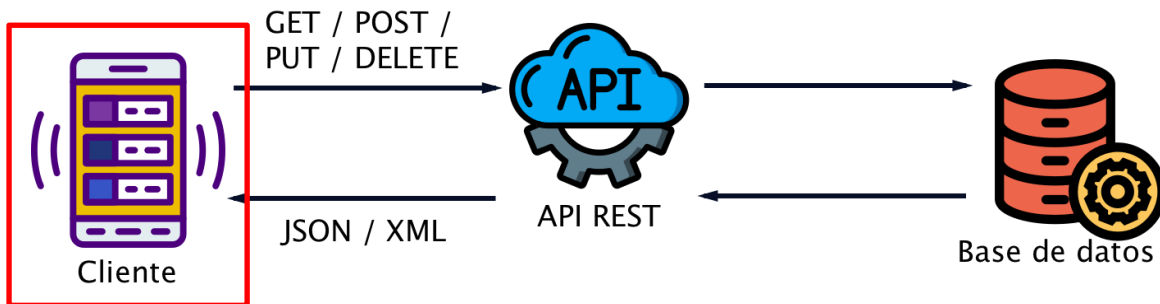


Desenvolvimento FrontEnd com Angular

Na aula de hoje, vamos começar a desenvolver uma aplicação FrontEnd com Angular para consumir a API de usuários desenvolvida na última aula.

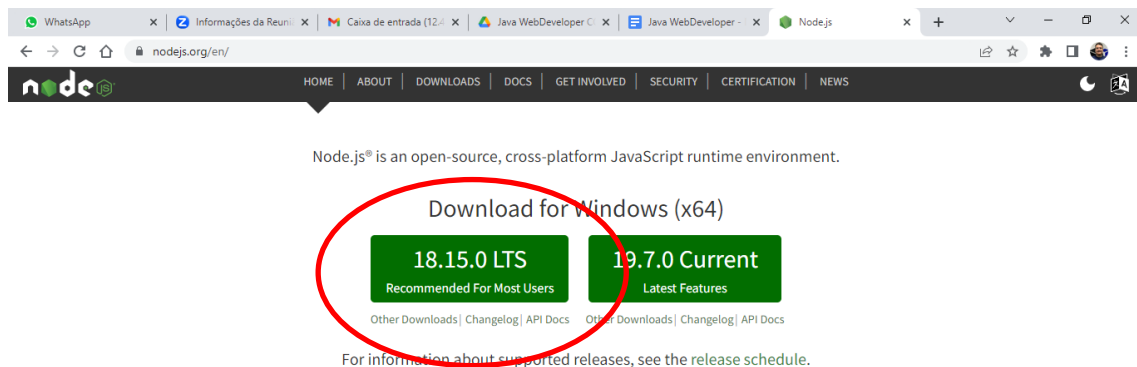


Setup do ambiente:

Precisamos configurar as ferramentas necessárias para desenvolvimento do projeto Angular (FrontEnd)

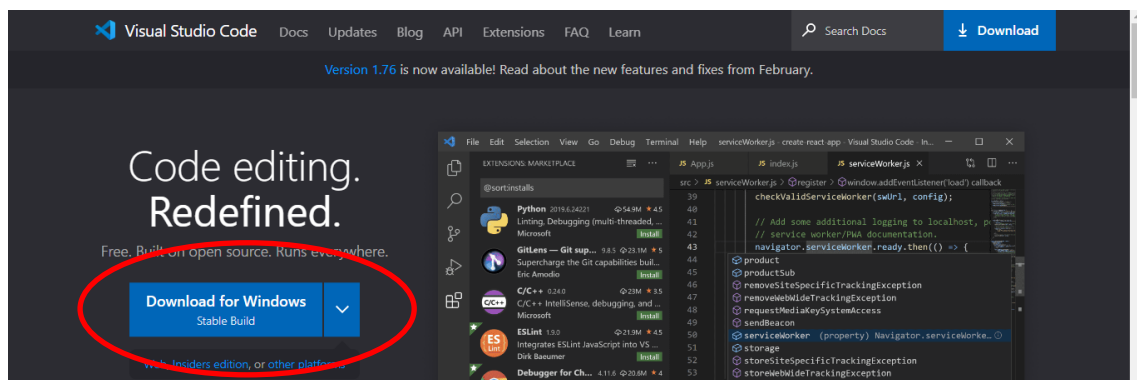
NodeJS

É a plataforma para desenvolvimento de aplicações baseadas em JavaScript. O framework do Angular é executado a partir de um servidor NodeJS (JavaScript)



Visual Studio Code

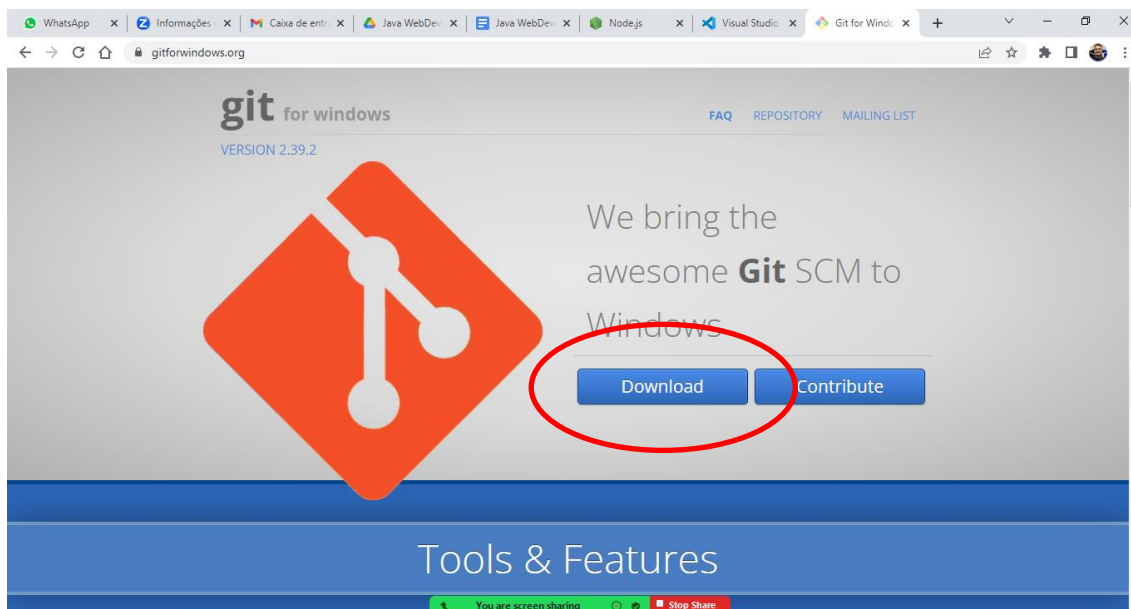
IDE para desenvolvimento dos projetos.



GITBASH

Terminal de linhas de comandos para o GITHUB.

<https://gitforwindows.org/>

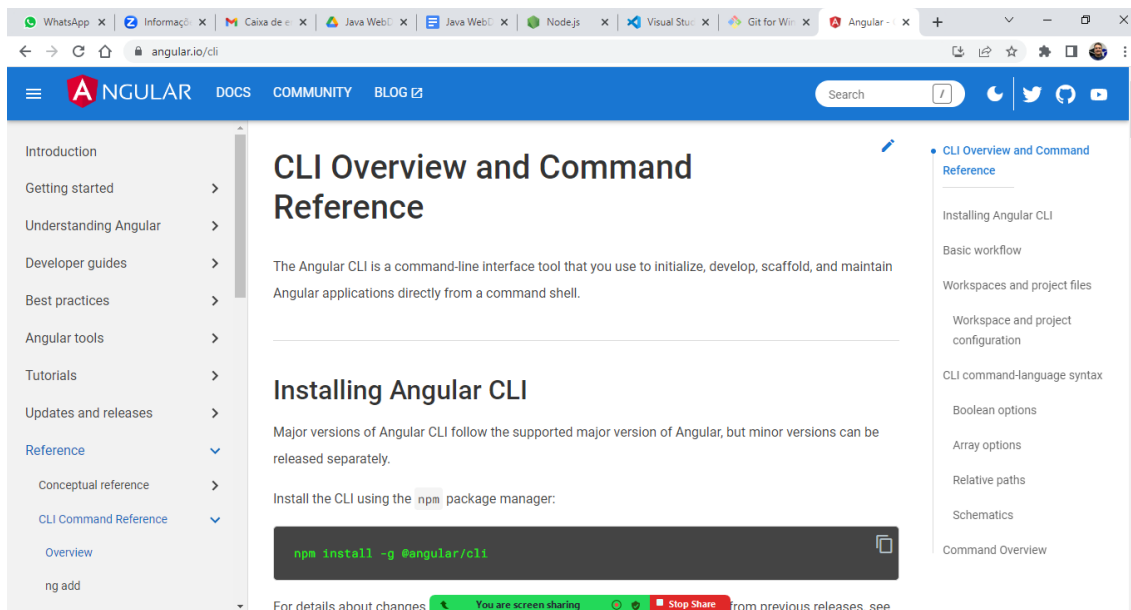


** Vamos criar uma pasta no raiz do c:\ para desenvolver os projetos:

@angular/cli

Conjunto de ferramentas para desenvolvimento de projetos baseados em Angular. Toolkit para criação e desenvolvimento com Angular.

<https://angular.io/cli>



Verificando se o NODEJS está instalado:

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ node -v  
v18.14.2
```

Instalando o @angular/cli

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ npm install -g @angular/cli
```

Verificando a versão do Angular:

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ ng version
```

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ ng version  
  
Angular CLI  
  
Angular CLI: 15.2.2  
Node: 18.14.2  
Package Manager: npm 8.13.2  
OS: win32 x64  
  
Angular:  
...  


| Package                    | Version             |
|----------------------------|---------------------|
| @angular-devkit/architect  | 0.1502.2 (cli-only) |
| @angular-devkit/core       | 15.2.2 (cli-only)   |
| @angular-devkit/schematics | 15.2.2 (cli-only)   |
| @schematics/angular        | 15.2.2 (cli-only)   |


```

Criando o projeto:

Desenvolvendo o sistema:

