: "→ Enter code"
- A row of eight empty square input boxes for a numeric code, which is highlighted by a thick orange rectangular border.

Para saber qual é o código, devemos ir até a conta de e-mail cadastrada e clicar na mensagem enviada pelo GitHub.

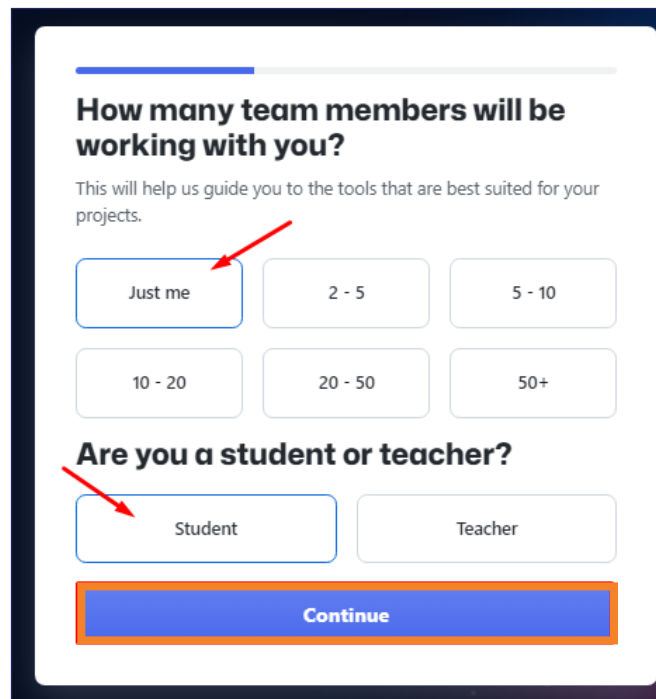


No e-mail, verificaremos qual é o código enviado. Em seguida, digitaremos na página do GitHub os mesmos números que aparecem na mensagem. Feito isso, a seguinte página irá abrir:

The image shows the GitHub welcome screen. On the left, the GitHub logo is at the top, followed by a group of people icon and the text "Welcome to GitHub" in large green letters, with "We are glad you're here." below it. On the right, there is a white box with a blue progress bar at the top. The first section is titled "How many team members will be working with you?" and includes a subtext "This will help us guide you to the tools that are best suited for your projects." Below this are six buttons: "Just me", "2 - 5", "5 - 10", "10 - 20", "20 - 50", and "50+". The second section is titled "Are you a student or teacher?" and has two buttons: "Student" and "Teacher". At the bottom of the white box is a large blue "Continue" button.

Na tela, aparecerá a seguinte pergunta: *How many team members will be working with you?*, que significa “quantos membros da equipe trabalharão com você?”. Devemos marcar a opção *Just me*, que em inglês significa “apenas eu”.

Logo abaixo, a próxima pergunta é “você é um estudante ou professor?”. Devemos marcar a opção *student*, que em inglês significa estudante. Logo em seguida, clicaremos em *Continue*.



How many team members will be working with you?

This will help us guide you to the tools that are best suited for your projects.

☒ Just me ☐ 2 - 5 ☐ 5 - 10

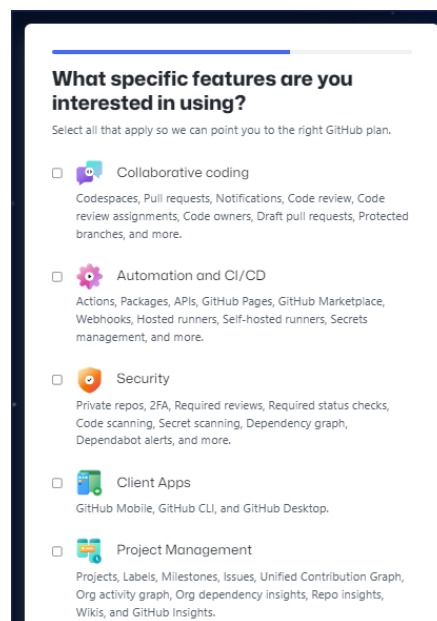
☐ 10 - 20 ☐ 20 - 50 ☐ 50+

Are you a student or teacher?






☒ Student ☐ Teacher

Continue

A próxima tela exibirá algumas ferramentas que podem ser adicionadas na área de trabalho no GitHub.



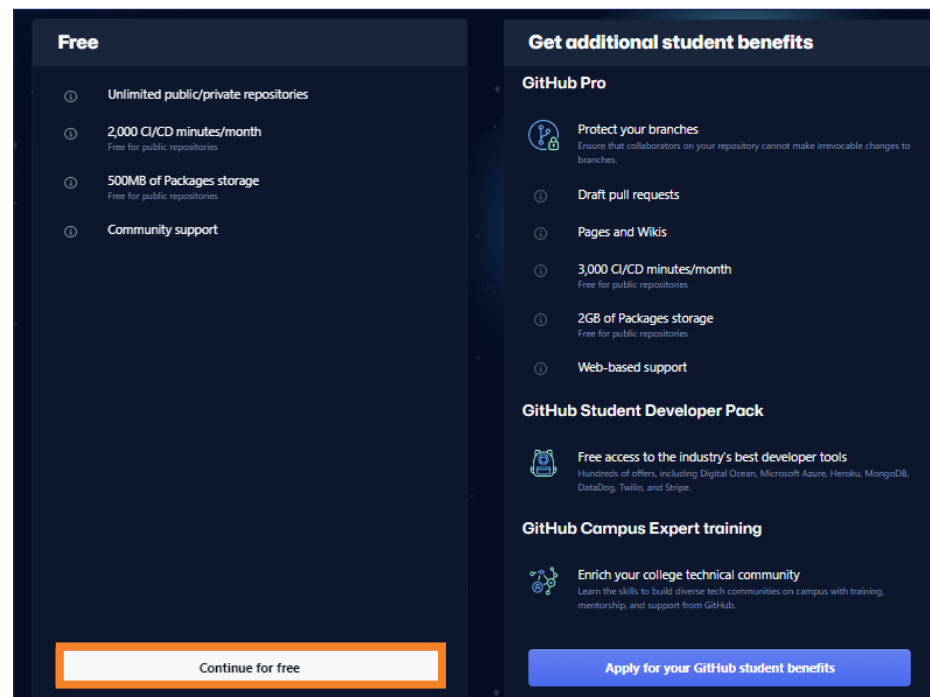
What specific features are you interested in using?
Select all that apply so we can point you to the right GitHub plan.

