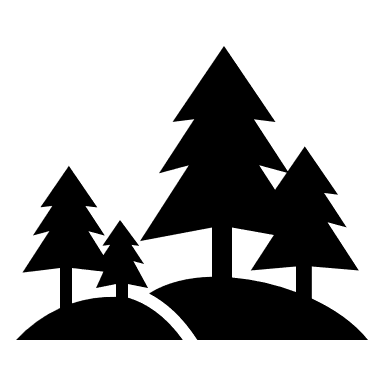
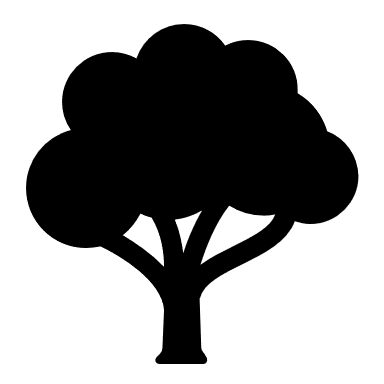
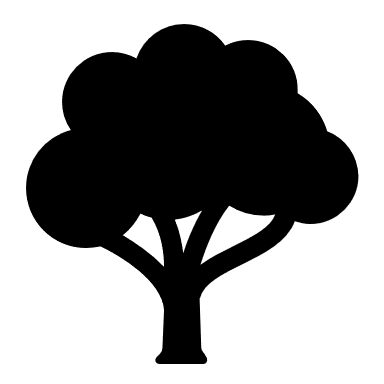
“Pequenas histórias e grandes Aplicativos”

Site Dinâmico de Receitas (Front e Back End)

Front End Back End

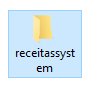
**História 01**

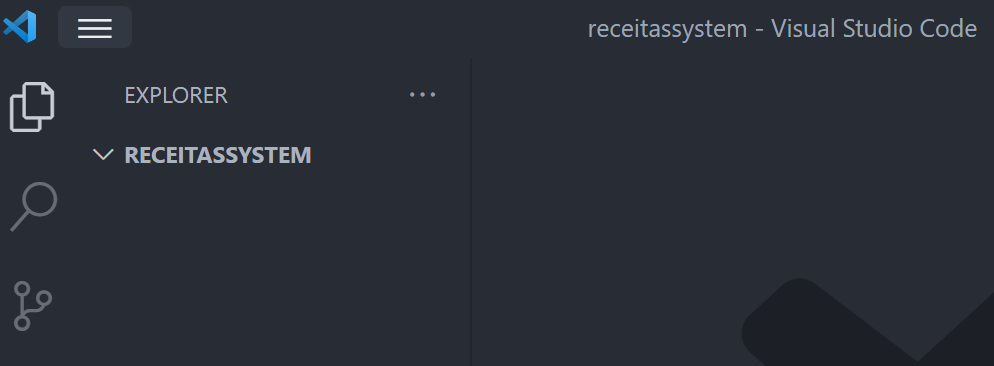
O usuário irá clicar em um botão para aparecer uma janela pop up.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Árvore decídua | **BackEnd** | | | |
|  | | **Branch : Módulo de Autenticação e Autorização** | | |
| desenho vetorial simples desenhado à mão. raminho com folhas, silhueta ...  **Esse ramo do sistema deve permitir autenticação e autorização do usuário na área de backend do site.** | | | | |
| **FolhaHistórias** | | | | |
| **Id** | | **Quem** | **Ação** | **Propósito** |
| 1 | | Usuário | Clica em um botão com o título **“Logar”** | Para aparecer uma mensagem “seja bem-vindo ao nosso sistema” em um pop up e tendo como título “Receitas System”.  [vídeo demonstrativo](https://vimeo.com/855102452?share=copy) |
| 2 | | Usuário | Vai digitar o login e senha | Para autenticar o seu acesso (use para teste login:master senha: 123) |

**Habilidades – História 01**

**Desenvolvimento Front e Back End (folha 01)**

**Passo 1**: Crie uma pasta 

**Passo 2:** Selecione a pasta 

**Passo 3:** Crie um arquivo index.html

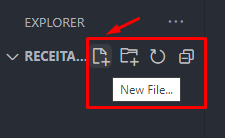


Figura - cuidado ( o primeiro "+" é arquivo, enquanto segundo "+" é pasta)

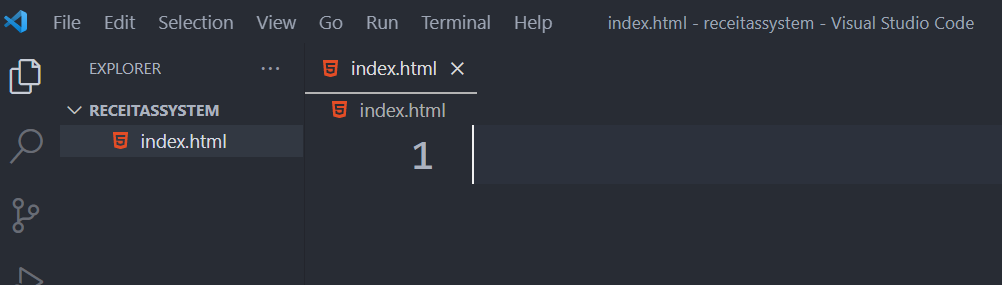
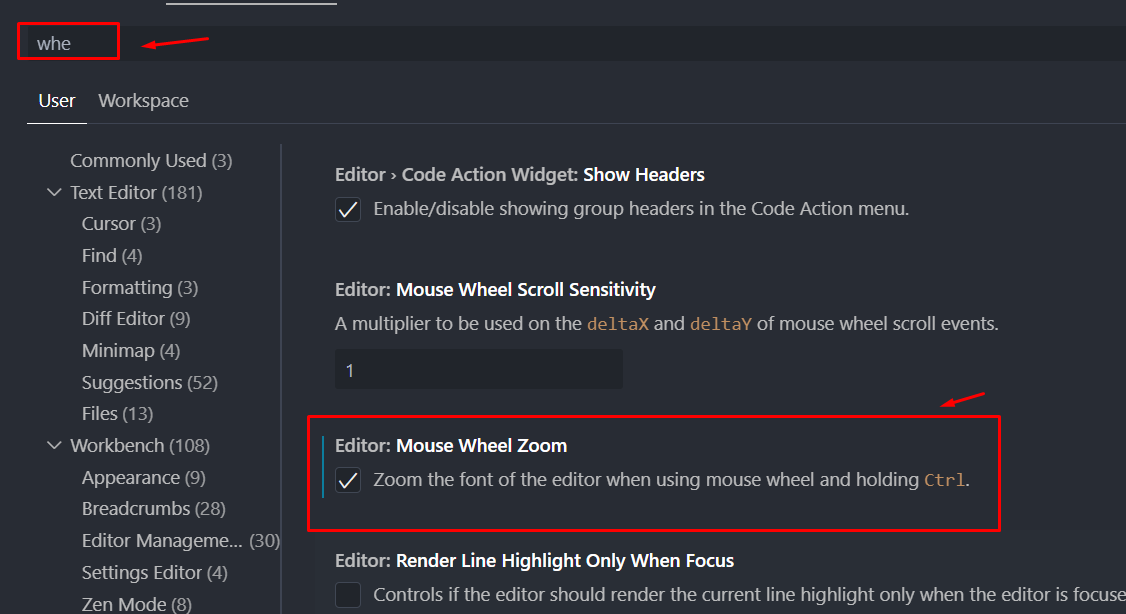