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ ng new projeto_angular
```

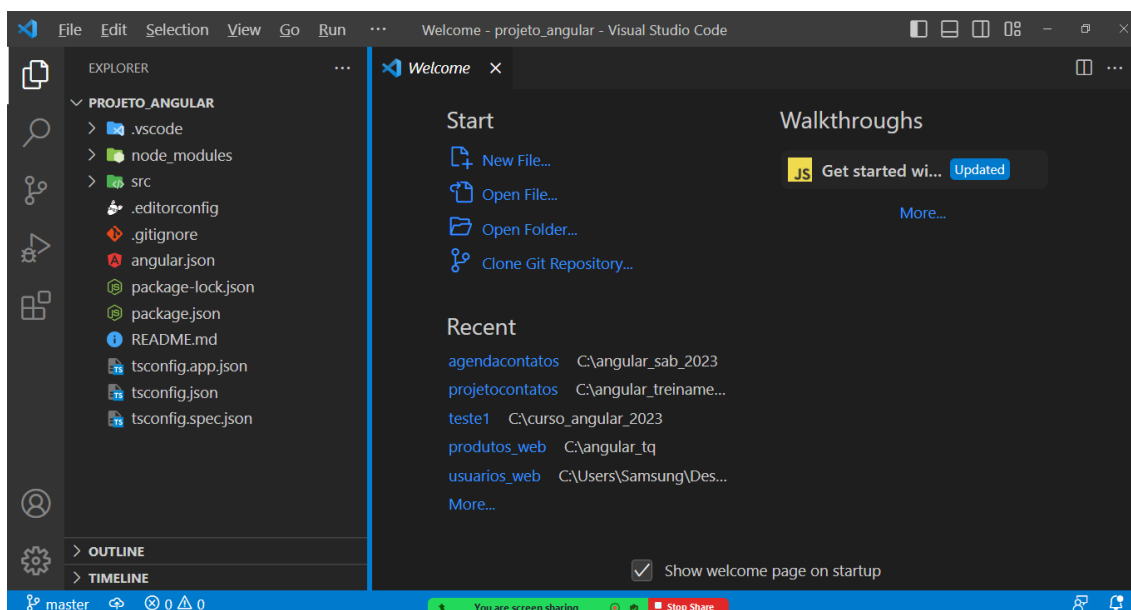
```
MINGW64: c:/dev_angular_sqs  
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ ng new projeto_angular  
CREATE projeto_angular/angular.json (2745 bytes)  
CREATE projeto_angular/package.json (1046 bytes)  
CREATE projeto_angular/README.md (1068 bytes)  
CREATE projeto_angular/tsconfig.json (901 bytes)  
CREATE projeto_angular/.editorconfig (274 bytes)  
CREATE projeto_angular/.gitignore (548 bytes)  
CREATE projeto_angular/tsconfig.app.json (263 bytes)  
CREATE projeto_angular/tsconfig.spec.json (273 bytes)  
CREATE projeto_angular/.vscode/extensions.json (130 bytes)  
CREATE projeto_angular/.vscode/launch.json (474 bytes)  
CREATE projeto_angular/.vscode/tasks.json (938 bytes)  
CREATE projeto_angular/src/favicon.ico (948 bytes)  
CREATE projeto_angular/src/index.html (300 bytes)  
CREATE projeto_angular/src/main.ts (214 bytes)  
CREATE projeto_angular/src/styles.css (80 bytes)  
CREATE projeto_angular/src/assets/.gitkeep (0 bytes)  
CREATE projeto_angular/src/app/app.module.ts (314 bytes)  
CREATE projeto_angular/src/app/app.component.html (23083 bytes)  
CREATE projeto_angular/src/app/app.component.spec.ts (983 bytes)  
CREATE projeto_angular/src/app/app.component.ts (219 bytes)  
CREATE projeto_angular/src/app/app.component.css (0 bytes)  
- Installing packages (npm)...
```

Acessando a pasta do projeto:

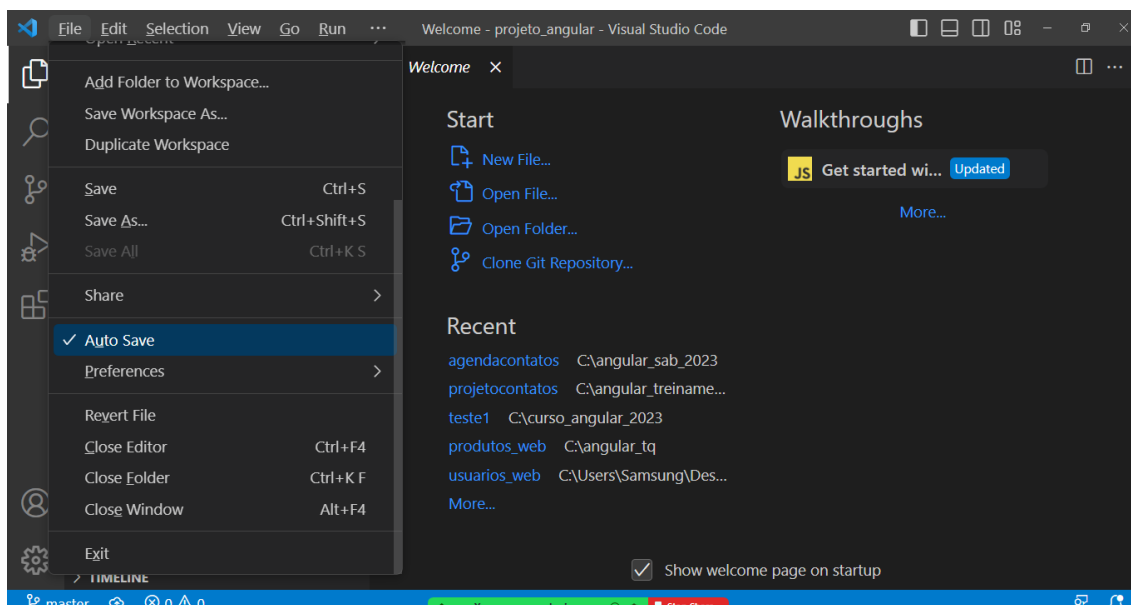
```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev_angular_sqs  
$ cd projeto_angular
```

Abrindo o projeto no VisualStudio Code:

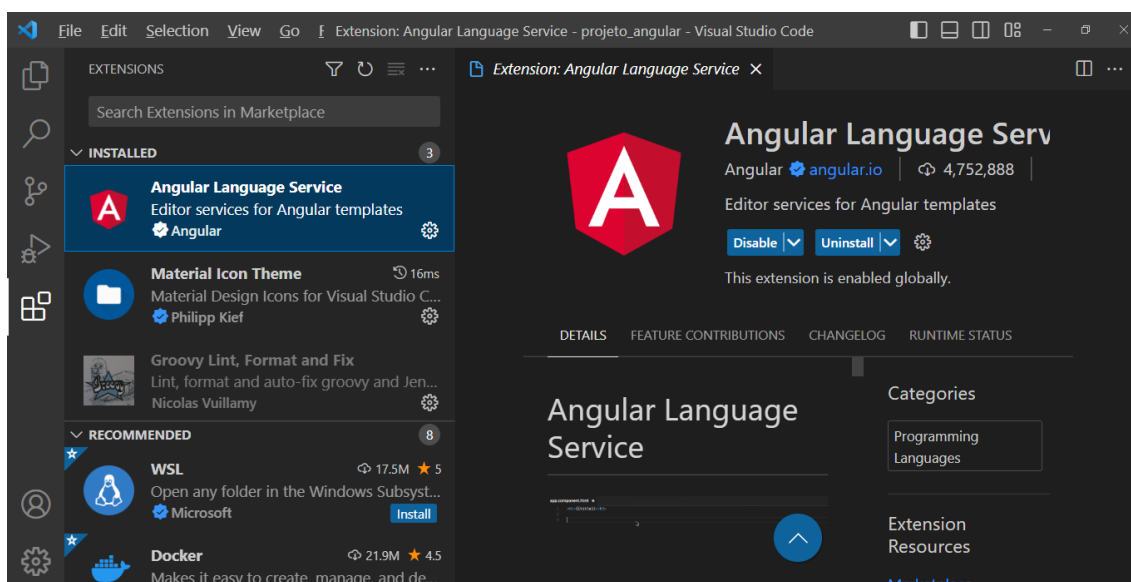
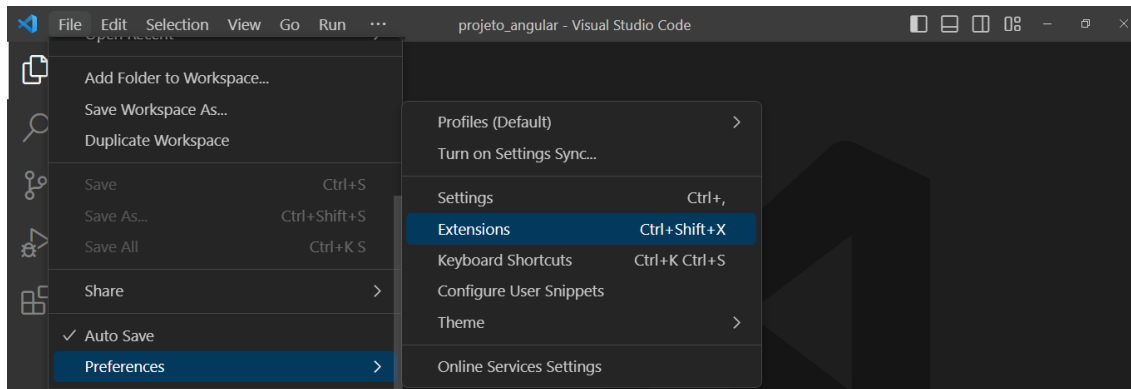
```
Samsung@DESKTOP-P9F6D9F MINGW64  
/c/dev_angular_sqs/projeto_angular (master)  
$ code .
```



Configurando o Visual Studio Code para trabalhar no modo AutoSave:



Instalando extensões no Visual Studio Code para trabalharmos com o framework Angular:



Executando o projeto:

Os projetos criados em Angular já possuem um servidor web “embarcado” através do qual podemos executar as aplicações.

```
Samsung@DESKTOP-P9F6D9F MINGW64
/c/dev/angular_sqs/projeto_angular (master)
$ ng s
```

```
Samsung@DESKTOP-P9F6D9F MINGW64 /c/dev/angular_sqs/projeto_angular (master)
$ ng s
- Generating browser application bundles (phase: setup)...
✓ Browser application bundle generation complete.