- ☐  Collaborative coding
Codespaces, Pull requests, Notifications, Code review, Code review assignments, Code owners, Draft pull requests, Protected branches, and more.
- ☐  Automation and CI/CD
Actions, Packages, APIs, GitHub Pages, GitHub Marketplace, Webhooks, Hosted runners, Self-hosted runners, Secrets management, and more.
- ☐  Security
Private repos, 2FA, Required reviews, Required status checks, Code scanning, Secret scanning, Dependency graph, Dependabot alerts, and more.
- ☐  Client Apps
GitHub Mobile, GitHub CLI, and GitHub Desktop.
- ☐  Project Management
Projects, Labels, Milestones, Issues, Unified Contribution Graph, Org activity graph, Org dependency insights, Repo insights, Wikis, and GitHub Insights.

Como não vamos utilizar essas ferramentas, podemos pular essa etapa clicando no fim da página em *Continue*”.

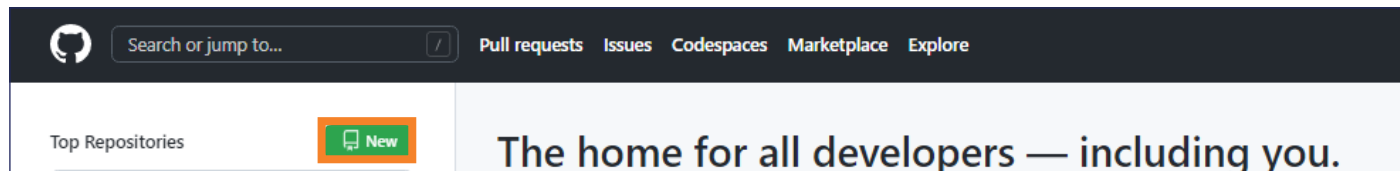
Continue

A próxima página exibirá duas opções de planos: uma gratuita (*free*) e uma para estudantes (*Student benefits*). Para escolher a opção de estudante, é necessário seguir algumas regras, dentre elas: estar matriculado em uma instituição de ensino e ter idade mínima igual a 13 anos. Para mais detalhes, podemos pesquisar mais informações no Google. Vamos utilizar a opção gratuita, clicando no botão *Continue for free*.



Após as etapas anteriores, a página do GitHub será carregada.

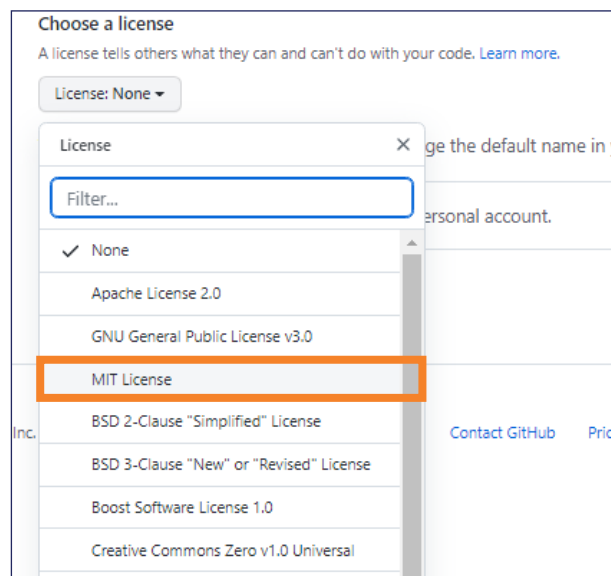
A conta está criada. Devemos então criar um repositório clicando no botão *New*, que significa “novo”.



No campo *Repository name*, devemos dar um nome para o novo repositório. Podemos escrever a sugestão de nome “Site-escola”. Além disso, devemos marcar a opção *Add a README file*, que significa “adicione um arquivo LEIA-ME”.

A screenshot of the "Create a new repository" form on GitHub. The form is titled "Create a new repository" and includes a sub-header: "A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository.](#)". The form fields include: "Owner" (a dropdown menu showing a user icon), "Repository name" (a text input field), and "Description (optional)" (a text input field). Below these fields, there are two radio button options: "Public" (selected) and "Private". The "Public" option is described as "Anyone on the internet can see this repository. You choose who can commit." and the "Private" option is described as "You choose who can see and commit to this repository." At the bottom, there is a section titled "Initialize this repository with:" followed by the instruction "Skip this step if you're importing an existing repository." and a checkbox labeled "Add a README file" which is checked. A red arrow points to the "Add a README file" checkbox. Below the checkbox, it says "This is where you can write a long description for your project. [Learn more.](#)"

No campo de licença, *Choose a license*, adicionaremos a licença do MIT.



Por fim, clicaremos no botão de criar repositório:

Create repository

Mas ainda não temos nenhum arquivo para ser adicionado em nosso repositório. Vamos iniciar a criação dos arquivos que vamos adicionar ao repositório.