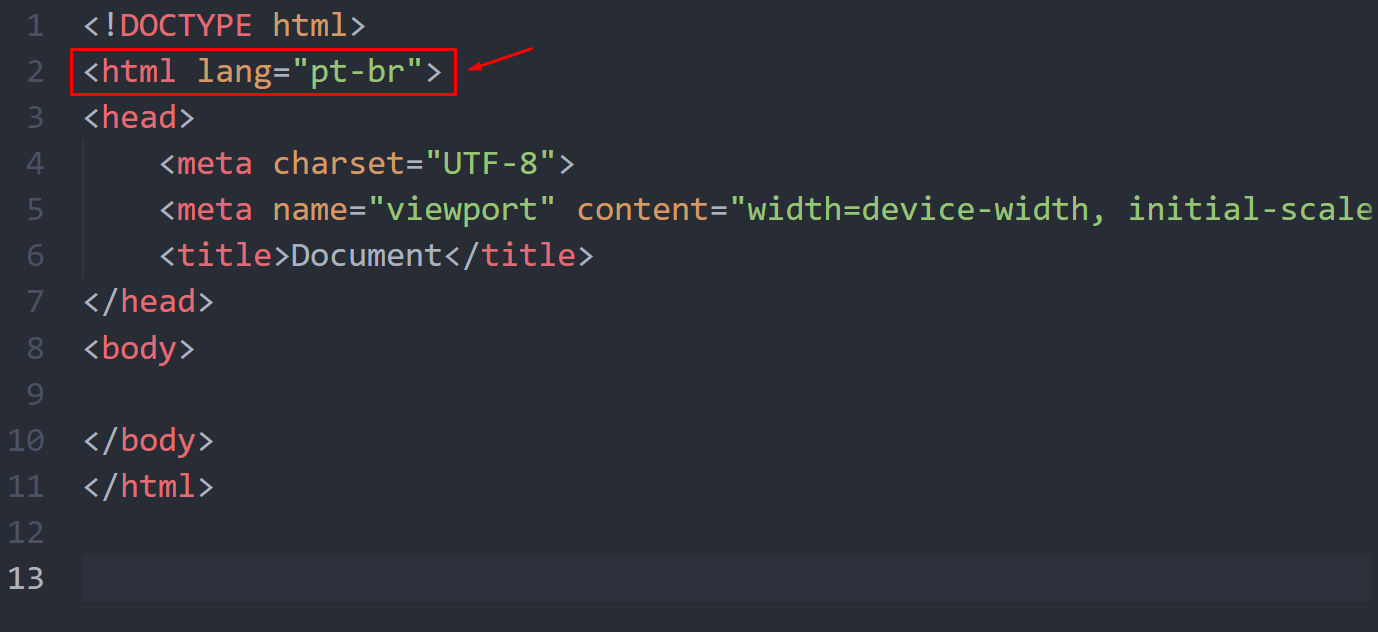
Figura - Editor de Código x IDE

Dica da Ferramenta: Zoom no Código



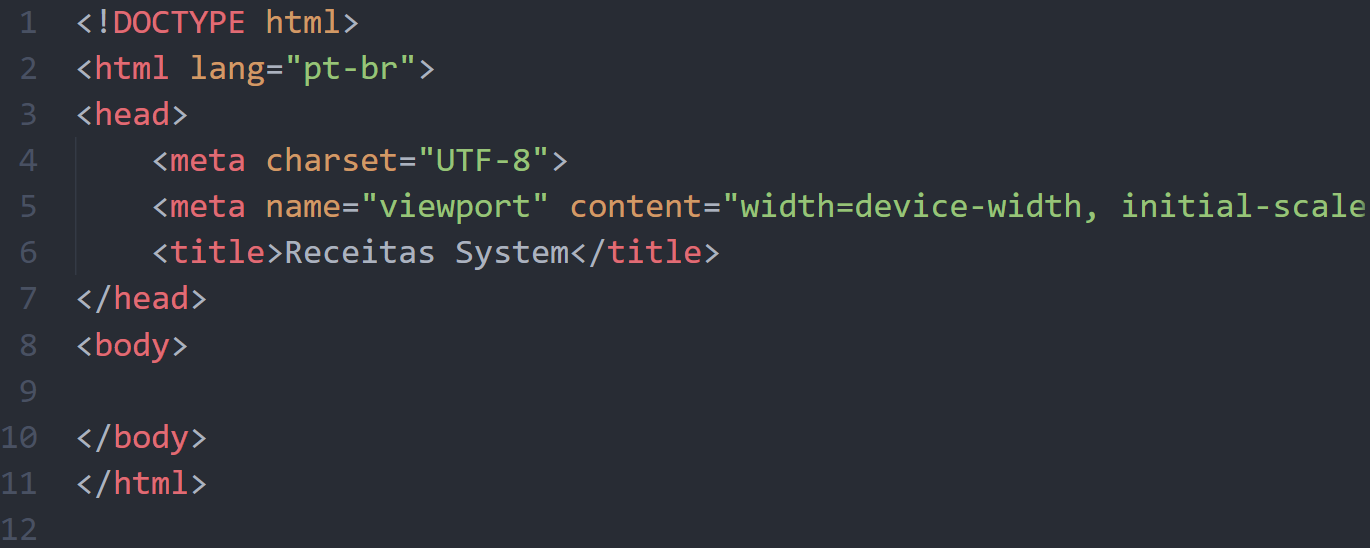
Passo 4:

Configure para o português Brasil



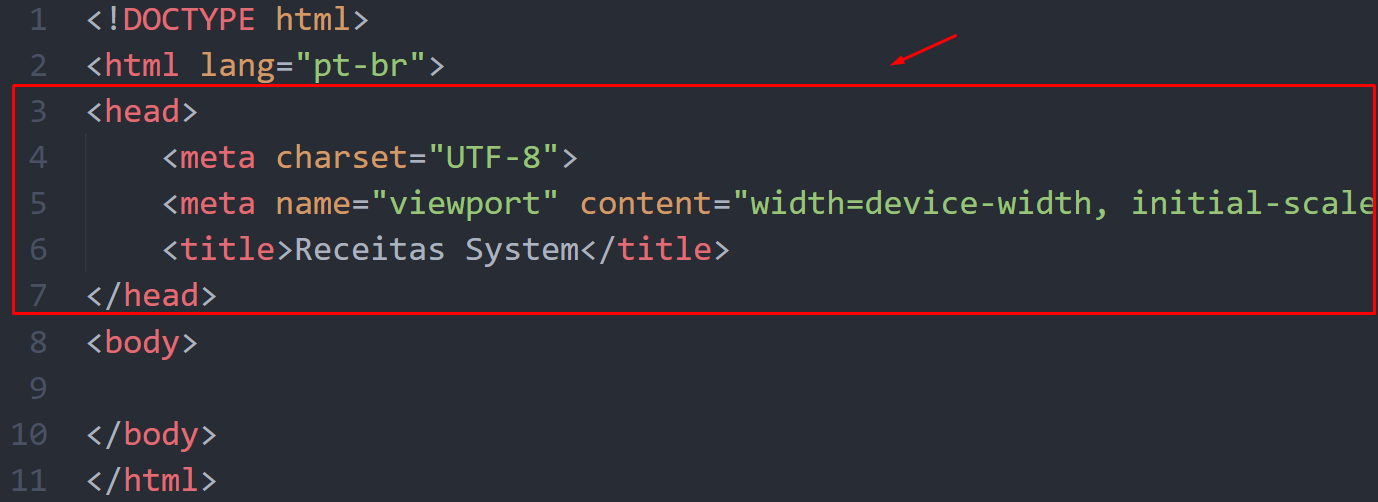
Passo 5:

Mude o título



Passo 6:

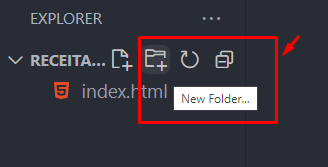
Entenda para que serve head (robô) e body (clientes e usuários)

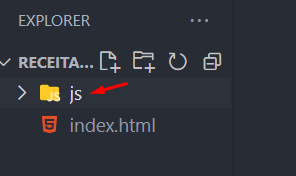


Passo 7: Vamos para o corpo do site

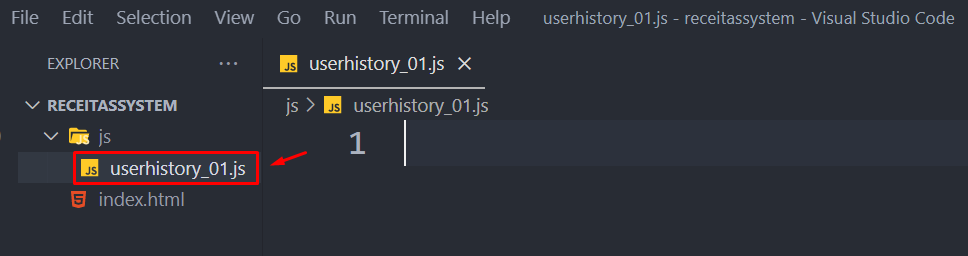


Passo 8: Crie uma pasta chamada js





Passo 9: Crie um arquivo userstory\_01.js



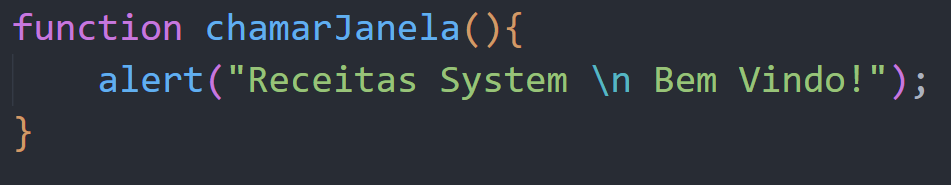
Passo 10: Volte no visual e dê um retoque no seu código

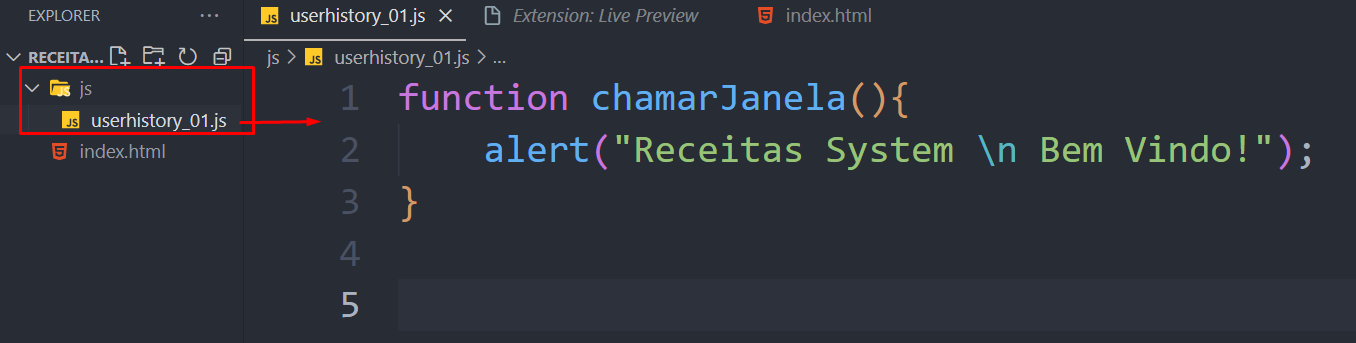


**Na *linha 10* colocamos um título para o botão.**

**Na *linha 11* inserimos o arquivo JavaScript (Ecma Script).**

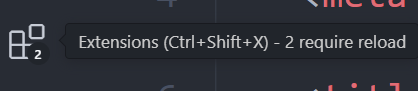
Passo 11: Crie uma função para mostrar o pop up

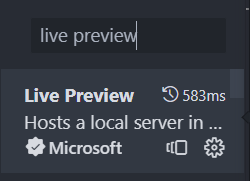


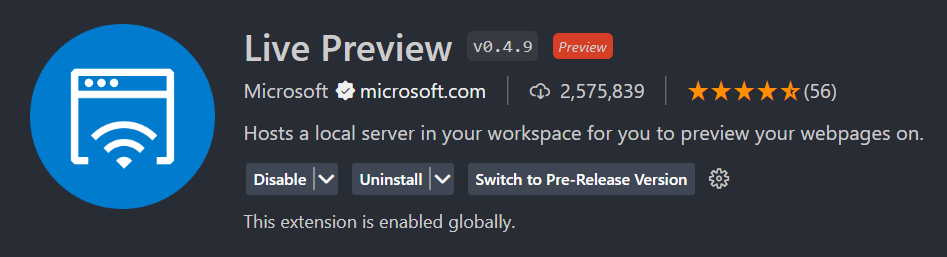


Executando e testando:

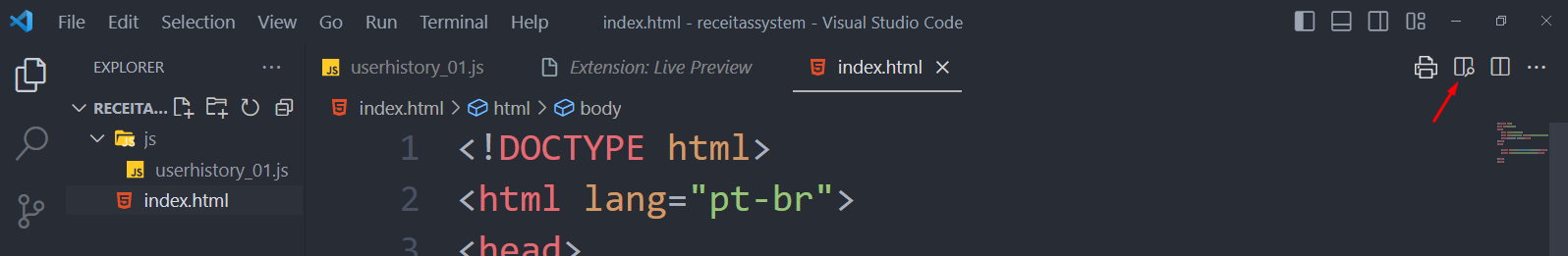
Localize a extensão:



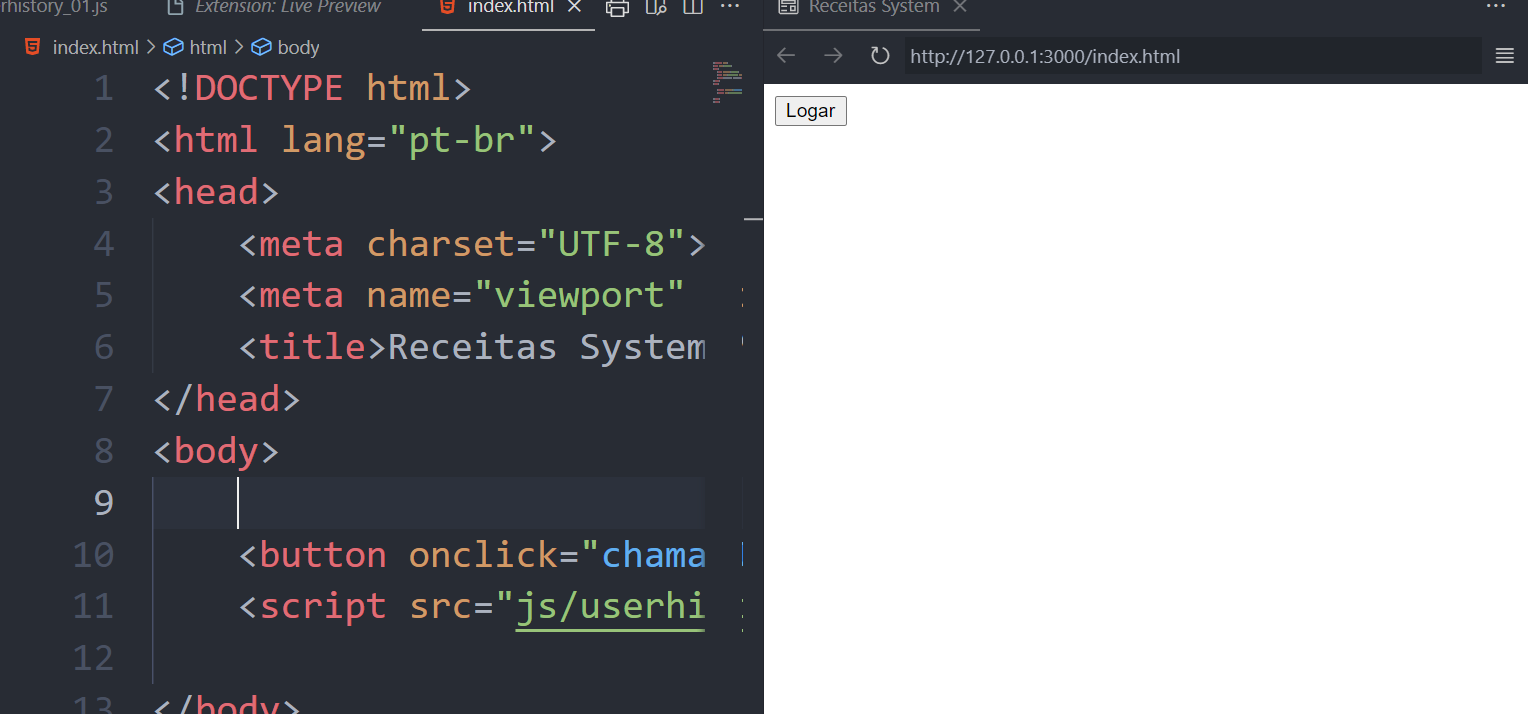




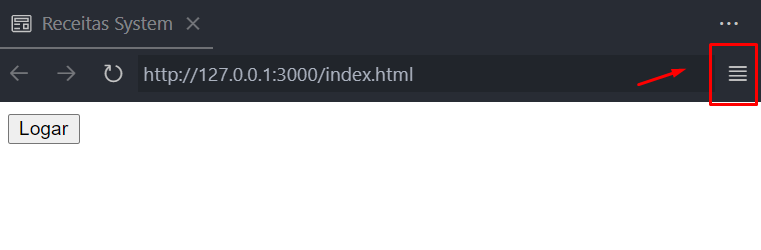
Localize a lupa

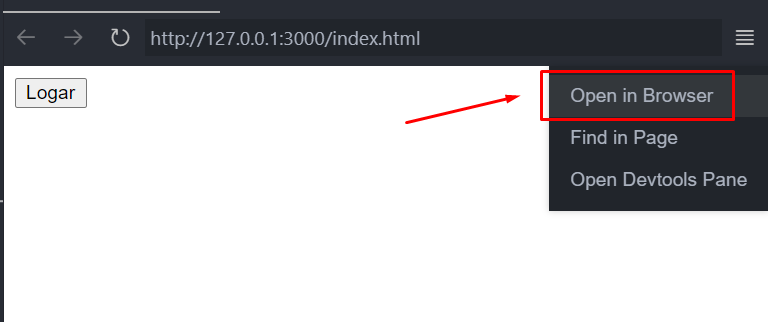


Clique na lupa



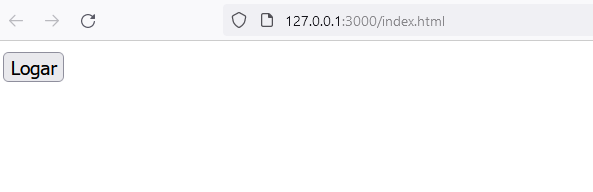
Clique nas barrinhas na lateral direita

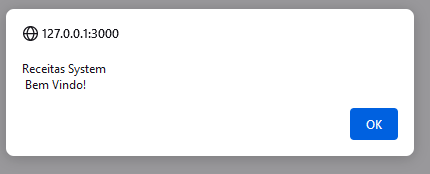




Veja o resultado

Clique no botão:



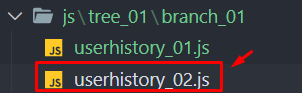


**Atitudes**

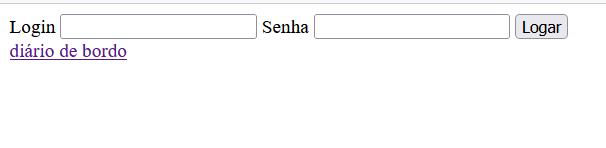
* **Paciência**:
  + São muitos erros.
  + Mas a vida é assim mesmo!
* **Atenção aos detalhes!**
  + Tem erro de ponto vírgula
  + **P**ascal**C**asing e **c**amel**C**ase
    - Para função do JavaScript usamos camelCase
      * function **c**hamar**J**anela( ) .....
* **Atenção aos detalhes!**
  + “As aspas “