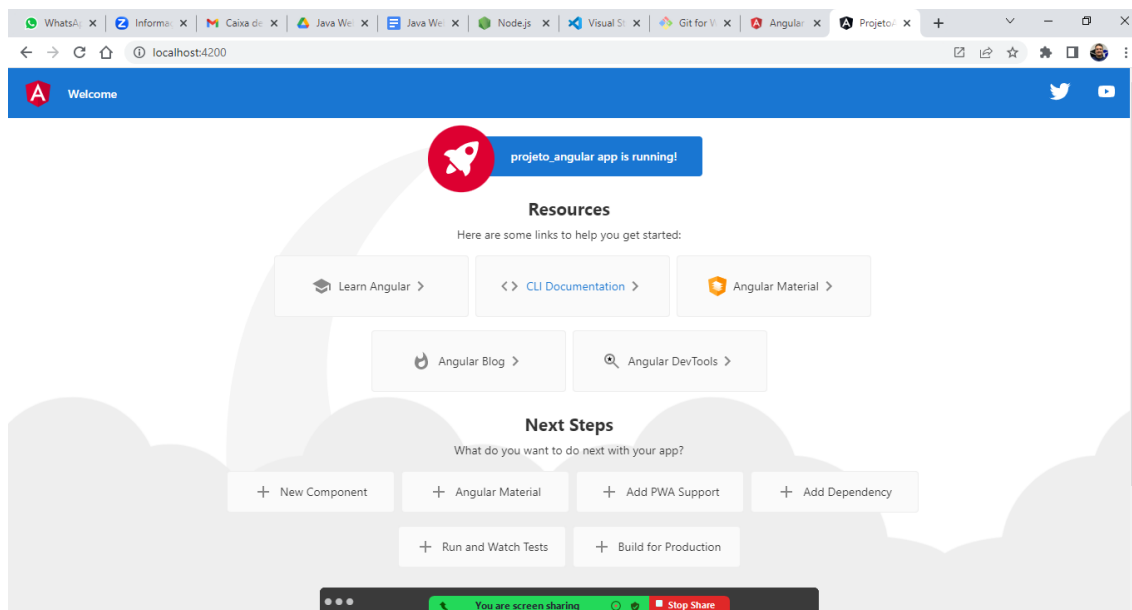
Initial Chunk Files | Names          | Raw Size
---
vendor.js           | vendor         | 1.71 MB
polyfills.js        | polyfills      | 314.28 kB
styles.css, styles.js | styles        | 209.41 kB
main.js             | main           | 46.01 kB
runtime.js          | runtime        | 6.53 kB

| Initial Total | 2.27 MB

Build at: 2023-03-13T21:59:26.832Z - Hash: 0d9bee97e2d6c88a - Time: 16525ms
** Angular Live Development Server is listening on localhost:4200, open your browser on http://localhost:4200/ **

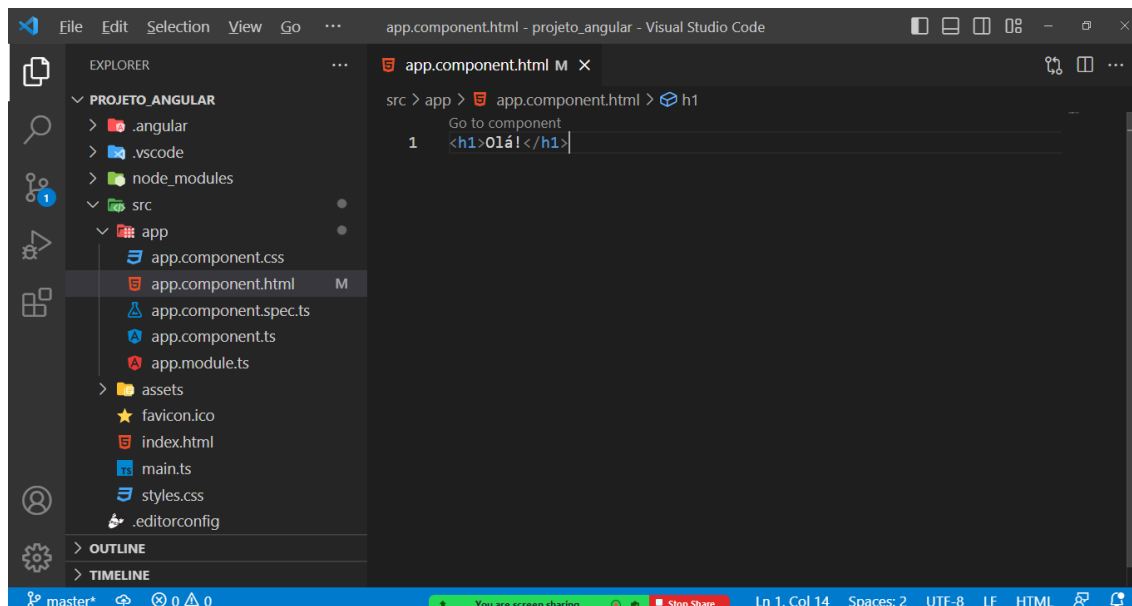
✓ Compiled successfully.
```

<http://localhost:4200/>



Esta página está desenhada no arquivo:

[/app.component.html](#)



Neste projeto, a primeira página que será exibida no navegador é um formulário para login do usuário.

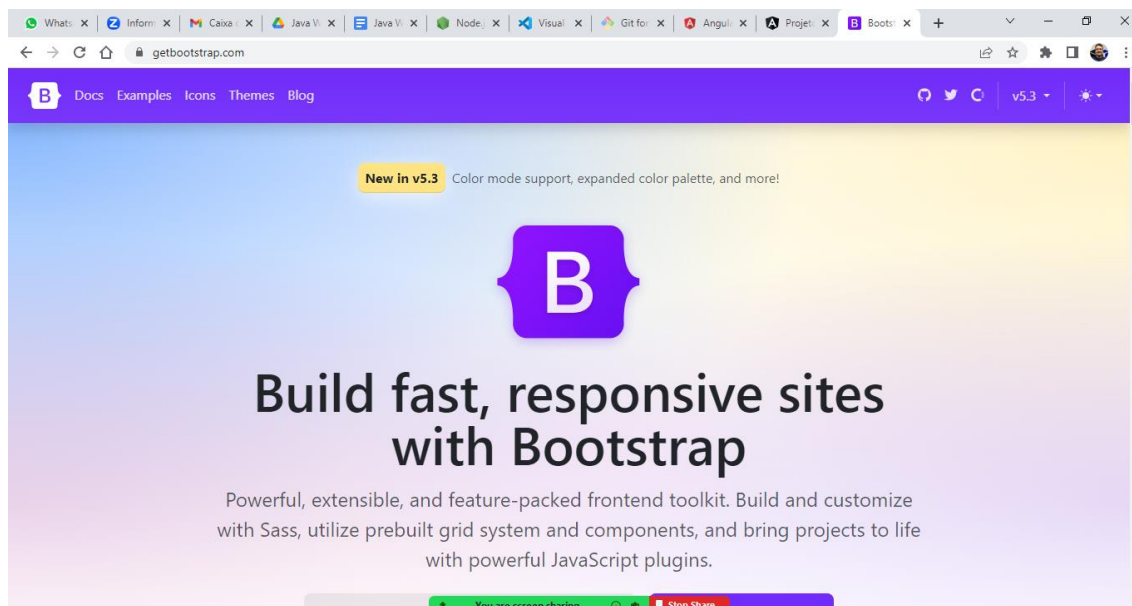
Para isso, precisamos instalar o bootstrap no projeto.

Bootstrap

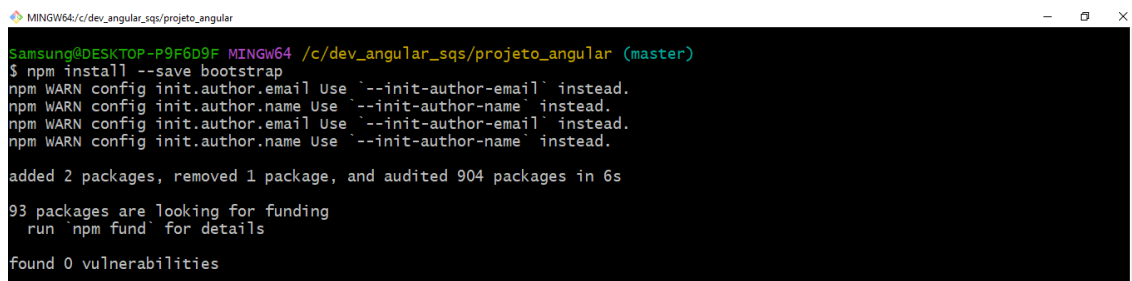
Toolkit para desenvolvimento CSS responsivo.

<https://getbootstrap.com/>

<https://getbootstrap.com/>

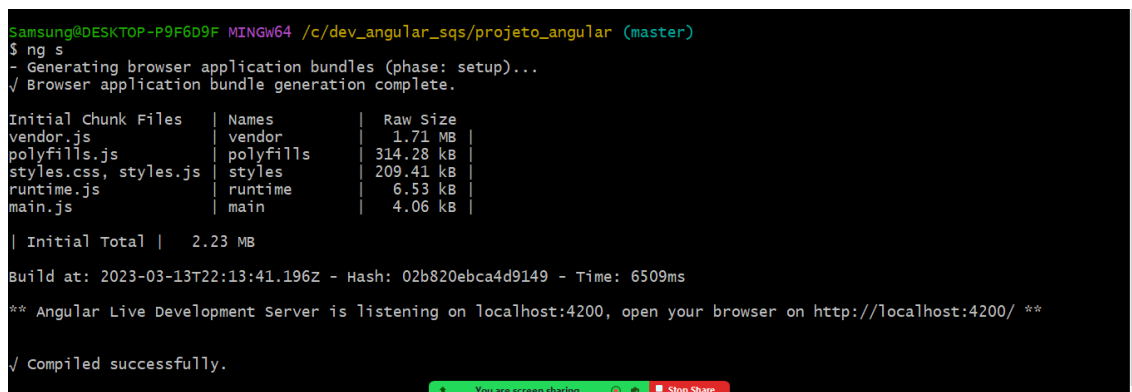


```
Samsung@DESKTOP-P9F6D9F MINGW64  
/c/dev_angular_sqs/projeto_angular (master)  
$ npm install --save bootstrap
```

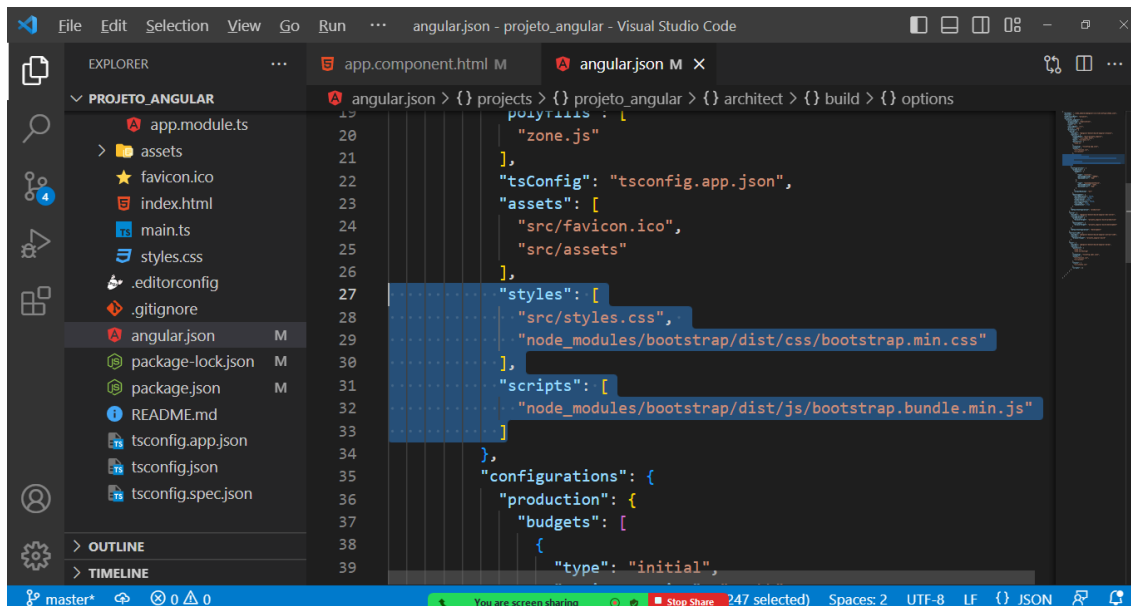


Após a instalação do bootstrap, execute o projeto novamente:

```
Samsung@DESKTOP-P9F6D9F MINGW64  
/c/dev_angular_sqs/projeto_angular (master)  
$ ng s
```



Configurando o uso do **CSS** e **JS** default do bootstrap no projeto:
/angular.json



The screenshot shows the Visual Studio Code editor with the 'angular.json' file open. The file is part of a project named 'projeto_angular'. The configuration includes a 'styles' array with 'src/styles.css' and 'node_modules/bootstrap/dist/css/bootstrap.min.css', and a 'scripts' array with 'node_modules/bootstrap/dist/js/bootstrap.bundle.min.js'. The 'production' configuration is also visible, showing a 'type' of 'initial'.