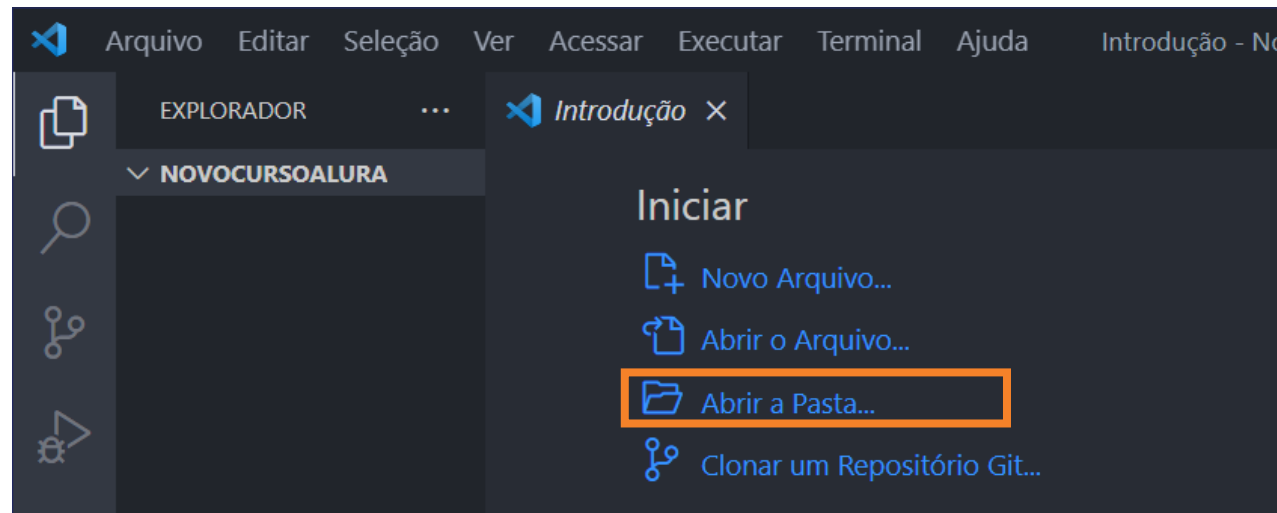
Devemos agora iniciar o projeto. Para isso, usaremos um editor de código chamado Visual Studio Code, por meio do qual podemos criar sites.



Visual Studio Code

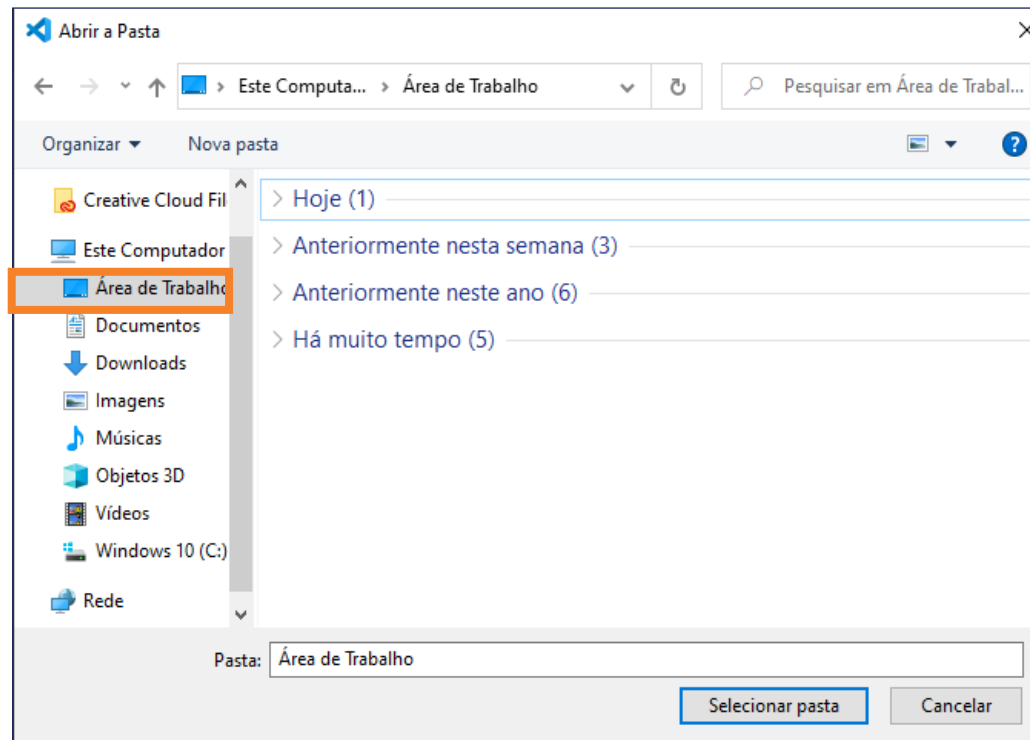
Aplicativo

Ao abrirmos o Visual Studio Code, ou VSCode que é sua abreviação, devemos criar uma pasta para que todos os arquivos sejam salvos em um só lugar. Então, vamos clicar em *Abrir a Pasta*.