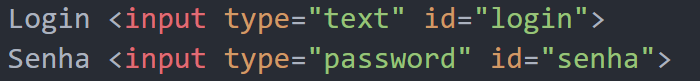
**Habilidades – História 02**

**Passo 1: Organizar as pastas**

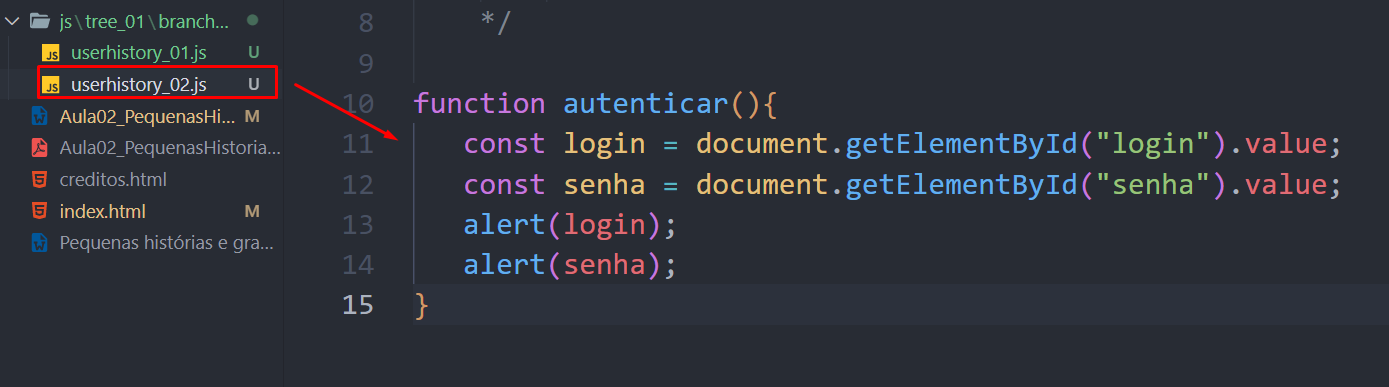


**Passo 2: Criar a tela de login**





**Passo 3: Criar um novo arquivo para receber os dados e validar esses dados.**



**Linha 10: cabeçalho da função**

**function <<nomedafuncao>>( )**

**Linha 10 e 15: o corpo da função**

**{**

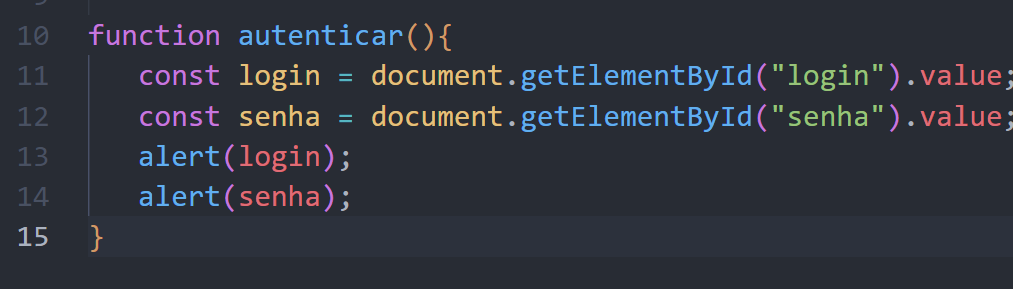
**}**

**function <<nomedafuncao>> () 🡪cabeçalho da função**

**{**

**//corpo da função**

**}**



**Linha 11: declara uma constante para armazenar um valor.**

**Linha 12: declara uma constante para armazenar um valor**

**Linha 13: mostra o valor**

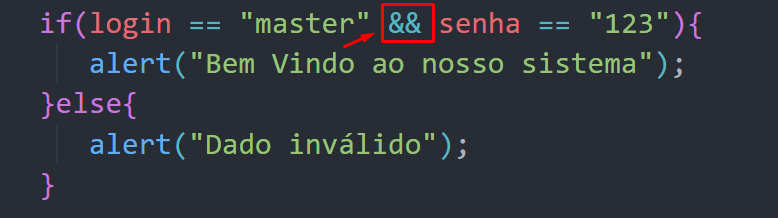
**Linha 14: mostra o valor**

**Lógica de Programação**

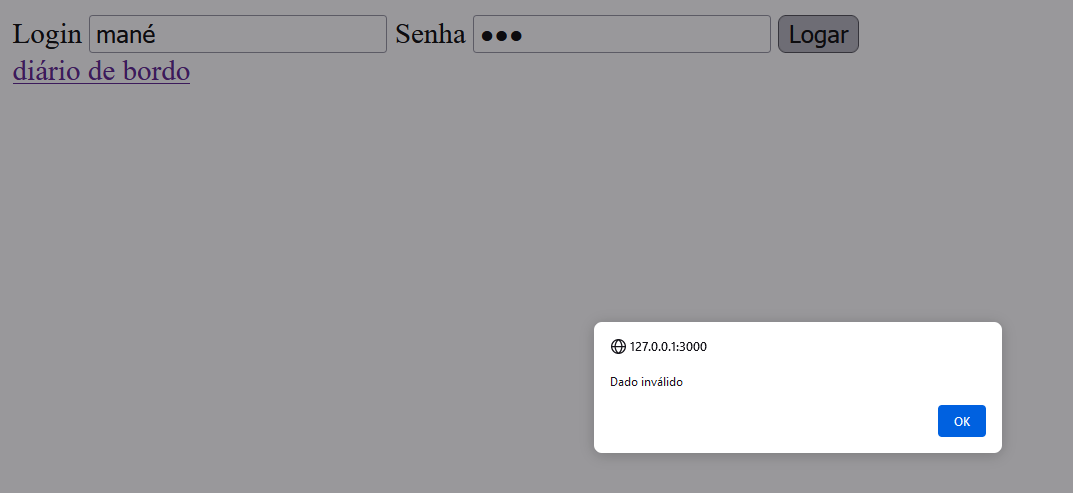
* **Aristóteles (lógica aristotélica) – Nem**
* **Lógica Booleana – Senai**
* **George Boole é discipulo de Aristóteles.**

|  |  |  |
| --- | --- | --- |
| **Rica** | **Bonita** | **R E B** |
| **V** | **V** | **V** |
| **V** | **F** | **F** |
| **F** | **V** | **F** |
| **F** | **F** | **F** |

|  |  |  |
| --- | --- | --- |
| **Principe** | **Encantado** | **R OU B** |
| **V** | **V** | **V** |
| **V** | **F** | **V** |
| **F** | **V** | **V** |
| **F** | **F** | **F** |

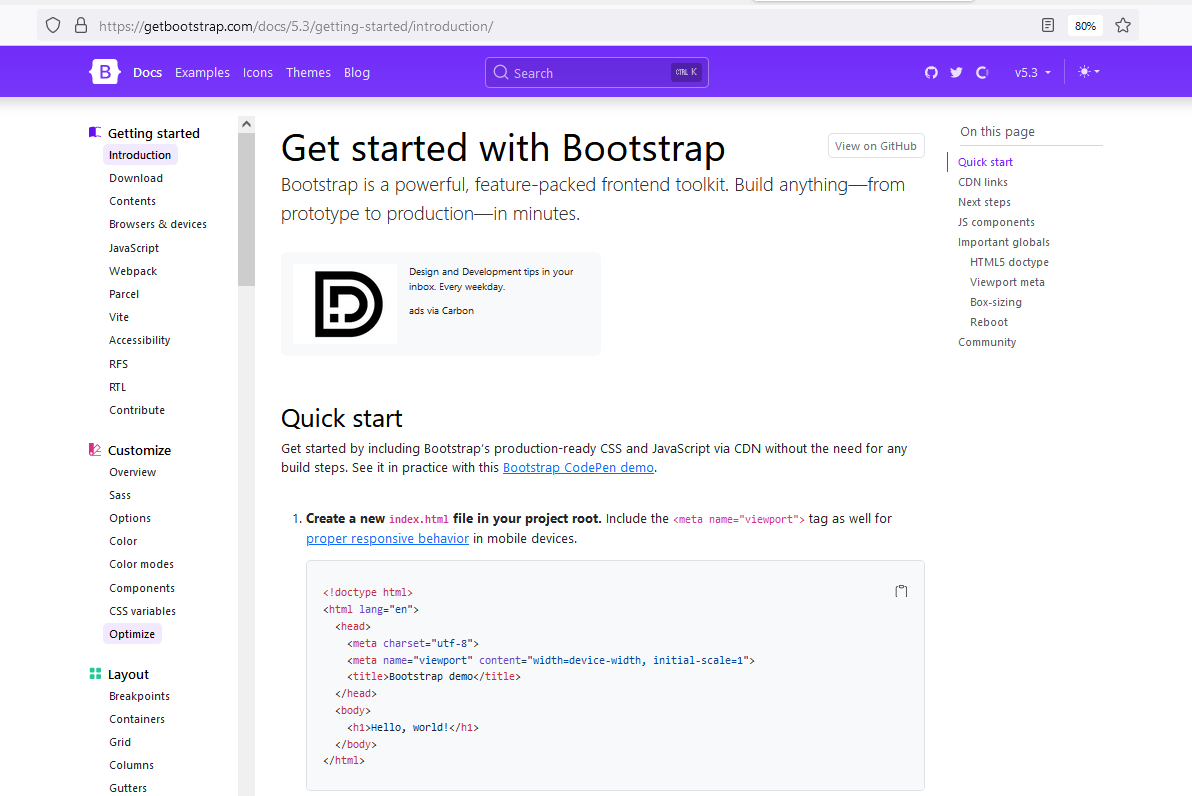






**“Porque nem só de código vive o homem, mas de todo design do design system”**

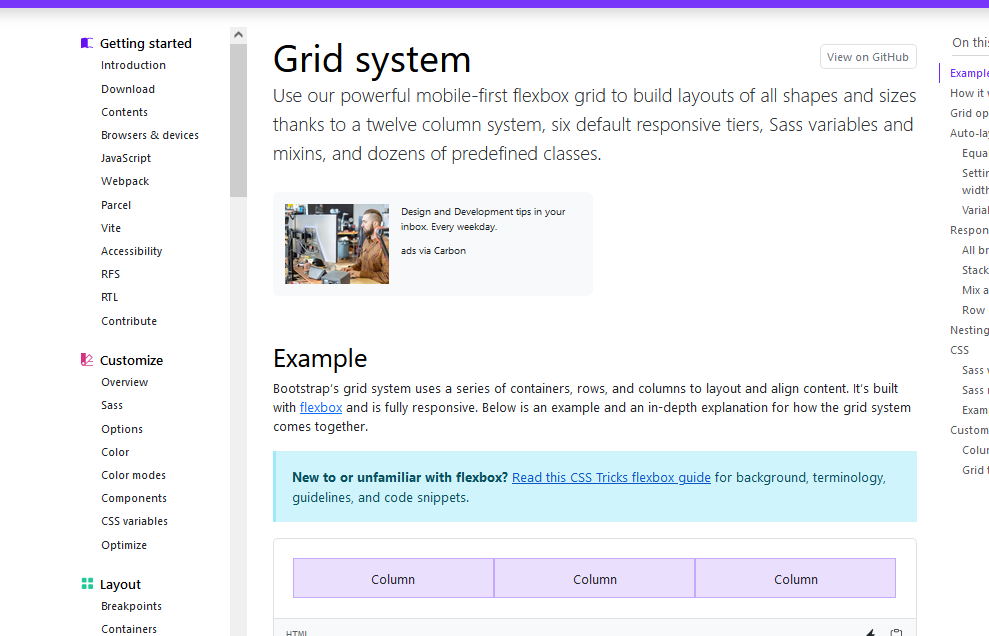
**Romullus 3.17**



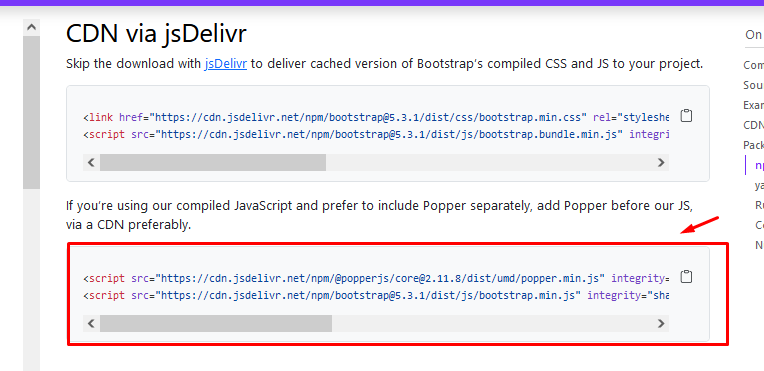
**É um biblioteca (CSS,JavaScript) – Mobile First – Responsividade**