```
"styles": [
  "src/styles.css",
  "node_modules/bootstrap/dist/css/bootstrap.min.css"
],
"scripts": [
  "node_modules/bootstrap/dist/js/bootstrap.bundle.min.js"
]
```

** Após a configuração acima, reinicie o servidor do projeto.

Desenhando uma página inicial para login do usuário:
/app.component.html

```
<div class="container">
  <div class="row justify-content-center mt-5">
    <div class="col-md-4">
      <div class="card">
        <div class="card-body">

          <div class="text-center">
            <h5 class="card-title text-center">
              Autenticação de Usuários
            </h5>
            <p>Entre com as suas credenciais de acesso</p>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```



```
</div>

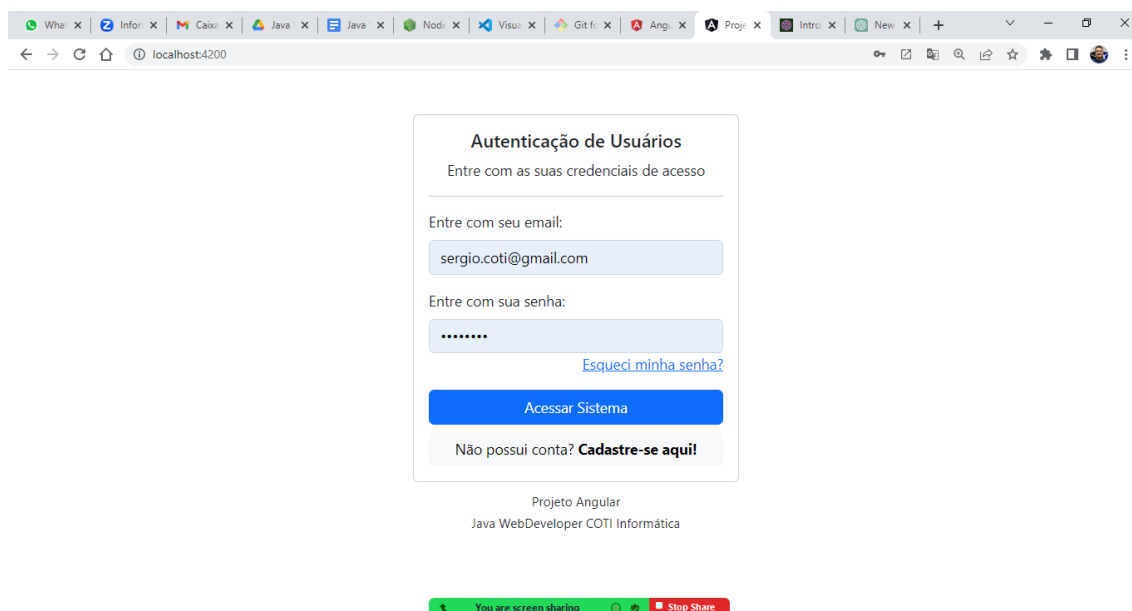
<hr/>

<form>
  <div class="mb-3">
    <label for="inputEmail" class="form-label">
      Entre com seu email:</label>
    <input
      type="email"
      class="form-control"
      id="inputEmail"
      aria-describedby="emailHelp"
      placeholder="Digite o seu e-mail"
      required
    />
  </div>
  <div class="mb-3">
    <label for="inputPassword" class="form-label">
      Entre com sua senha:</label>
    <input
      type="password"
      class="form-control"
      id="inputPassword"
      placeholder="Digite a sua senha"
      required
    />
  <div class="text-end mb-2">
    <a href="#">
      Esqueci minha senha?
    </a>
  </div>
</div>
<div class="d-grid gap-2">
  <button class="btn btn-primary" type="submit">
    Acessar Sistema
  </button>
</div>
<div class="d-grid gap-2 mt-2">
  <a href="#" class="btn btn-light">
    Não possui conta? <strong>Cadastre-se aqui!</strong>
  </a>
</div>
</form>
</div>
</div>
```

```
<div class="mt-2 text-center">
  <small>
    Projeto Angular<br/>Java WebDeveloper COTI Informática
  </small>
</div>

</div>
</div>
</div>
```

Resultado:



Autenticação de Usuários

Entre com as suas credenciais de acesso

Entre com seu email:

sergio.coti@gmail.com

Entre com sua senha:

.....

[Esqueci minha senha?](#)

Acessar Sistema

Não possui conta? **Cadastre-se aqui!**

Projeto Angular
Java WebDeveloper COTI Informática

Desenhando a página completa:

/app.component.html

```
<div class="container">
  <div class="row justify-content-center mt-5">
    <div class="col-md-4">
      <div class="card">
        <div class="card-body">

          <div class="text-center">
            <h5 class="card-title text-center">
              Autenticação de Usuários</h5>
            <p>Entre com as suas credenciais de acesso</p>
          </div>

        </div>
      </div>
    </div>
  </div>
</div>
```

```
<form>
  <div class="mb-3">
    <label for="inputEmail" class="form-label">
      Entre com seu email:</label>
    <input
      type="email"
      class="form-control"
      id="inputEmail"
      aria-describedby="emailHelp"
      placeholder="Digite o seu e-mail"
      required
    />
  </div>
  <div class="mb-3">
    <label for="inputPassword" class="form-label">
      Entre com sua senha:</label>
    <input
      type="password"
      class="form-control"
      id="inputPassword"
      placeholder="Digite a sua senha"
      required
    />
  </div>
  <div class="text-end mb-2">
    <a href="#" data-bs-toggle="modal" data-bs-target="#janelaRecuperarSenha">
      Esqueci minha senha?
    </a>
  </div>
  </div>
  <div class="d-grid gap-2">
    <button class="btn btn-primary" type="submit">
      Acessar Sistema
    </button>
  </div>
  <div class="d-grid gap-2 mt-2">
    <a href="#" class="btn btn-light" data-bs-toggle="modal" data-bs-target="#janelaCadastro">
      Não possui conta? <strong>Cadastre-se aqui!</strong>
    </a>
  </div>
</form>
</div>
</div>

<div class="mt-2 text-center">
  <small>
```

```
Projeto Angular<br/>Java WebDeveloper COTI Informática
</small>
</div>

</div>
</div>
</div>

<!-- Modal para recuperação de senha do usuário -->
<div class="modal fade" id="janelaRecuperarSenha" tabindex="-1" aria-
labelledby="exampleModallabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5" id="exampleModallabel">
          Esqueci minha senha</h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal"
          aria-label="Close"></button>
      </div>
      <div class="modal-body">

        <form>

          <div class="mb-3">
            <label for="inputEmail" class="form-label">
              Entre com seu email:</label>
            <input
              type="email"
              class="form-control"
              id="inputEmail"
              aria-describedby="emailHelp"
              placeholder="Digite o seu e-mail"
              required
            />
          </div>

          <div class="d-grid gap-2">
            <button class="btn btn-primary" type="submit">
              Confirmar
            </button>
          </div>

        </form>

      </div>
    </div>
  </div>
</div>
```

```
</div>

<!-- Modal para cadastro do usuário -->
<div class="modal fade" id="janelaCadastro" tabindex="-1" aria-
labelledby="exampleModallabel" aria-hidden="true">
  <div class="modal-dialog modal-dialog-centered">
    <div class="modal-content">
      <div class="modal-header">
        <h1 class="modal-title fs-5" id="exampleModallabel">Criar Conta
        de Usuário</h1>
        <button type="button" class="btn-close" data-bs-dismiss="modal"
        aria-label="Close"></button>
      </div>
      <div class="modal-body">

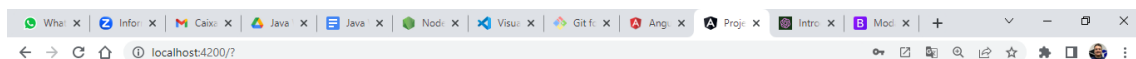
        <form>

          <div class="mb-3">
            <label for="inputNome" class="form-label">Entre com seu
            nome:</label>
            <input
              type="text"
              class="form-control"
              id="inputNome"
              placeholder="Digite o seu nome"
              required
            />
          </div>

          <div class="mb-3">
            <label for="inputEmail" class="form-label">Entre com seu
            email:</label>
            <input
              type="email"
              class="form-control"
              id="inputEmail"
              aria-describedby="emailHelp"
              placeholder="Digite o seu e-mail"
              required
            />
          </div>

          <div class="mb-3">
            <label for="inputSenha" class="form-label">Entre com sua
            senha:</label>
            <input
              type="password"
```

```
        class="form-control"
        id="inputSenha"
        placeholder="Digite a sua senha"
        required
    />
</div>
<div class="mb-3">
    <label for="inputSenhaConfirmacao"
        class="form-label">Confirme a sua senha:</label>
    <input
        type="password"
        class="form-control"
        id="inputSenhaConfirmacao"
        placeholder="Confirme a sua senha"
        required
    />
</div>
<div class="d-grid gap-2">
    <button class="btn btn-primary" type="submit">
        Realizar Cadastro
    </button>
</div>
</form>
</div>
</div>
</div>
</div>
```



Autenticação de Usuários

Entre com as suas credenciais de acesso

Entre com seu email:

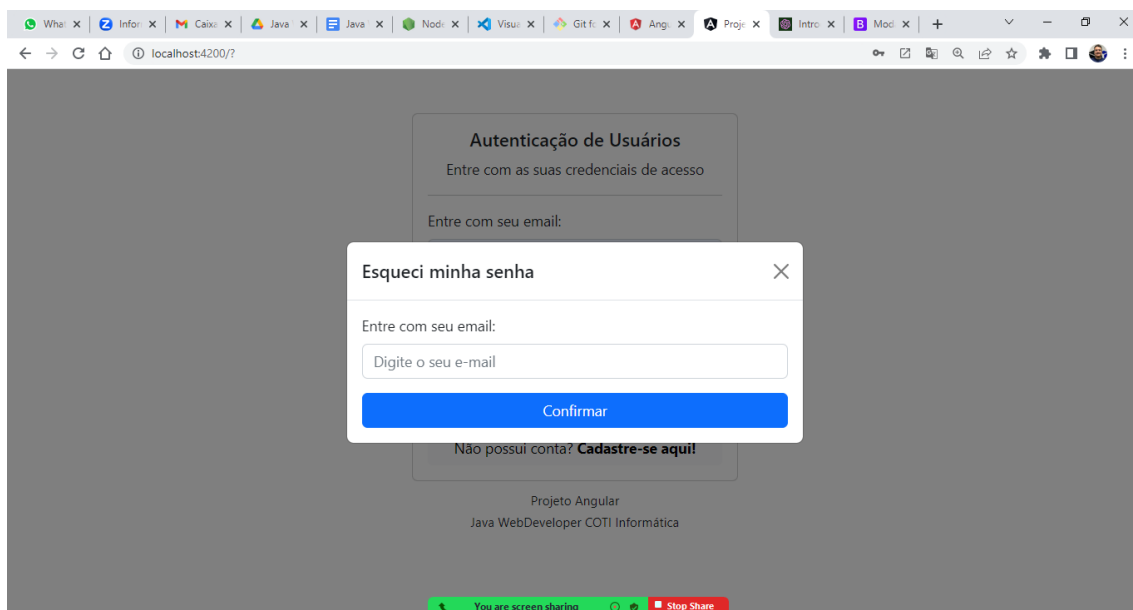
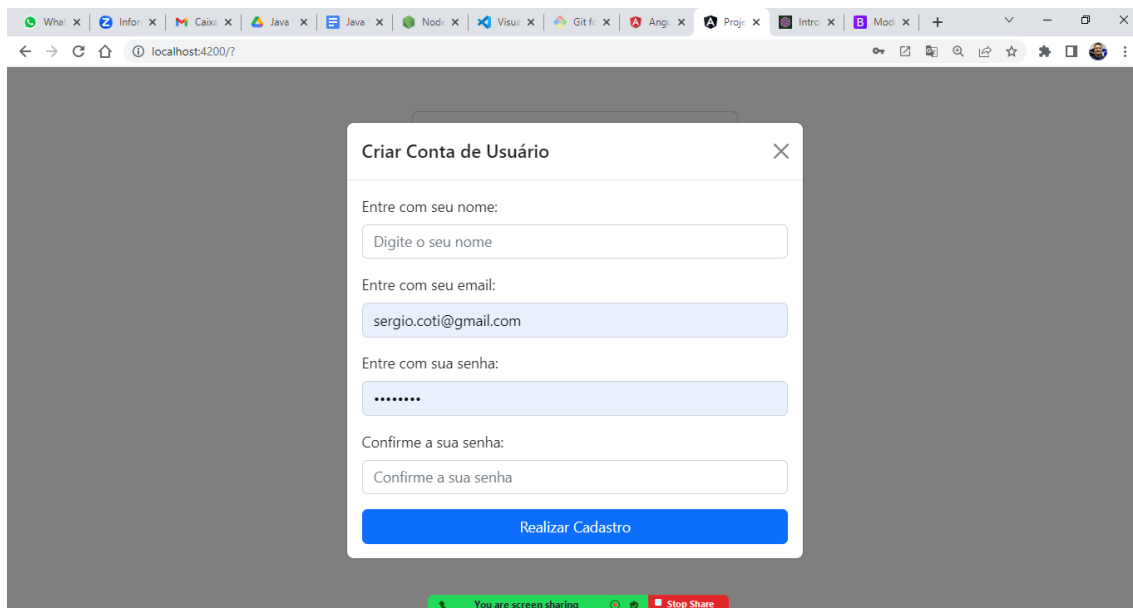
Entre com sua senha:

[Esqueci minha senha?](#)

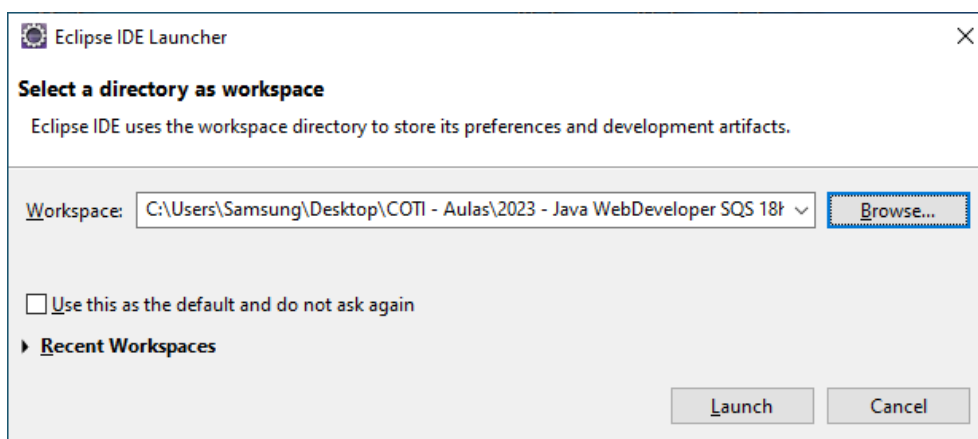
Acessar Sistema

Não possui conta? **Cadastre-se aqui!**

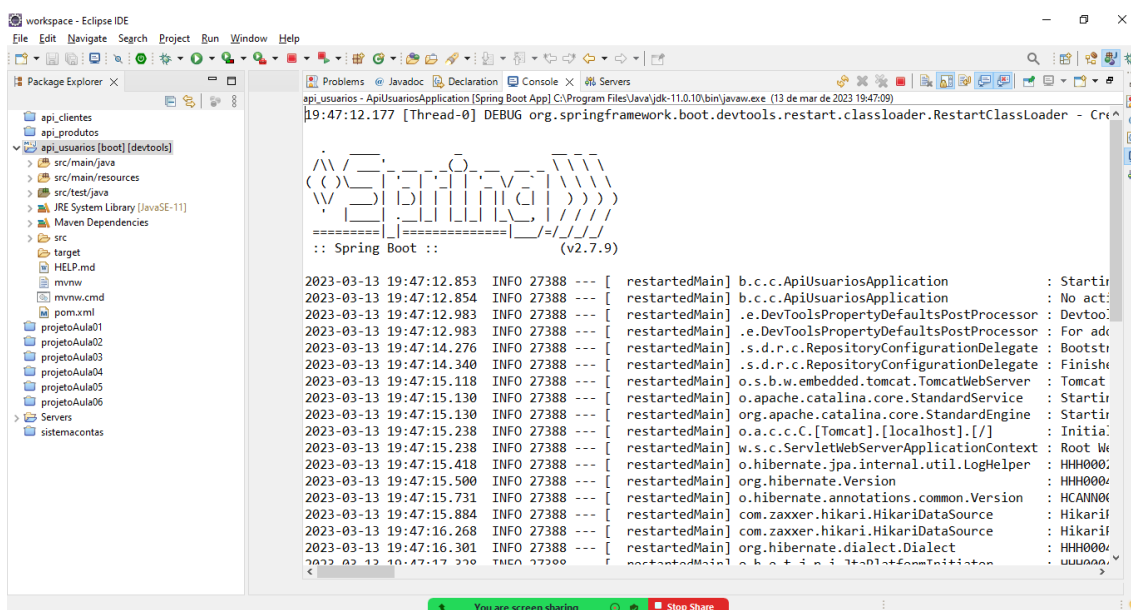
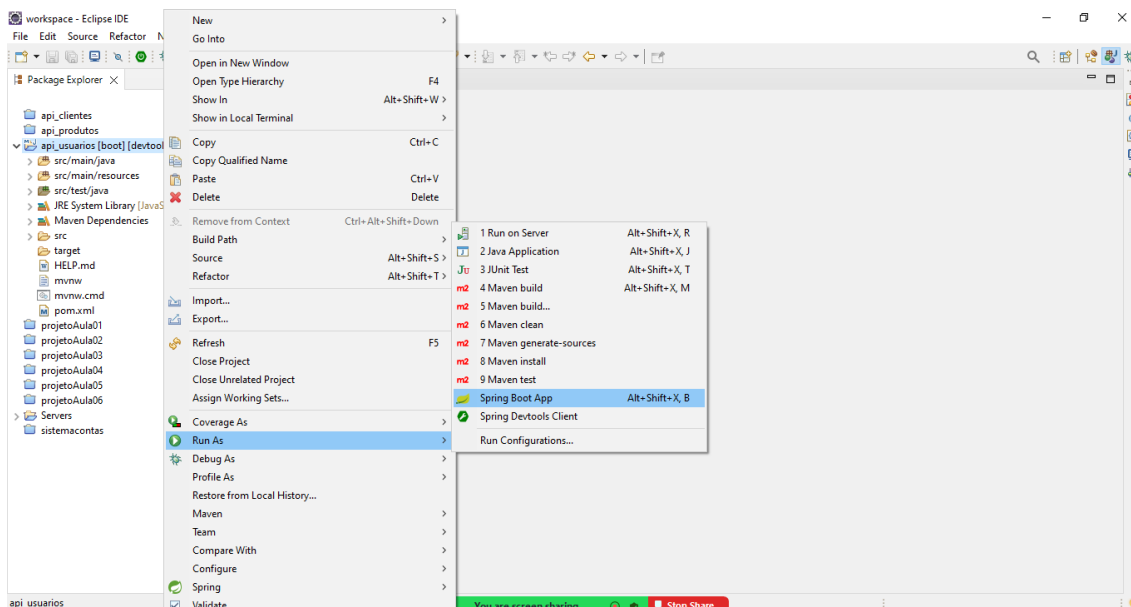
Projeto Angular
Java WebDeveloper COTI Informática



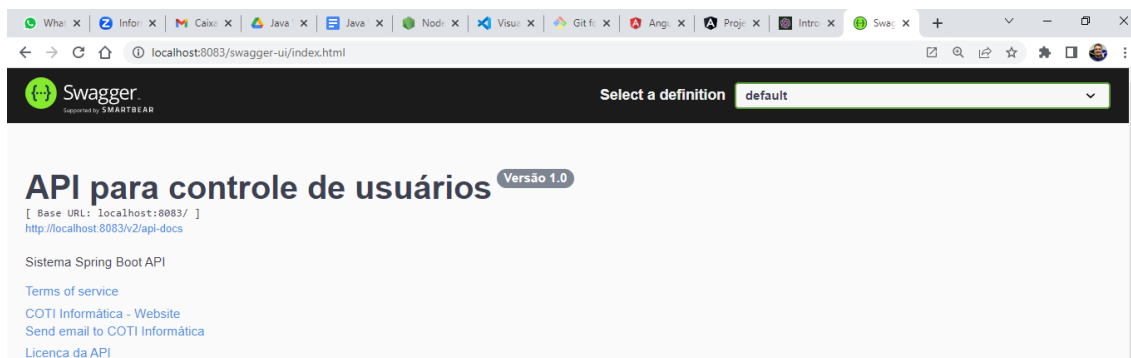
Abrindo o eclipse (projeto Java):



Executando a API:

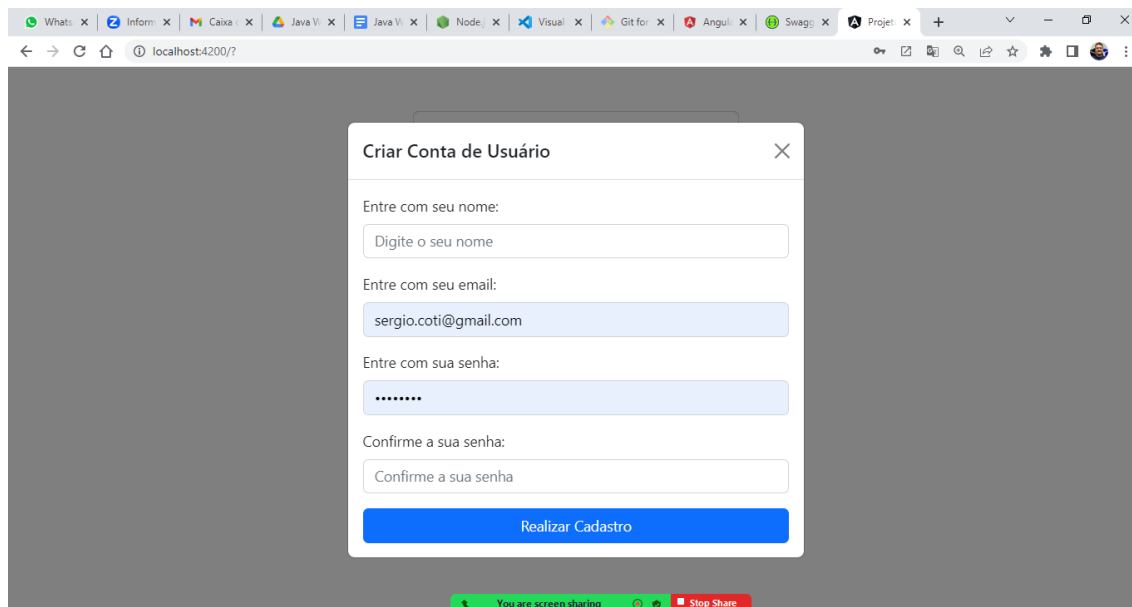


<http://localhost:8083/swagger-ui/index.html>



A primeira integração que vamos construir será do formulário para **cadastro de conta de usuário**.

FRONTEND:

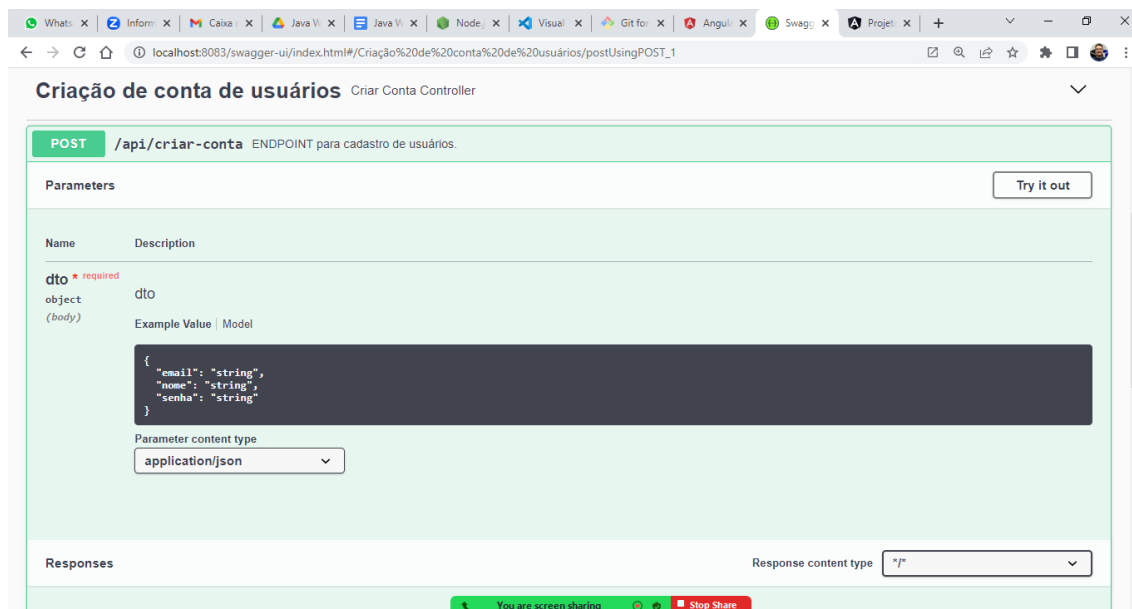


The screenshot shows a web browser window with a modal form titled 'Criar Conta de Usuário'. The form contains the following fields and elements:

- Field: 'Entre com seu nome:' with a placeholder 'Digite o seu nome'.
- Field: 'Entre com seu email:' with the value 'sergio.coti@gmail.com'.
- Field: 'Entre com sua senha:' with a masked password '.....'.
- Field: 'Confirme a sua senha:' with a placeholder 'Confirme a sua senha'.
- Button: 'Realizar Cadastro'.

At the bottom of the browser window, there is a green status bar that says 'You are screen sharing' and a red button that says 'Stop Share'.

BACKEND:



The screenshot shows the Swagger UI interface for the 'Criar Conta de Usuários' endpoint. The endpoint is a POST request to '/api/criar-conta' with a description 'ENDPOINT para cadastro de usuários.'.

The 'Parameters' section shows a required parameter 'dto' of type 'object' (body) with a description 'dto'. The 'Example Value' is shown as a JSON object:

```
{
  "email": "string",
  "nome": "string",
  "senha": "string"
}
```

The 'Parameter content type' is set to 'application/json'.

The 'Responses' section is empty, and the 'Response content type' is set to '*/'.

Para isso, precisamos fazer com que o angular seja capaz de capturar todos os dados preenchidos no formulário. E para isso vamos usar uma biblioteca nativa do angular chamada de: **@angular/forms**

Esta biblioteca permite a criação de formulários reativos. Ou seja, formulários capazes de capturar validar informações no FRONTEND em tempo real.

Ativando e configurando a biblioteca para desenvolvimento de formulários:
/app.module.ts

** Os arquivos de extensão .ts são programados em linguagem **TypeScript**

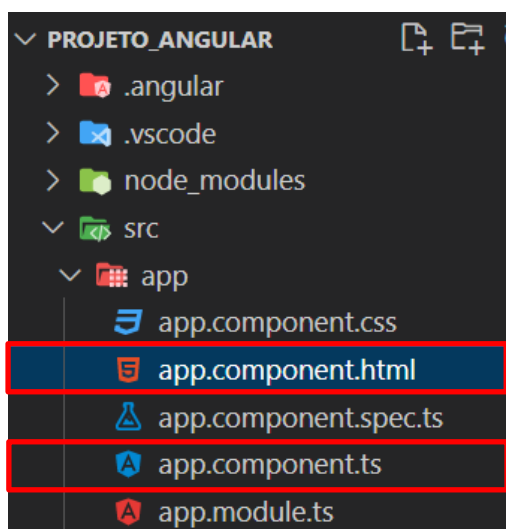
```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';

import { FormsModule, ReactiveFormsModule } from '@angular/forms';

import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    FormsModule, //formulários reativos
    ReactiveFormsModule //formulários reativos
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Para desenvolvermos a captura dos formulários, vamos utilizar um arquivo de extensão **TypeScript** voltado para programação do “Código por de trás” da página principal:



/app.component.ts

Criando a estrutura para capturar o formulário

```
import { Component } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  //criando um objeto para capturar os
  //campos do formulário
  formCriarConta = new FormGroup({
    nome: new FormControl('', []), //campo 'nome'
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //função para capturar o SUBMIT do formulário
  criarConta(): void {
    //TODO
  }

}
```

/app.component.html

Integrando o conteúdo HTML da página com a programação feita para capturar os campos do formulário.

```
<form [formGroup]="formCriarConta" (ngSubmit)="criarConta()">

  <div class="mb-3">
    <label for="inputNome" class="form-label">
      Entre com seu nome:
    </label>
    <input
      type="text"
      class="form-control"
      id="inputNome"
      placeholder="Digite o seu nome"
      required
      formControlName="nome"
    />
  </div>
</form>
```

```
</div>

<div class="mb-3">
  <label for="inputEmail" class="form-label">
    Entre com seu email:
  </label>
  <input
    type="email"
    class="form-control"
    id="inputEmail"
    aria-describedby="emailHelp"
    placeholder="Digite o seu e-mail"
    required
    formControlName="email"
  />
</div>

<div class="mb-3">
  <label for="inputSenha" class="form-label">
    Entre com sua senha:
  </label>
  <input
    type="password"
    class="form-control"
    id="inputSenha"
    placeholder="Digite a sua senha"
    required
    formControlName="senha"
  />
</div>

<div class="d-grid gap-2">
  <button class="btn btn-primary" type="submit">
    Realizar Cadastro
  </button>
</div>

</form>
```

Exibindo os dados capturados:

```
import { Component } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';

@Component({
```

```
selector: 'app-root',
templateUrl: './app.component.html',
styleUrls: ['./app.component.css']
})
export class AppComponent {

  //criando um objeto para capturar os
  //campos do formulário
  formCriarConta = new FormGroup({
    nome: new FormControl('', []), //campo 'nome'
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

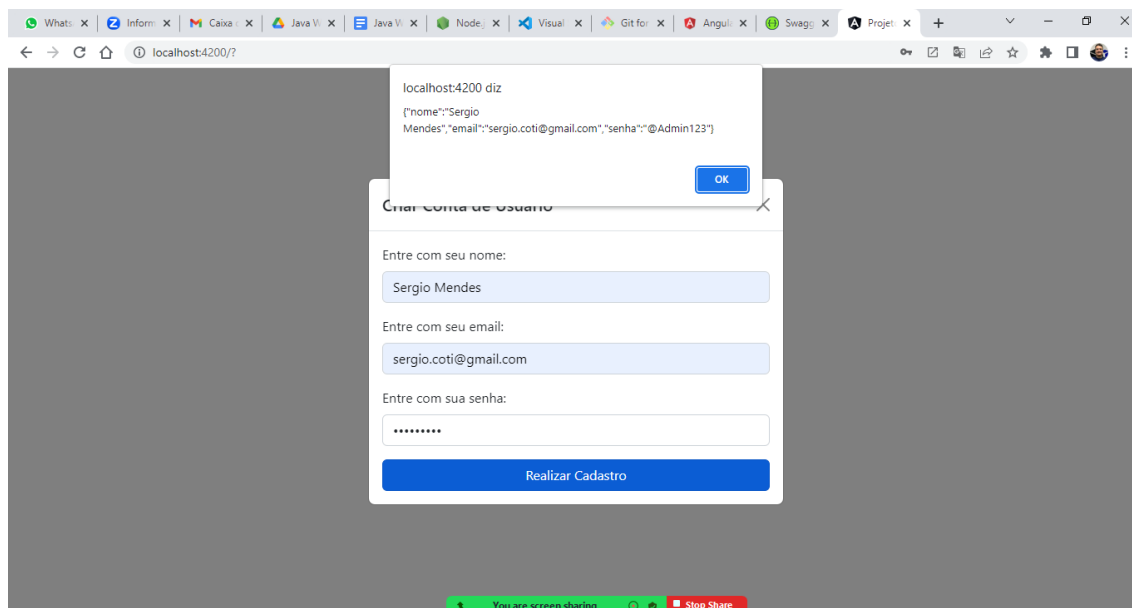
  //função para capturar o SUBMIT do formulário
  criarConta(): void {