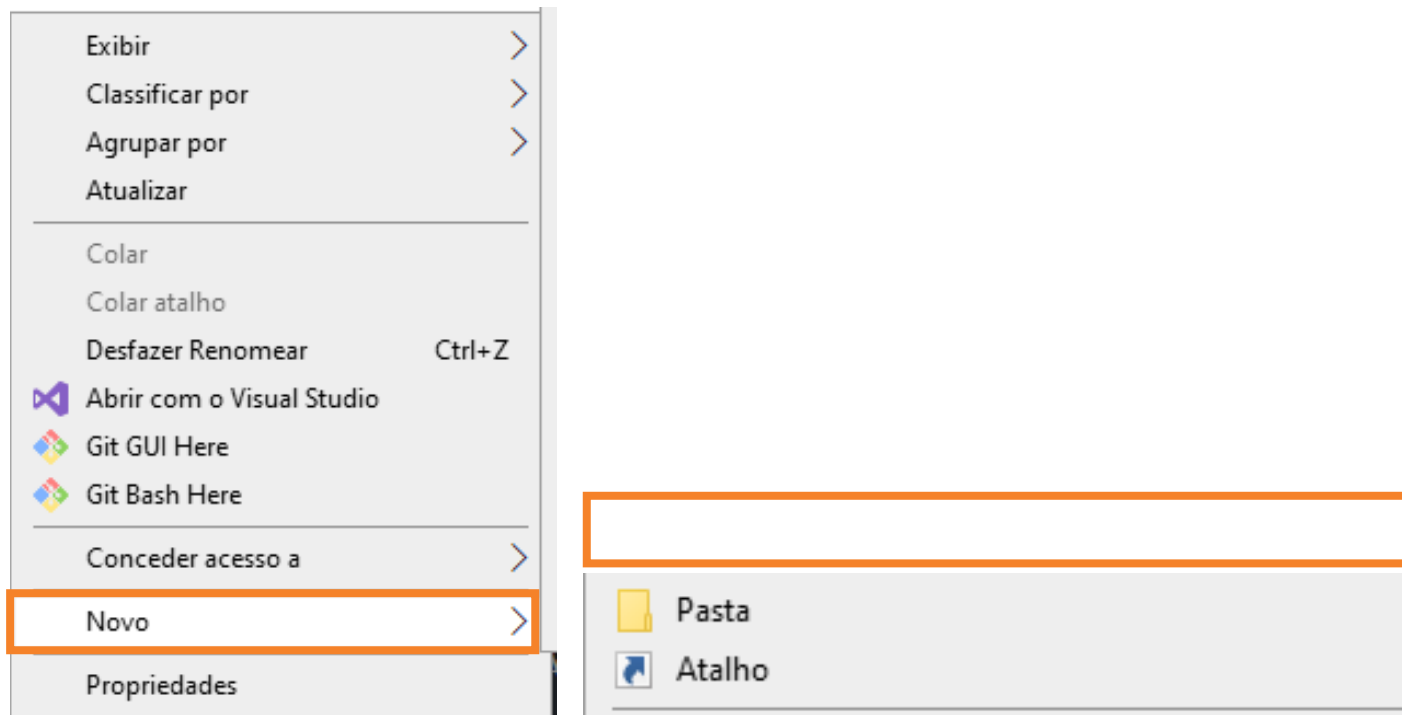


Para criar uma pasta, devemos seguir estes passos:

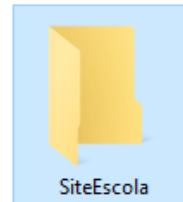
- Selecionar onde a pasta será criada; como sugestão, clique em *Área de trabalho* ou *Documentos*.



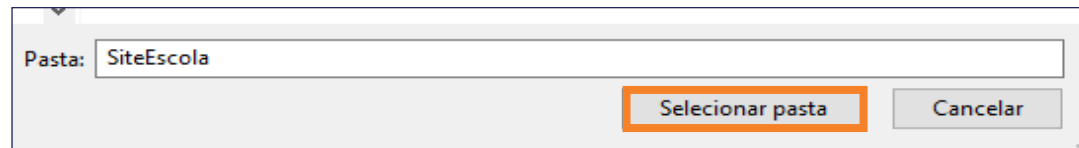
- Clicar com o botão direito do mouse em qualquer parte da área branca da janela que abrirmos, e então clicar em *Novo -> Pasta*.



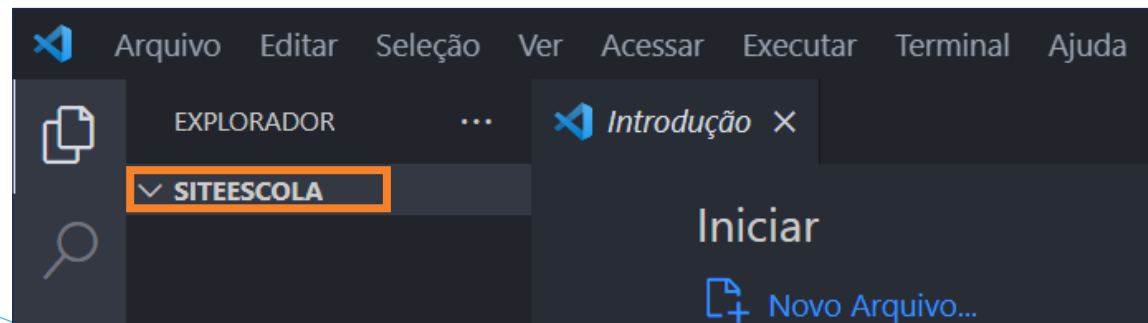
- Em seguida, criar um nome para a pasta; como sugestão, nomear como “SiteEscola”:



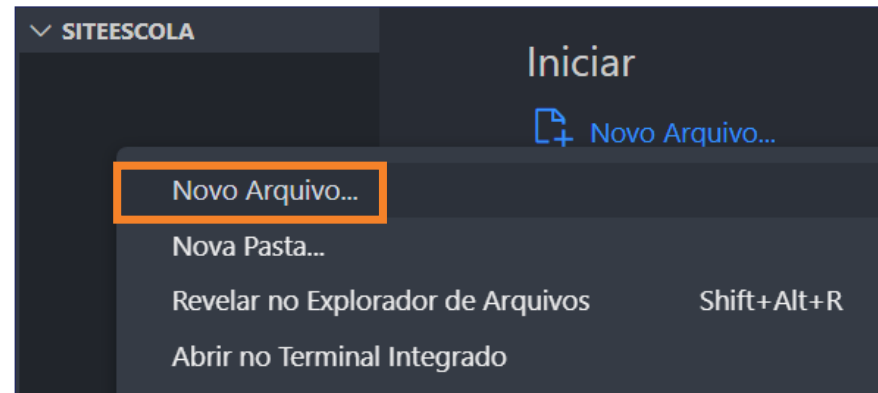
- Clicar no botão *Selecionar pasta*:




A pasta será aberta no VSCode e já podemos iniciar a criação do site.



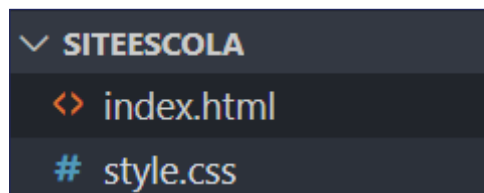
Vamos criar o arquivo index.html. Clicaremos com o botão direito do mouse na coluna à esquerda da tela. Em seguida, clicaremos em *Novo Arquivo*.



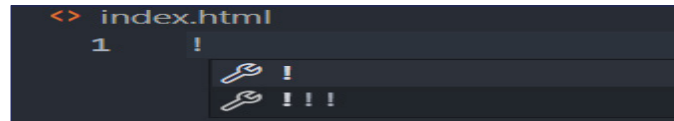
Um campo em branco será aberto. Digitaremos “index.html” para criar um arquivo.

 index.html

Faremos o mesmo processo para criar um arquivo style.css. Clicaremos com o botão direito do mouse em *Novo Arquivo*. Em seguida, em alterar o nome para “style.css”.