**Grid (com 12 colunas)**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** | **11** | **12** |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |



**Instalando Bootstrap**

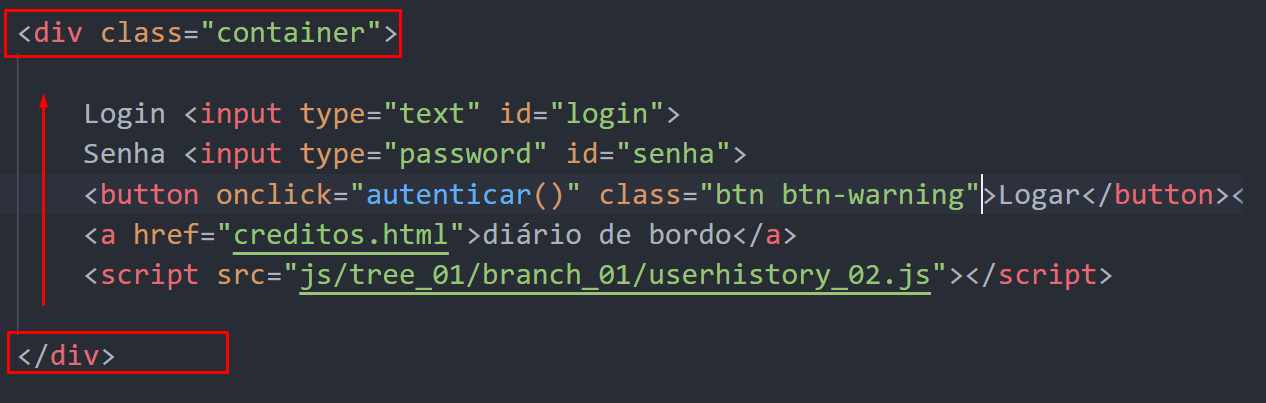


**Colar**



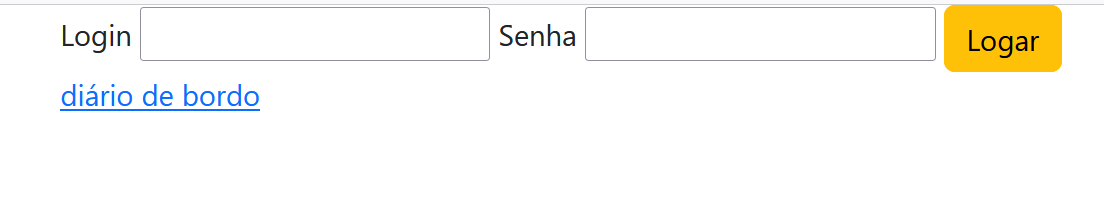
Se você colou então tudo ok.

Coloque tudo no container:

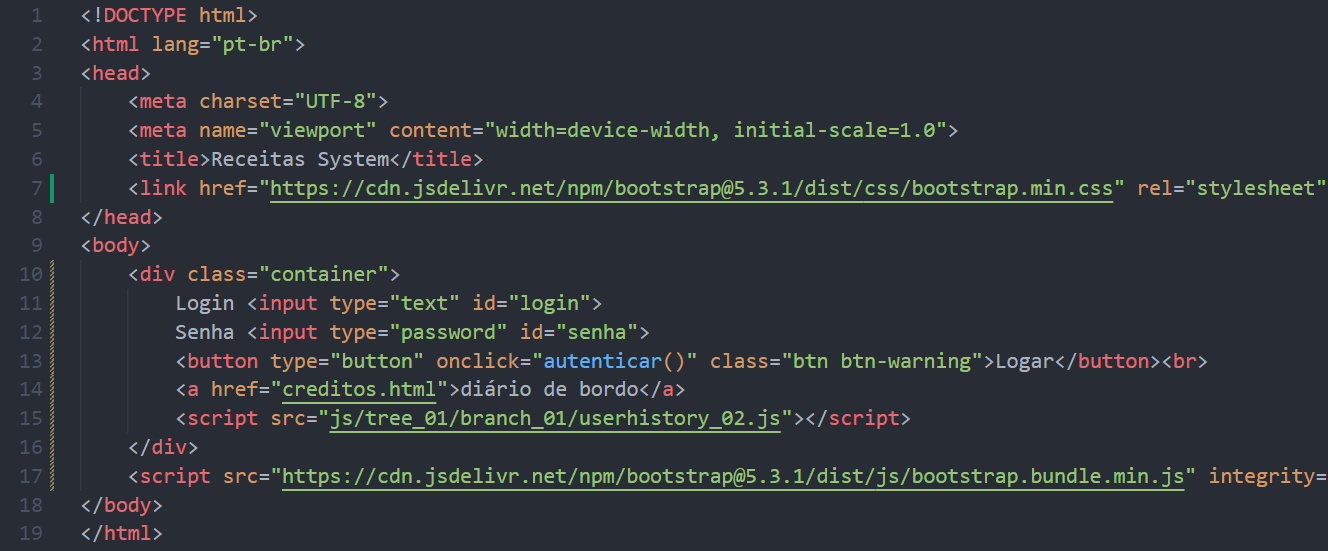


Deixei seu **botão mais bonito**:





Vamos ver o código completo:



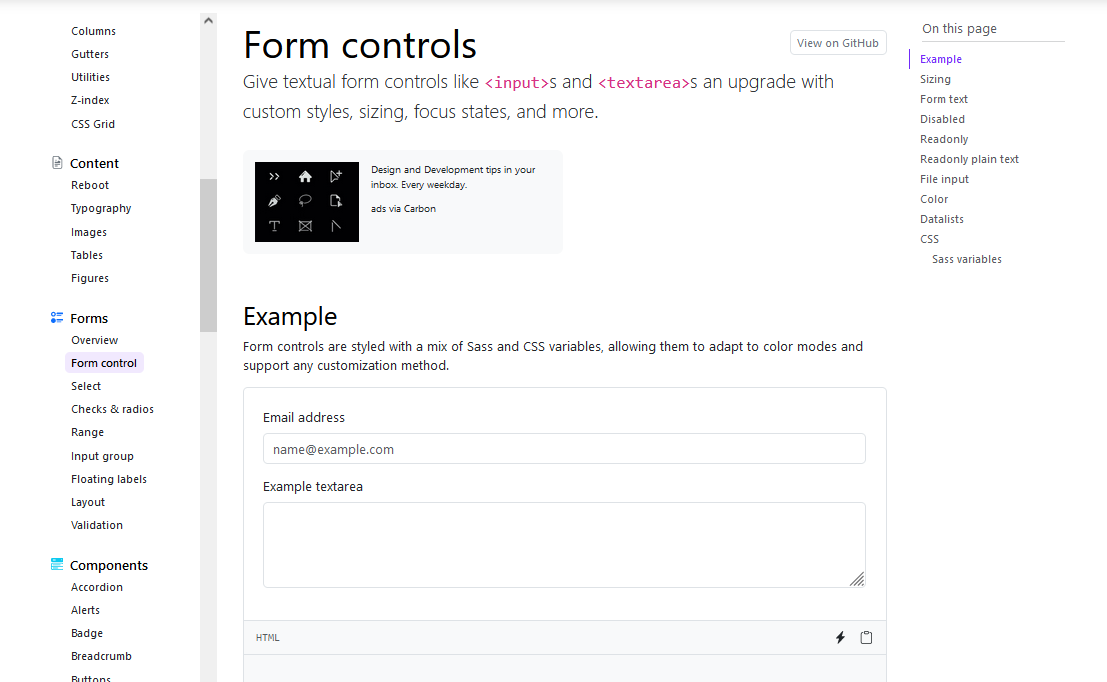
**Linha 7**: importa o css do bootstrap via cdn

**Linha 10**: container (colocar tudo)

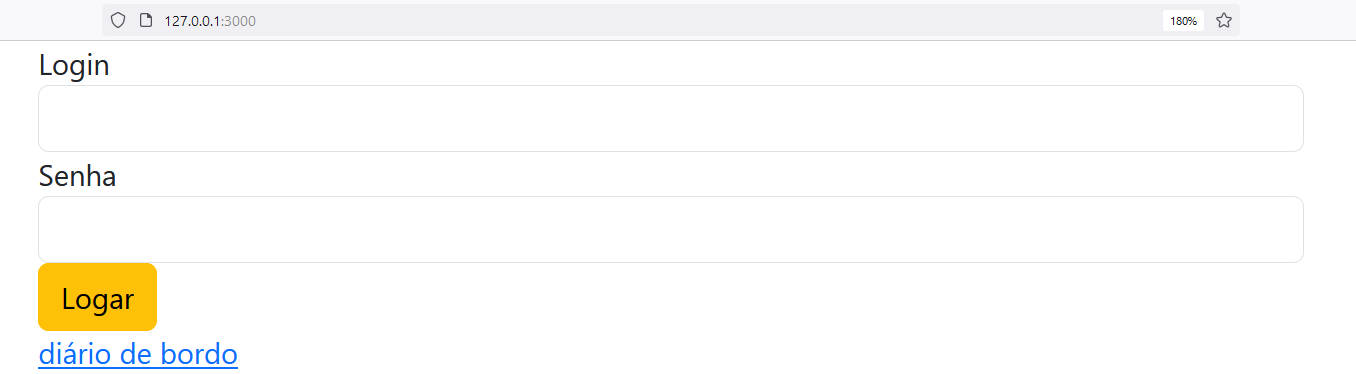
**Linha 13**: deixar o seu botão bonito class**=” btn btn-warning”**