    //capturando os valores preenchidos no formulário
    var dados = this.formCriarConta.value;

    //Teste!
    alert(JSON.stringify(dados));

  }
}
```

Ao executar, vemos que os dados preenchidos no formulário são capturados com sucesso:



Para que possamos integrar outras aplicações com uma API JAVA, precisamos habilitar na API quais projetos poderão fazer requisições (chamadas) para os seus ENDPOINTS.

O nome desta configuração chama-se
CORS (CROSS ORIGIN RESOURCE SHARING)

```
package br.com.cotiinformatica.application.controllers;

import javax.validation.Valid;

import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.http.HttpStatus;
import org.springframework.http.ResponseEntity;
import org.springframework.web.bind.annotation.CrossOrigin;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;
import org.springframework.web.bind.annotation.RestController;

import br.com.cotiinformatica.application.dtos.PostAutenticarDTO;
import br.com.cotiinformatica.application.dtos.ResponseAutenticarDTO;
import br.com.cotiinformatica.domain.interfaces.IUsuarioDomainService;
import io.swagger.annotations.Api;
import io.swagger.annotations.ApiOperation;

@Api(tags = "Autenticação de usuários")
@RestController
public class AutenticarController {

    @Autowired // injeção de dependência
    private IUsuarioDomainService usuarioDomainService;

    @CrossOrigin(origins = "*")
    @ApiOperation("ENDPOINT para autenticação de usuários e obtenção de Token.")
    @PostMapping("/api/autenticar")
    public ResponseEntity<ResponseAutenticarDTO> post
        (@Valid @RequestBody PostAutenticarDTO dto) {

        ResponseAutenticarDTO response = null;

        try {

            response = usuarioDomainService.autenticar(dto);
            return ResponseEntity.status(HttpStatus.OK).body(response);
        } catch (IllegalArgumentException e) {

            response = new ResponseAutenticarDTO();
            response.setStatus(401); // UNAUTHORIZED (CLIENT ERROR)
            response.setMensagem(e.getMessage());

            return ResponseEntity.status
                (HttpStatus.BAD_REQUEST).body(response);
        } catch (Exception e) {
```



```
        response = new ResponseAutenticarDTO();
        response.setStatus(500); // INTERNAL SERVER ERROR
        response.setMensagem(e.getMessage());

        return ResponseEntity.status
            (HttpStatus.INTERNAL_SERVER_ERROR).body(response);
    }
}
```

@angular/common/http

Biblioteca utilizada no Angular para realizar as chamadas dos serviços das APIs integradas ao frontend.

/app.module.ts

```
import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { FormsModule, ReactiveFormsModule } from '@angular/forms';
import { HttpClientModule } from '@angular/common/http';

import { AppComponent } from './app.component';

@NgModule({
  declarations: [
    AppComponent
  ],
  imports: [
    BrowserModule,
    FormsModule, //formulários reativos
    ReactiveFormsModule, //formulários reativos
    HttpClientModule //executar chamadas para APIs
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

Voltando no componente:

/app.component.ts

```
import { Component } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';
import { HttpClient } from '@angular/common/http';
```

```
@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  constructor(
    private httpClient: HttpClient //injeção de dependência
  ) {

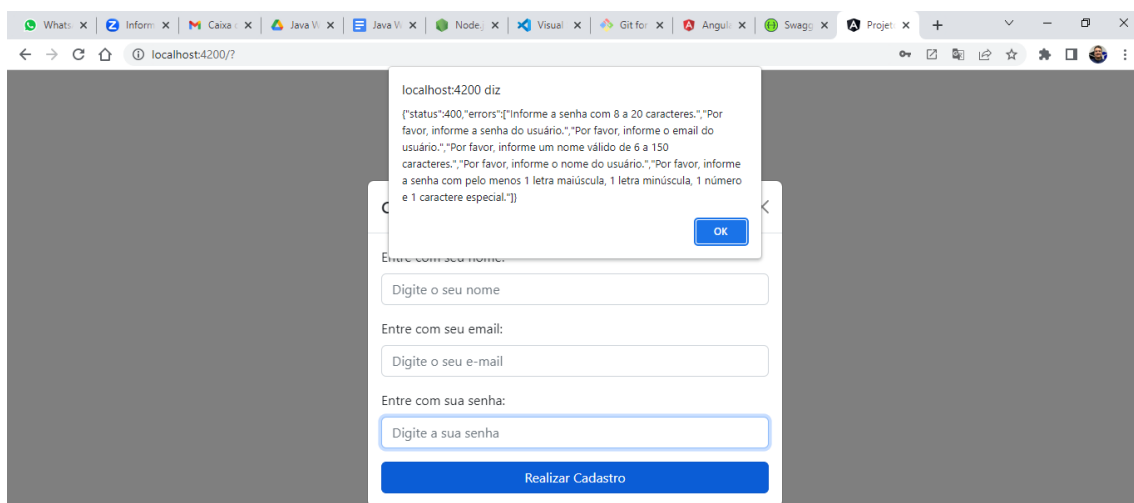
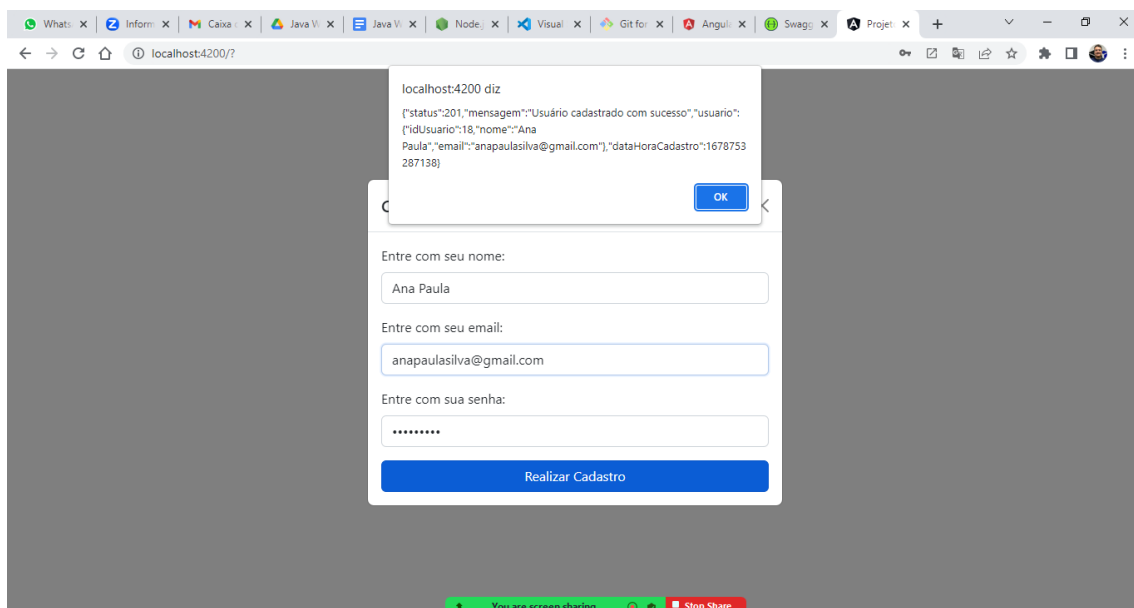
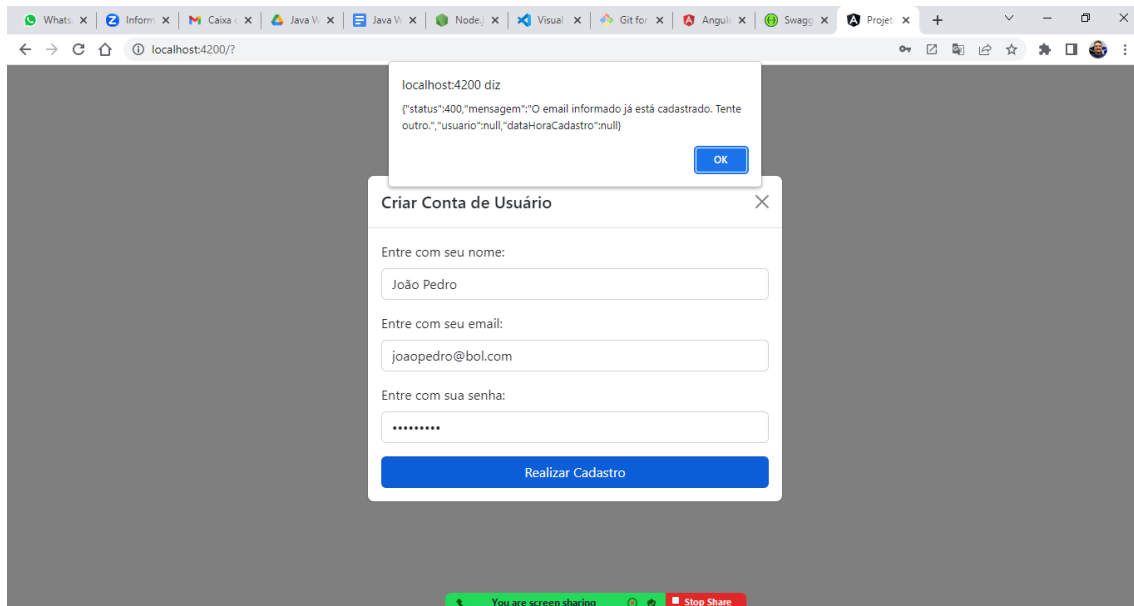
  }

  //criando um objeto para capturar os
  //campos do formulário
  formCriarConta = new FormGroup({
    nome: new FormControl('', []), //campo 'nome'
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //função para capturar o SUBMIT do formulário
  criarConta(): void {

    //capturando os valores preenchidos no formulário
    var dados = this.formCriarConta.value;

    //executando a chamada para a API
    this.httpClient.post('http://localhost:8083/api/criar-conta', dados)
      .subscribe({ //capturando a resposta da API
        next: (data) => { //sucesso!
          alert(JSON.stringify(data));
        },
        error: (e) => { //erro!
          alert(JSON.stringify(e.error));
        }
      });
  }
}
```