Vamos criar a estrutura do arquivo.html. Para isso, digitaremos o ponto de exclamação “!” na tela e apertaremos a tecla *Enter*.



```
<> index.html
1      !
```

Logo em seguida, surgirá uma estrutura padrão de código de uma página HTML. Vamos entendê-la melhor em seguida.

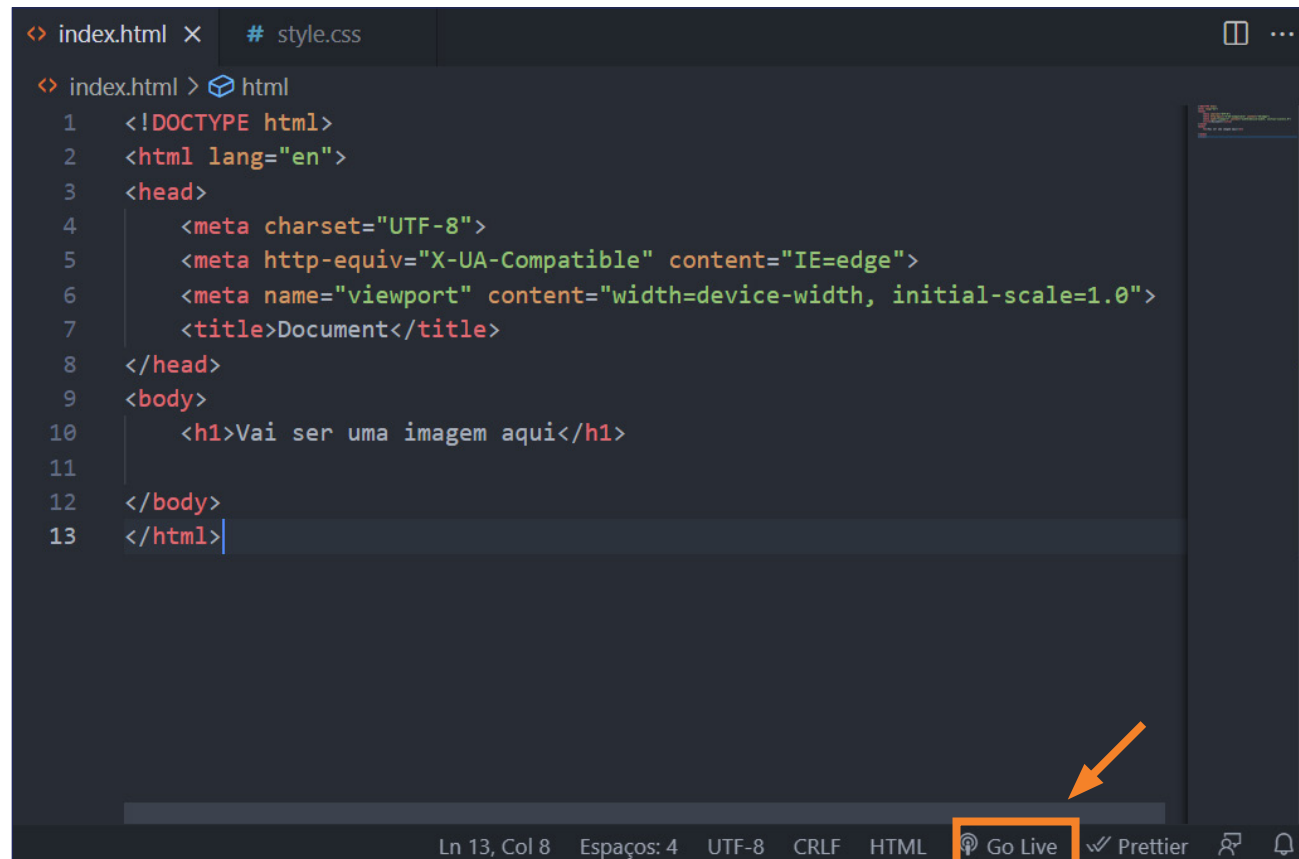


```
<> index.html x # style.css
<> index.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta http-equiv="X-UA-Compatible" content="IE=edge">
6    <meta name="viewport" content="width=device-width, initial-scale=1.0">
7    <title>Document</title>
8  </head>
9  <body>
10
11 </body>
12 </html>
```


Vamos praticar a escrita no arquivo index.html. iniciaremos escrevendo uma frase “vai ser uma imagem aqui”. Para isso, usaremos a tag `<h1>`, que corresponde a uma tag de título. Essa frase deve ser adicionada dentro do espaço da tag `<body>`, que corresponde ao corpo da página.

```
<body>  
<h1>Vai ser uma imagem aqui</h1>  
</body>
```

Para vermos o resultado no site, vamos clicar em *Go Live* no canto inferior direito da tela, que é uma extensão do VSCode que permite testar a página criada.



The screenshot shows the Visual Studio Code editor interface. The top bar displays the active file 'index.html' and the editor mode '# style.css'. The main editor area shows the following HTML code:

```
<> index.html >  html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta http-equiv="X-UA-Compatible" content="IE=edge">
6      <meta name="viewport" content="width=device-width, initial-scale=1.0">
7      <title>Document</title>
8  </head>
9  <body>
10     <h1>Vai ser uma imagem aqui</h1>
11
12 </body>
13 </html>
```