**Linha 17**: importar o javascript para o bootstrap

**5 minutos para navegar no site**

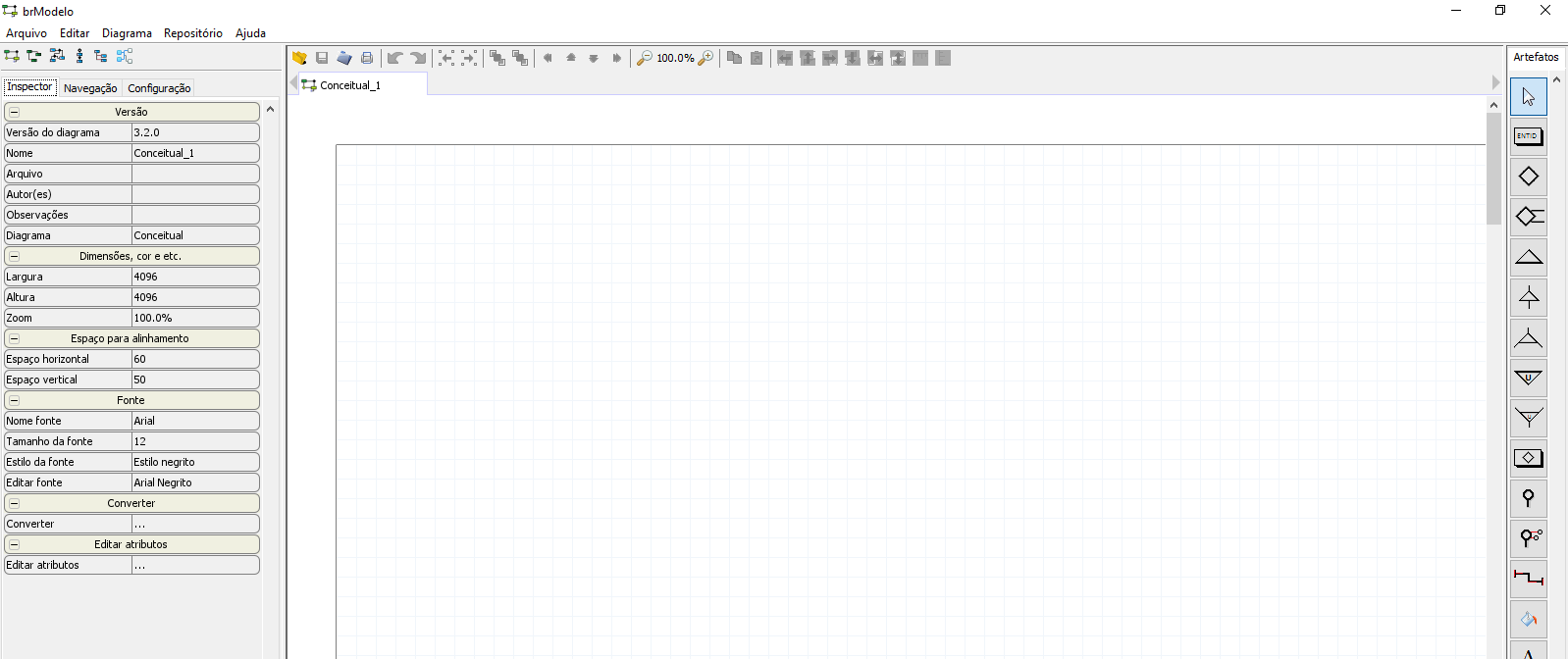


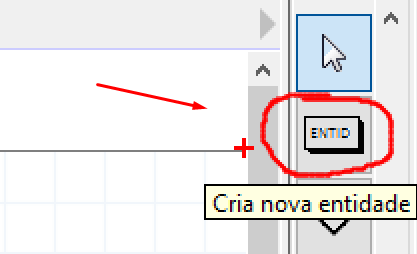
**Ficou bonito?**



**Engenheiro de Dados**

**Modelo Conceitual dos Dados**







**Insira atributos**

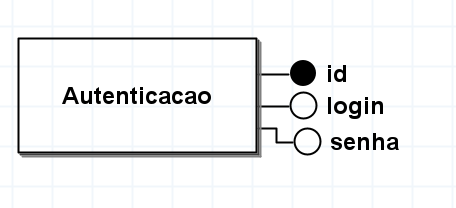
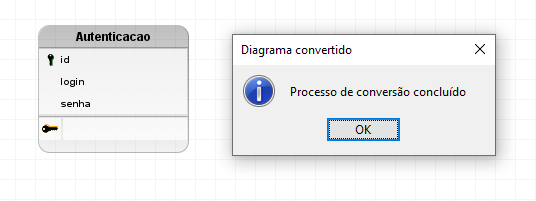


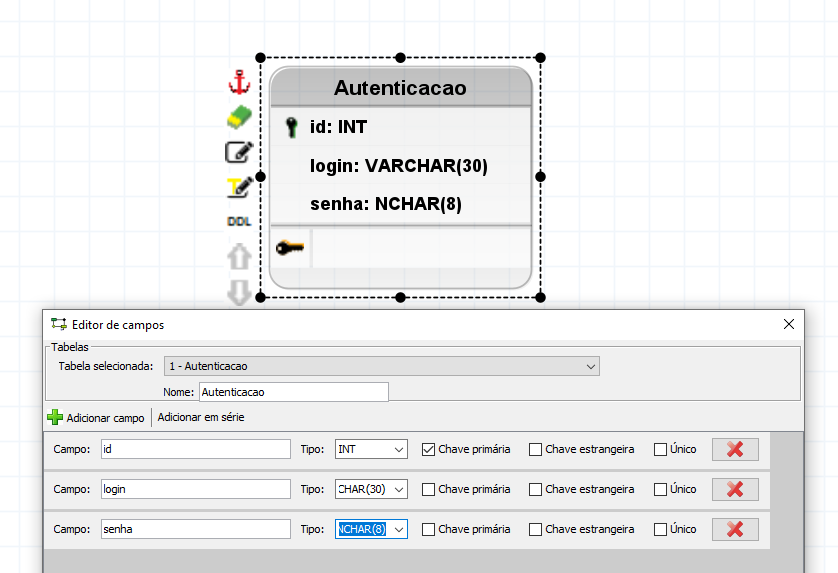
Figura - Entidade Autenticacao com três atributos.

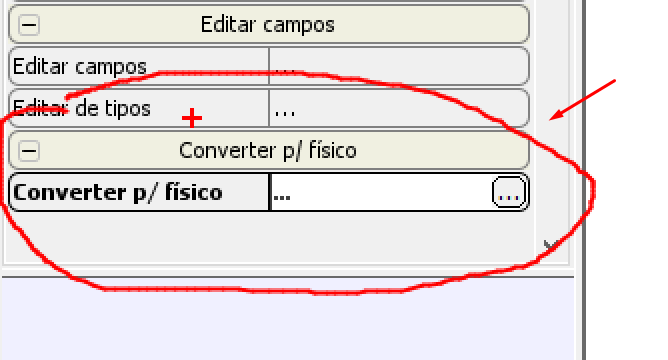
O primeiro atributo é chamado de atributo identificador (chave primária)