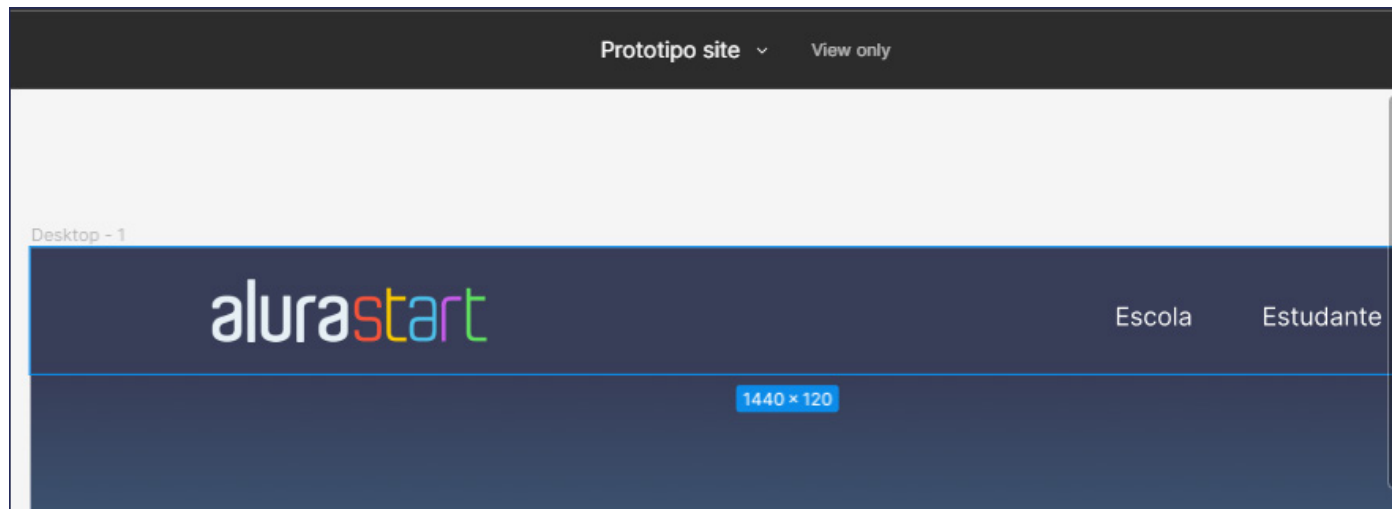
The bottom status bar shows the current position 'Ln 13, Col 8', settings 'Espaços: 4', encoding 'UTF-8', line endings 'CRLF', language 'HTML', and a 'Go Live' button (a lightning bolt icon) which is highlighted with an orange box and an orange arrow. Other icons in the status bar include a checkmark for 'Prettier', a search icon, and a bell icon.

Ao carregarmos, a seguinte página será exibida:



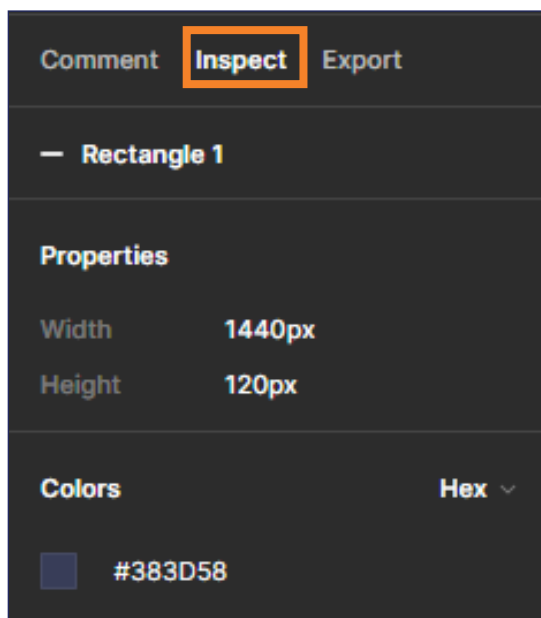
Ou seja, o arquivo index.html está funcionando. Para testarmos o arquivo style.css, vamos conferir uma propriedade em nosso protótipo no Figma.com.

Nosso protótipo no Figma tem um cabeçalho na cor azul e uma imagem:

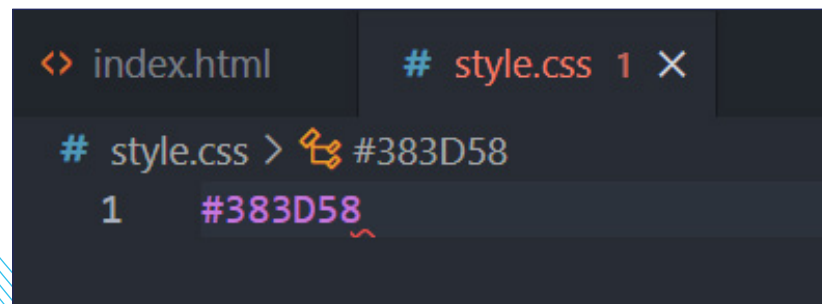


No arquivo index.html, indicamos por meio do texto “Vai ser uma imagem aqui” onde a imagem deve ficar.

Para conferirmos se o arquivo `style.css` está funcionando, vamos atribuir a cor de fundo azul, a mesma do protótipo, em nossa página HTML. Para fazermos isso, clicaremos no cabeçalho do protótipo no Figma, e então, na coluna à direita da tela, clicaremos na aba *Inspect*.



Perceberemos que a cor azul possui um código hexadecimal. Vamos digitar esse mesmo código hexadecimal no arquivo `style.css`.



No entanto, apenas copiar o código da cor não será suficiente para aplicá-la. Devemos utilizar a tag `<body>` e envolver o código entre as chaves dessa tag. Quando escrevemos o código no arquivo `style.css` não usamos o símbolo `<>`.

```
body {  
  #383D58  
}
```

Feito isso, vamos adicionar a propriedade `background-color`, que significa “cor do fundo”, para que esta receba o código de cor correto.

```
body {  
  background-color: #383D58;  
}
```

Ao carregarmos a página, a página não sofreu quaisquer alterações.

Vai ser uma imagem aqui

Isso acontece porque não relacionamos o arquivo `style.css` com o arquivo `index.html`.

Para adicionarmos o arquivo style.css como parte da página HTML, utilizamos a tag `<link>` e logo em seguida a propriedade `rel`, que identifica o tipo de arquivo. Por fim, apontamos qual arquivo deve ser inserido por meio da propriedade `href`.

```
<title>Document</title>  
<link rel="stylesheet" href="style.css">
```

Ao carregarmos a página, podemos observar que a cor de fundo foi aplicada.

Vai ser uma imagem aqui

Seguindo o protótipo, ainda no cabeçalho temos outros elementos de texto, a seção Escola e a seção Estudante; vamos criá-las no arquivo index.html.



Para isso, usaremos a tag ``, que cria uma lista não ordenada de elementos.

```
<h1>Vai ser uma imagem aqui</h1>
<ul>

</ul>
```

Em uma lista de uma página HTML, cada elemento é representado pela tag ``. Como temos dois elementos na lista, vamos adicionar duas vezes a tag ``.

```
<ul>
  <li>Estudante</li>
  <li>Escola</li>
</ul>
```

Ao carregarmos a página, podemos observar que os elementos de texto que representam as seções foram criados e estão dispostos em uma lista.

Vai ser uma imagem aqui

- Estudante
- Escola

Para mantermos a organização do site, vamos definir que todos os elementos destacados na imagem fazem parte da mesma seção cabeçalho. Para isso, temos uma tag que representa o cabeçalho de uma página, e ela se chama `<header>`.

Então, vamos envolver todo o código criado na tag `<header>`.

```
<header>
  <h1>Vai ser uma imagem aqui</h1>
  <ul>
    <li>Estudante</li>
    <li>Escola</li>
  </ul>
</header>
```

A cor da fonte dos elementos do cabeçalho deve ser branca. Para estilizá-la vamos até o arquivo style.css e definiremos a cor **white**, que significa “branco” em inglês, à propriedade **color**, que significa “cor” em inglês.

```
body {  
  background-color: #383D58;  
  color: white;  
}
```

Ao carregarmos a página, podemos observar que as alterações foram realizadas.

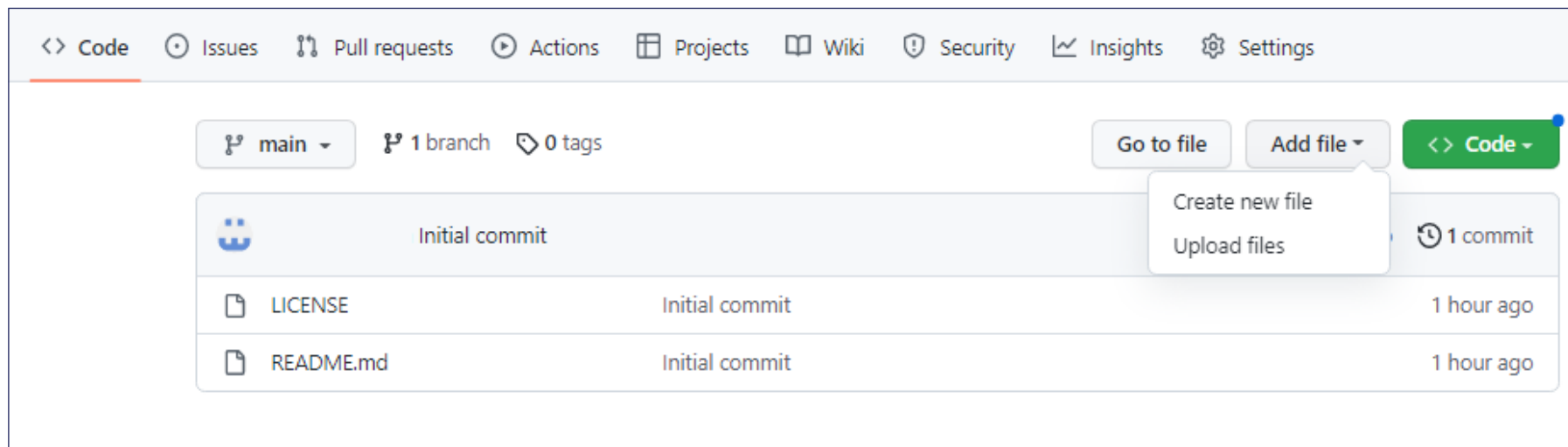
Vai ser uma imagem aqui

- Estudante
- Escola

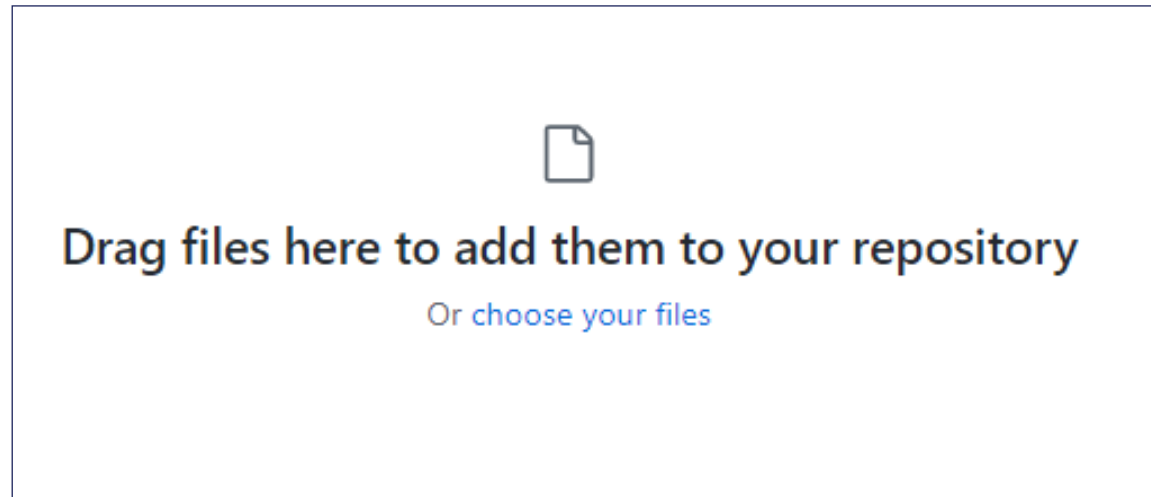
Para garantirmos que as alterações só sejam feitas nos elementos do cabeçalho, alteraremos a tag `<body>` no arquivo `style.css` para `<header>`. Desta forma, os próximos elementos adicionados poderão ter um padrão de cor diferente.

```
header {  
  background-color: #383D58;  
  color: white;  
}
```