**Modelo Conceitual: ficar mais próximo do cliente e mais longe do técnico.**

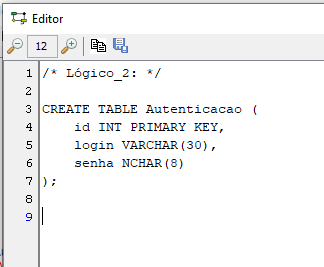
**Modelo Lógico:**



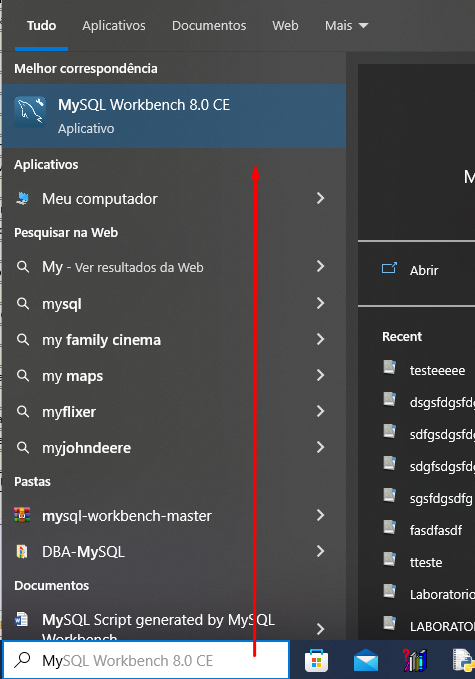




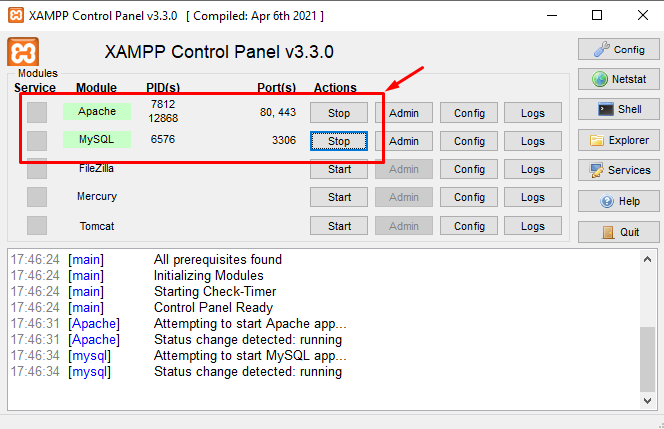
**Modelo Físico:**



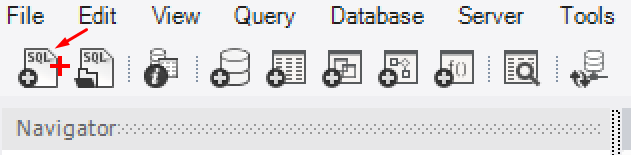
**Agora sim.... É um banco de verdade**

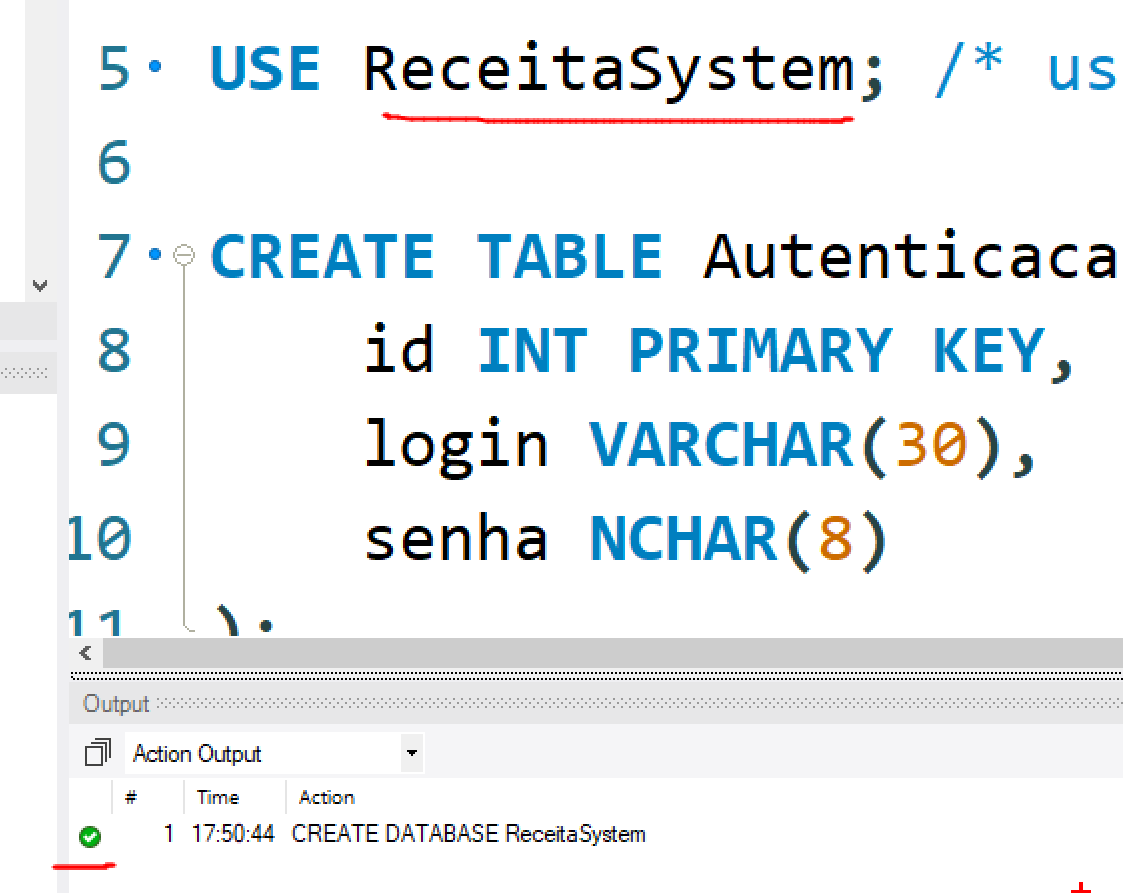


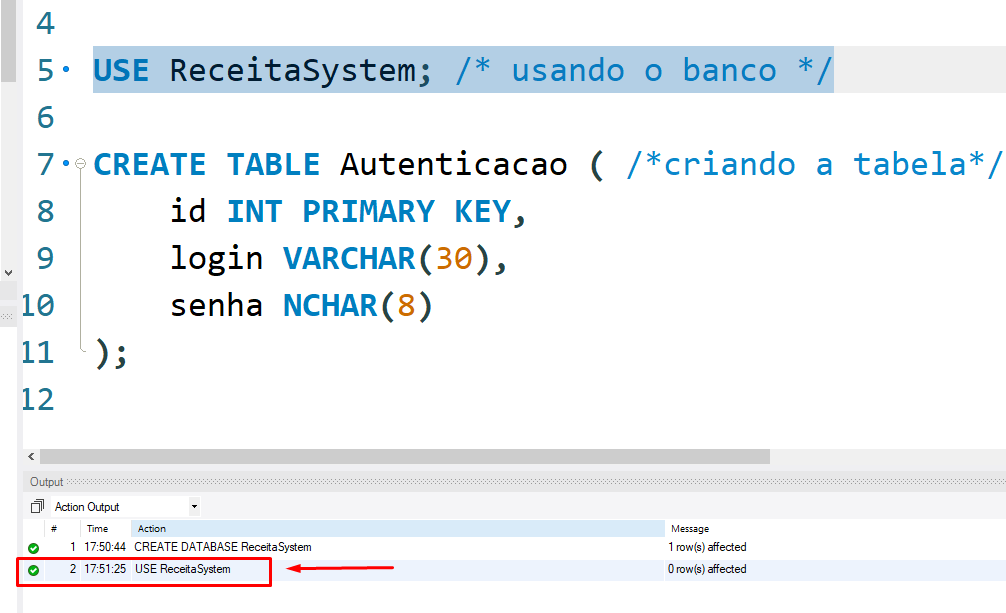
**Startar o Apache e o MySQL**

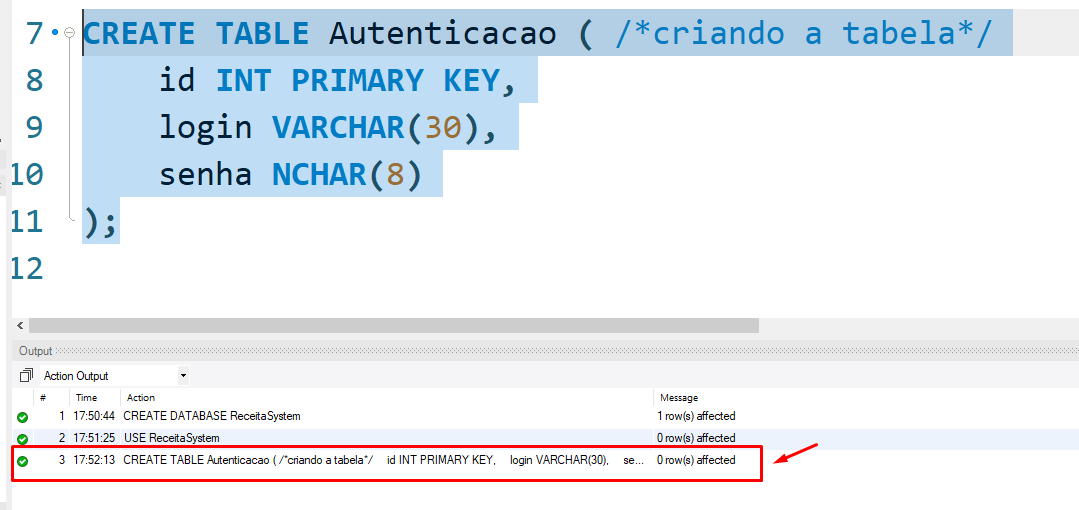


**Agora é pra valer**









**Código do Banco de Dados**

CREATE DATABASE ReceitaSystem; /\*criando o banco\*/

USE ReceitaSystem; /\* usando o banco \*/

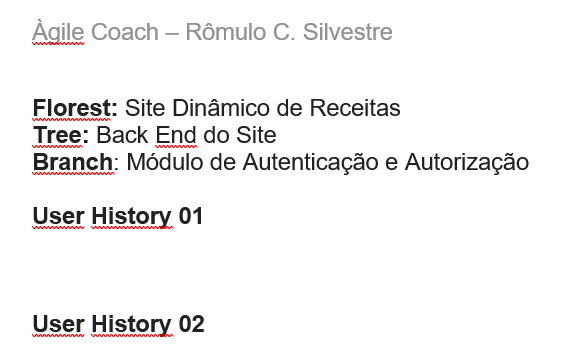
CREATE TABLE Autenticacao ( /\*criando a tabela\*/

id INT PRIMARY KEY,

login VARCHAR(30),

senha NCHAR(8)

);