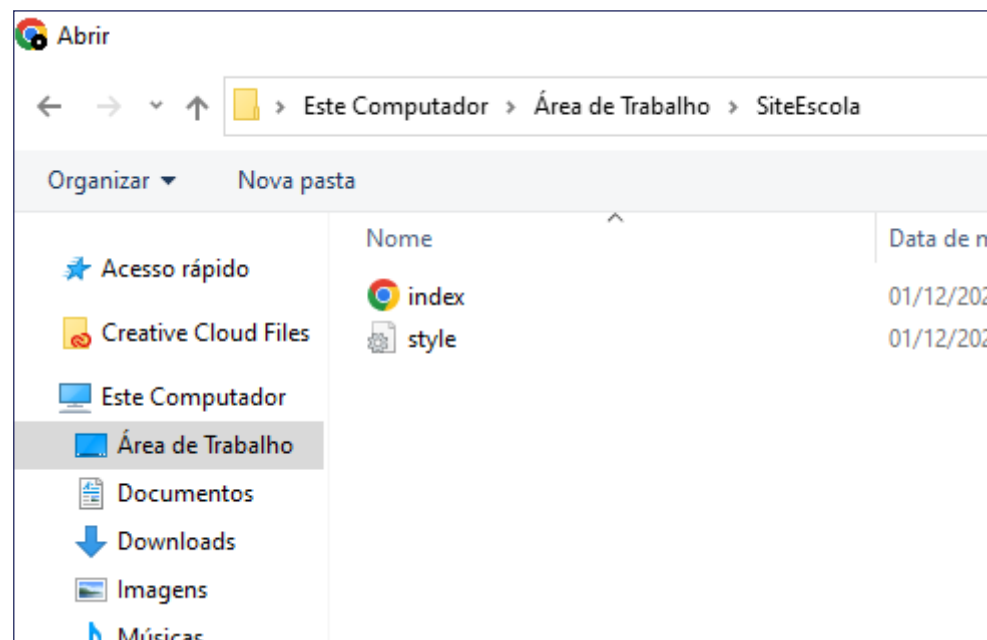
Por fim, vamos adicionar os arquivos desenvolvidos no repositório que criamos no início desta aula. Para isso, devemos ir até o site do GitHub e buscar pelo repositório. Então, clicar no botão *Add file* → *Upload files*.



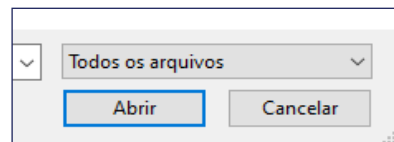
Clicaremos em escolher arquivos, na opção *Choose your files*.



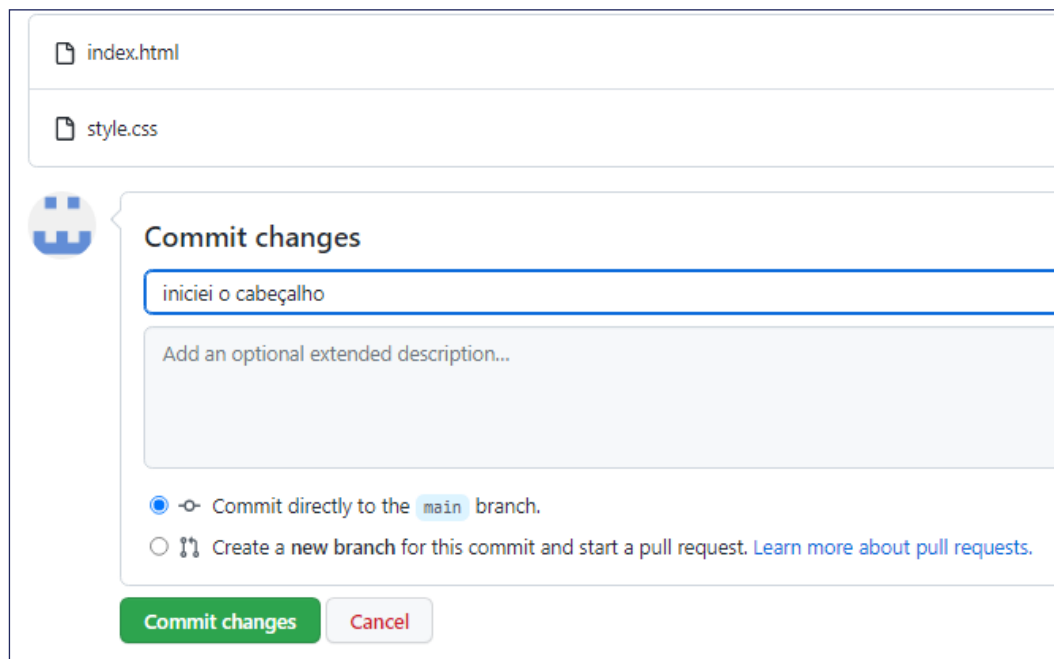
Buscaremos pelos arquivos criados: index.html e style.css.



Selecionamos e clicamos em *Abrir*.



Os arquivos serão carregados. Agora, no campo *Commit changes*, que em inglês significa “enviar alterações”, escrevemos brevemente sobre o assunto dos arquivos, isto é, o que você produziu. Como sugestão, vamos escrever “inicie o cabeçalho”. Por fim, clicaremos no botão verde, chamado *Commit changes*.

A screenshot of the GitHub web interface for committing changes. At the top, two files are listed: 'index.html' and 'style.css'. Below this is the 'Commit changes' section, which includes the GitHub logo, a text input field containing 'inicie o cabeçalho', and a larger text area with the placeholder 'Add an optional extended description...'. At the bottom of this section are two radio button options: 'Commit directly to the main branch.' (which is selected) and 'Create a new branch for this commit and start a pull request. Learn more about pull requests.' At the very bottom are two buttons: a green 'Commit changes' button and a grey 'Cancel' button.

Com esses passos iniciamos o desenvolvimento do site da Escola e armazenamos nosso arquivo em um repositório do GitHub.

Desafio

Para explorar e praticar as propriedades e tags aprendidas, crie algumas frases sobre a escola e tente trocar a cor da fonte para duas cores diferentes. Para isso, você deverá utilizar tanto o arquivo index.html quanto o arquivo style.css. O objetivo deste desafio é incentivar sua autonomia e criatividade no desenvolvimento de uma página HTML.

Avalie este material acessando o link: <https://forms.gle/EcEZdj59zGS9QTwy9>