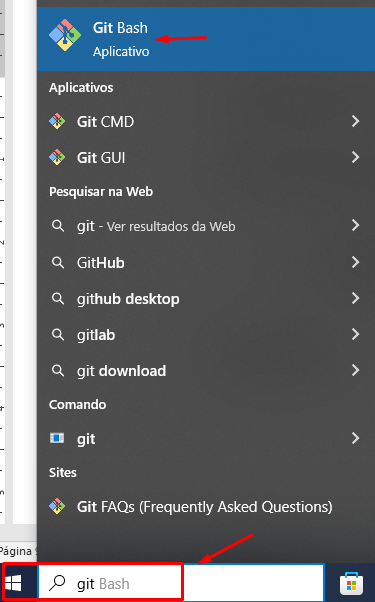
**Trabalhar em Equipe**

**Versionamento de códigos**

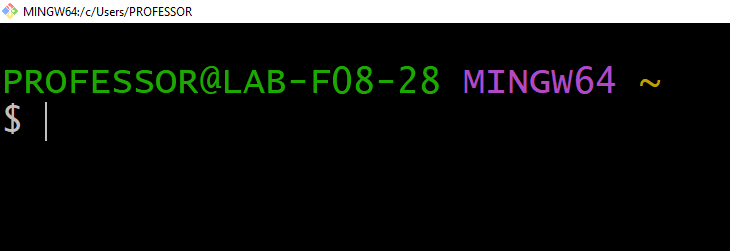
1. Sozinho
2. Com o time

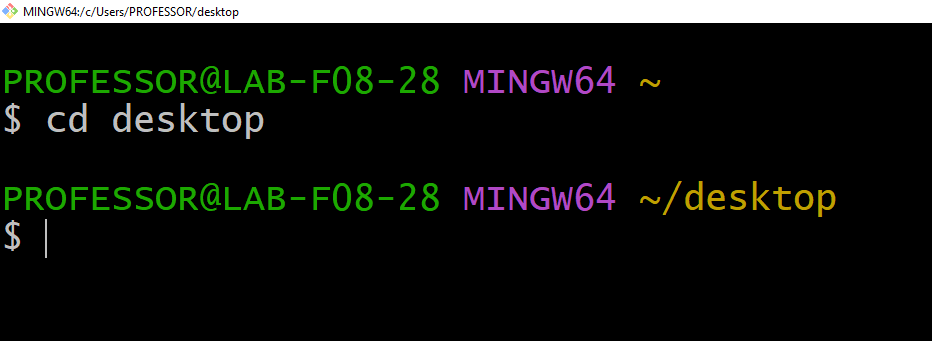
**Sozinho**

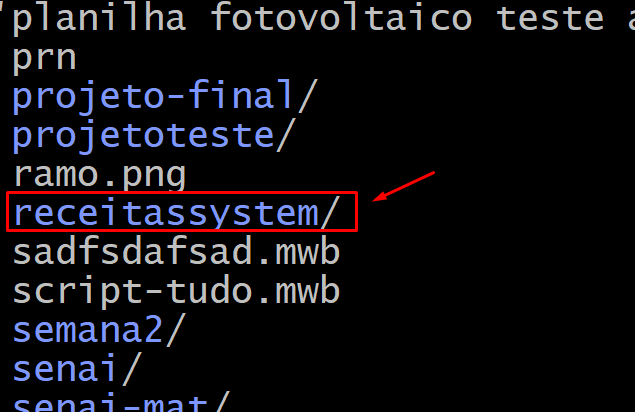
**Passo 1: Abra o gitbash**

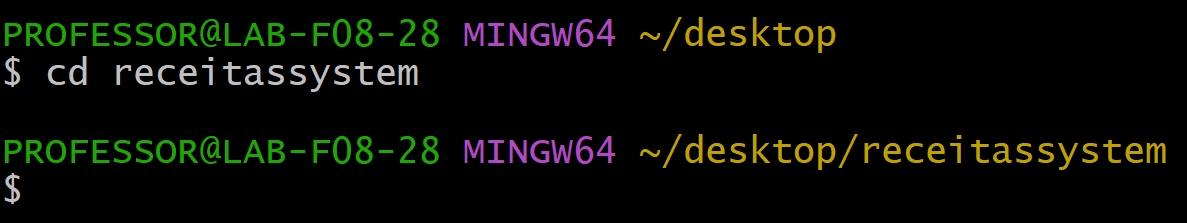


**Nossa missão entrar a pasta.**

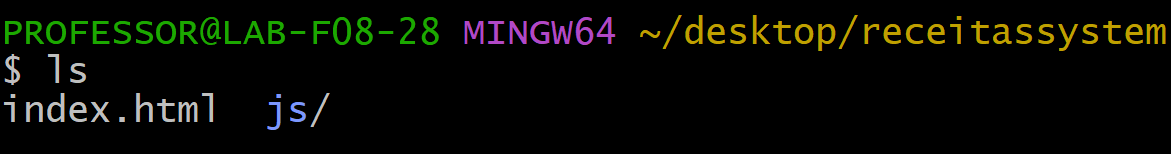




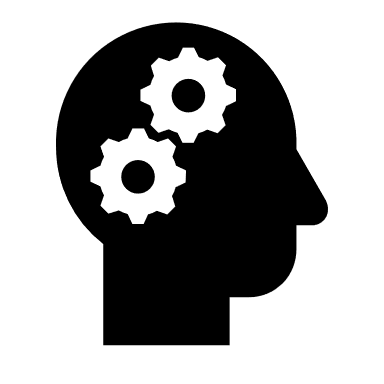




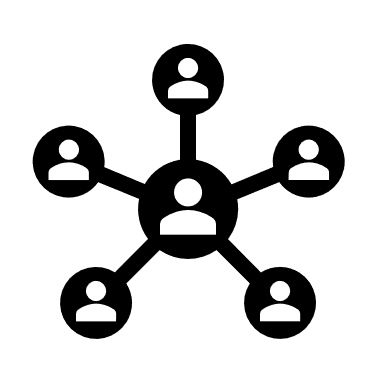
**Dê um ls para verificar os arquivos na pasta**



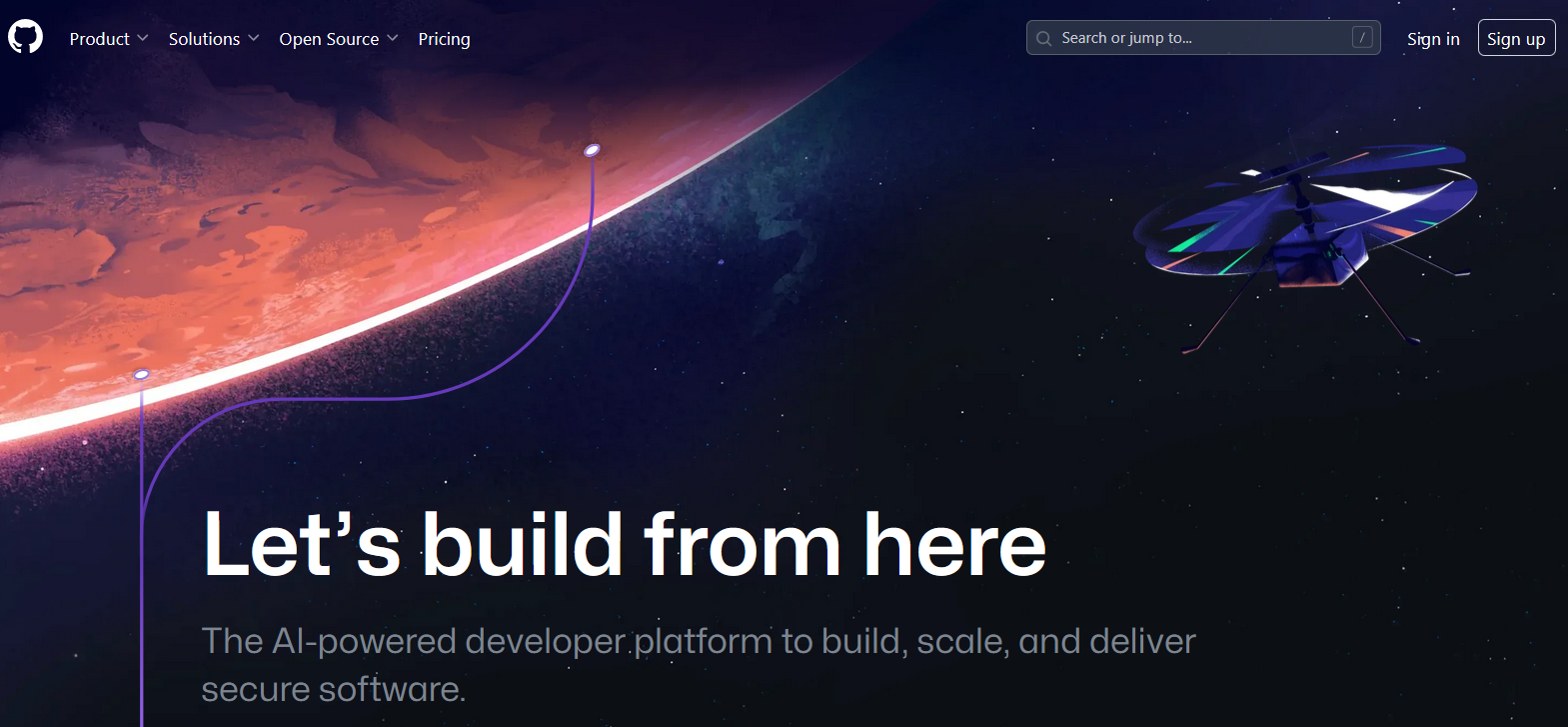
**Esse é o diretório principal do nosso projeto.**

**Pense sozinho**

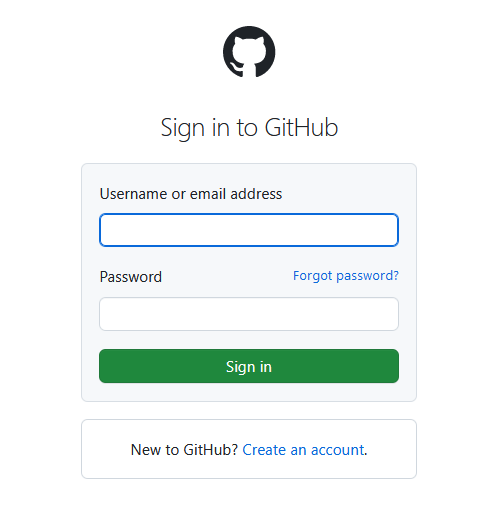
|  |  |
| --- | --- |
| **Comando Git** | **Para que serve?** |
| **Git init** | **Para iniciar** |
| **Git status** | **Verificar como se encontra a nossa árvore** |
| **Git branch** | **Para mostrar os ramos da nossa árvore** |
| **Git add .** | **Para deixar os arquivos gerenciáveis** |
| **Git commit** | **Para salvar os arquivos** |
| Figura - Repositório Inicializado    Figura - galho máster – arquivos não gerenciados (untracked)    Figura - gerenciar os arquivos    Figura - os arquivos estão gerenciados (verde)  **Não gerenciado**    **Gerenciado** | |

**Pense no time**

**Antes de tudo você tem que criar uma conta github.**

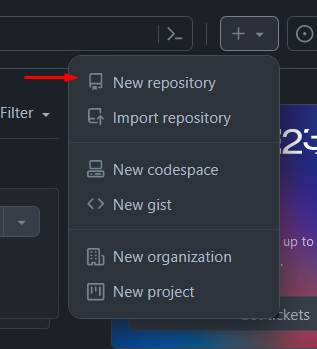


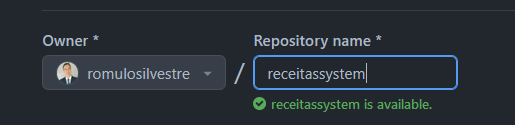
**Realize a autenticação:**



**Octocat**

[**https://myoctocat.com/**](https://myoctocat.com/)







|  |  |
| --- | --- |
| **Comando Git** | **Para que serve?** |
| **git remote add** | **Para adicionar um repositório remoto** |
| **git push** | **Empurar os dados para o servidor** |
|  | |

O que aprendemos hoje:

<ol>

<li>**Disciplined Agile®</**li>

<li>**Design System**</li>

<li>**Development Front End**</li>

<li>**Software Versioning**</li>

</ol>