/app.component.ts

Capturando os demais formulários do sistema.

```
import { Component } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';
import { HttpClient } from '@angular/common/http';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  apiUrl: string = 'http://localhost:8083';

  constructor(
    private httpClient: HttpClient //injeção de dependência
  ) {

  }

  //criando um objeto para capturar os
  //campos do formulário de criação de usuário
  formCriarConta = new FormGroup({
    nome: new FormControl('', []), //campo 'nome'
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //criando um objeto para capturar os
  //campos do formulário de autenticação
  formAutenticar = new FormGroup({
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //criando um objeto para capturar os
  //campos do formulário de recuperação de senha
  formRecuperarSenha = new FormGroup({
    email: new FormControl('', []), //campo 'email'
  });

  //função para capturar o SUBMIT do formulário
  criarConta(): void {
    //capturando os valores preenchidos no formulário
    var dados = this.formCriarConta.value;
```

```
//executando a chamada para a API
this.httpClient.post(this.apiUrl + '/api/criar-conta', dados)
  .subscribe({ //capturando a resposta da API
    next: (data) => { //sucesso!
      alert(JSON.stringify(data));
    },
    error: (e) => { //erro!
      alert(JSON.stringify(e.error));
    }
  });
}

//função para capturar o SUBMIT do formulário
autenticar(): void {
  //capturando os valores preenchidos no formulário
  var dados = this.formAutenticar.value;
  //executando a chamada para a API
  this.httpClient.post(this.apiUrl + '/api/autenticar', dados)
    .subscribe({ //capturando a resposta da API
      next: (data) => { //sucesso!
        alert(JSON.stringify(data));
      },
      error: (e) => { //erro!
        alert(JSON.stringify(e.error));
      }
    });
}

//função para capturar o SUBMIT do formulário
recuperarSenha(): void {
  //capturando os valores preenchidos no formulário
  var dados = this.formRecuperarSenha.value;
  //executando a chamada para a API
  this.httpClient.post(this.apiUrl + '/api/recuperar-senha', dados)
    .subscribe({ //capturando a resposta da API
      next: (data) => { //sucesso!
        alert(JSON.stringify(data));
      },
      error: (e) => { //erro!
        alert(JSON.stringify(e.error));
      }
    });
}
}
```

/app.component.html

Capturando os demais formulários do sistema.

```
<div class="container">
  <div class="row justify-content-center mt-5">
    <div class="col-md-4">
      <div class="card">
        <div class="card-body">

          <div class="text-center">
            <h5 class="card-title text-center">
              Autenticação de Usuários</h5>
            <p>Entre com as suas credenciais de acesso</p>
          </div>

          <hr/>

          <form [formGroup]="formAutenticar" (ngSubmit)="autenticar()">

            <div class="mb-3">
              <label for="inputEmail" class="form-label">
                Entre com seu email:</label>
              <input
                type="email"
                class="form-control"
                id="inputEmail"
                aria-describedby="emailHelp"
                placeholder="Digite o seu e-mail"
                required
                formControlName="email"
              />
            </div>

            <div class="mb-3">
              <label for="inputPassword" class="form-label">
                Entre com sua senha:</label>
              <input
                type="password"
                class="form-control"
                id="inputPassword"
                placeholder="Digite a sua senha"
                required
                formControlName="senha"
              />
            </div>

            <div class="text-end mb-2">
              <a href="#" data-bs-toggle="modal" data-bs-
                target="#janelaRecuperarSenha">
                Esqueci minha senha?
              </a>
            </div>
          </form>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

        </a>
    </div>
</div>
<div class="d-grid gap-2">
    <button class="btn btn-primary" type="submit">
        Acessar Sistema
    </button>
</div>
<div class="d-grid gap-2 mt-2">
    <a href="#" class="btn btn-light" data-bs-toggle="modal"
        data-bs-target="#janelaCadastro">
        Não possui conta? <strong>Cadastre-se aqui!</strong>
    </a>
</div>
</form>
</div>
</div>

<div class="mt-2 text-center">
    <small>
        Projeto Angular<br/>Java WebDeveloper COTI Informática
    </small>
</div>

</div>
</div>
</div>

<!-- Modal para recuperação de senha do usuário -->
<div class="modal fade" id="janelaRecuperarSenha" tabindex="-1" aria-
labelledby="exampleModalLabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h1 class="modal-title fs-5" id="exampleModalLabel">
                    Esqueci minha senha</h1>
                <button type="button" class="btn-close" data-bs-dismiss="modal"
                    aria-label="Close"></button>
            </div>
            <div class="modal-body">

                <form [formGroup]="formRecuperarSenha"
                    (ngSubmit)="recuperarSenha()">

                    <div class="mb-3">
                        <label for="inputEmail" class="form-label">
                            Entre com seu email:</label>

```



```

        <input
            type="email"
            class="form-control"
            id="inputEmail"
            aria-describedby="emailHelp"
            placeholder="Digite o seu e-mail"
            required
            formControlName="email"
        />
    </div>

    <div class="d-grid gap-2">
        <button class="btn btn-primary" type="submit">
            Confirmar
        </button>
    </div>

</form>

</div>
</div>
</div>
</div>

<!-- Modal para cadastro do usuário -->
<div class="modal fade" id="janelaCadastro" tabindex="-1" aria-
labelledby="exampleModallabel" aria-hidden="true">
    <div class="modal-dialog modal-dialog-centered">
        <div class="modal-content">
            <div class="modal-header">
                <h1 class="modal-title fs-5" id="exampleModallabel">
                    Criar Conta de Usuário</h1>
                <button type="button" class="btn-close" data-bs-dismiss="modal"
                    aria-label="Close"></button>
            </div>
            <div class="modal-body">

                <form [formGroup]="formCriarConta" (ngSubmit)="criarConta()">

                    <div class="mb-3">
                        <label for="inputNome" class="form-label">
                            Entre com seu nome:</label>
                        <input
                            type="text"
                            class="form-control"
                            id="inputNome"
                            placeholder="Digite o seu nome"

```

```
        required
        formControlName="nome"
    />
</div>

<div class="mb-3">
    <label for="inputEmail" class="form-label">
        Entre com seu email:</label>
    <input
        type="email"
        class="form-control"
        id="inputEmail"
        aria-describedby="emailHelp"
        placeholder="Digite o seu e-mail"
        required
        formControlName="email"
    />
</div>

<div class="mb-3">
    <label for="inputSenha" class="form-label">
        Entre com sua senha:</label>
    <input
        type="password"
        class="form-control"
        id="inputSenha"
        placeholder="Digite a sua senha"
        required
        formControlName="senha"
    />
</div>

<div class="d-grid gap-2">
    <button class="btn btn-primary" type="submit">
        Realizar Cadastro
    </button>
</div>

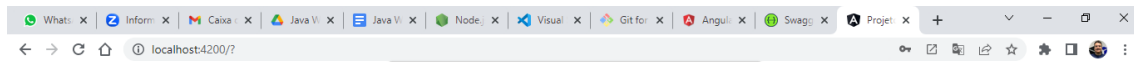
</form>

</div>

</div>

</div>

</div>
```



localhost:4200 diz
{"status":401,"mensagem":"Acesso negado. Usuário inválido.", "usuario":"null","token":null}

Entre com seu email:

sergio.coti@gmail.com

Entre com sua senha:

[Esqueci minha senha?](#)

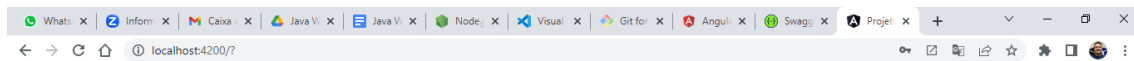
Acessar Sistema

Não possui conta? **Cadastre-se aqui!**

Projeto Angular

Java WebDeveloper COTI Informática

You are screen sharing Stop Share



localhost:4200 diz
{"status":200,"mensagem":"Usuário autenticado com sucesso.", "usuario":{"idUsuario":19,"nome":"Sergio Mendes","email":"sergio.coti@aol.com","token":"eyJhbGciOiJIUzUxMiIsInR5cGU6IjY4bnVpY290aU8hbnBzZwUyZ9tiwiYXV0aG9yaXpZXiMiOiIsImlhZC16MTY3OjDc1NDE1OCwiZXBwIjo4Njc4NzY2MTU4fQjklvXHM0d6pToXtIEA88sRe8onxT41njIEWptlWpWLSvwqATivojc654x0ybMW9PtkccC-E4tkv4n41QD9v75fC1Mh4u75

[Esqueci minha senha?](#)

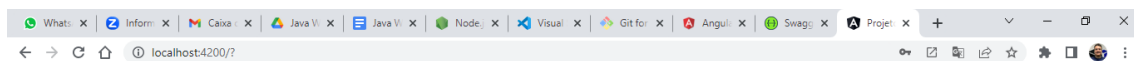
Acessar Sistema

Não possui conta? **Cadastre-se aqui!**

Projeto Angular

Java WebDeveloper COTI Informática

You are screen sharing Stop Share



localhost:4200 diz
{"status":400,"mensagem":"Email não encontrado. Usuário inválido."}

Entre com seu email:

sergio.coti@aol.com

Esqueci minha senha

Entre com seu email:

sergio.coti123123@gmail.com

Confirmar

Projeto Angular

Java WebDeveloper COTI Informática

Exibindo as mensagens do cadastro do usuário na página:

```
import { Component } from '@angular/core';
import { FormGroup, FormControl } from '@angular/forms';
import { HttpClient } from '@angular/common/http';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {

  apiUrl: string = 'http://localhost:8083';

  mensagem_criarconta: string = '';

  constructor(
    private httpClient: HttpClient //injeção de dependência
  ) {

  }

  //criando um objeto para capturar os
  //campos do formulário de criação de usuário
  formCriarConta = new FormGroup({
    nome: new FormControl('', []), //campo 'nome'
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //criando um objeto para capturar os
  //campos do formulário de autenticação
  formAutenticar = new FormGroup({
    email: new FormControl('', []), //campo 'email'
    senha: new FormControl('', []) //campo 'senha'
  });

  //criando um objeto para capturar os
  //campos do formulário de recuperação de senha
  formRecuperarSenha = new FormGroup({
    email: new FormControl('', []), //campo 'email'
  });

  //função para capturar o SUBMIT do formulário
  criarConta(): void {
```

```
//capturando os valores preenchidos no formulário
var dados = this.formCriarConta.value;
//executando a chamada para a API
this.httpClient.post(this.apiUrl + '/api/criar-conta', dados)
  .subscribe({ //capturando a resposta da API
    next: (data: any) => { //sucesso!
      this.mensagem_criarconta = data.mensagem;
      this.formCriarConta.reset(); //limpar o formulário
    },
    error: (e) => { //erro!
      if(e.error.mensagem){
        this.mensagem_criarconta = e.error.mensagem;
      }
      else if(e.error.errors){
        this.mensagem_criarconta = e.error.errors;
      }
    }
  });
}

//função para capturar o SUBMIT do formulário
autenticar(): void {
  //capturando os valores preenchidos no formulário
  var dados = this.formAutenticar.value;
  //executando a chamada para a API
  this.httpClient.post(this.apiUrl + '/api/autenticar', dados)
    .subscribe({ //capturando a resposta da API
      next: (data) => { //sucesso!
        alert(JSON.stringify(data));
      },
      error: (e) => { //erro!
        alert(JSON.stringify(e.error));
      }
    });
}

//função para capturar o SUBMIT do formulário
recuperarSenha(): void {
  //capturando os valores preenchidos no formulário
  var dados = this.formRecuperarSenha.value;
  //executando a chamada para a API
  this.httpClient.post(this.apiUrl + '/api/recuperar-senha', dados)
    .subscribe({ //capturando a resposta da API
      next: (data) => { //sucesso!
        alert(JSON.stringify(data));
      },
      error: (e) => { //erro!
```

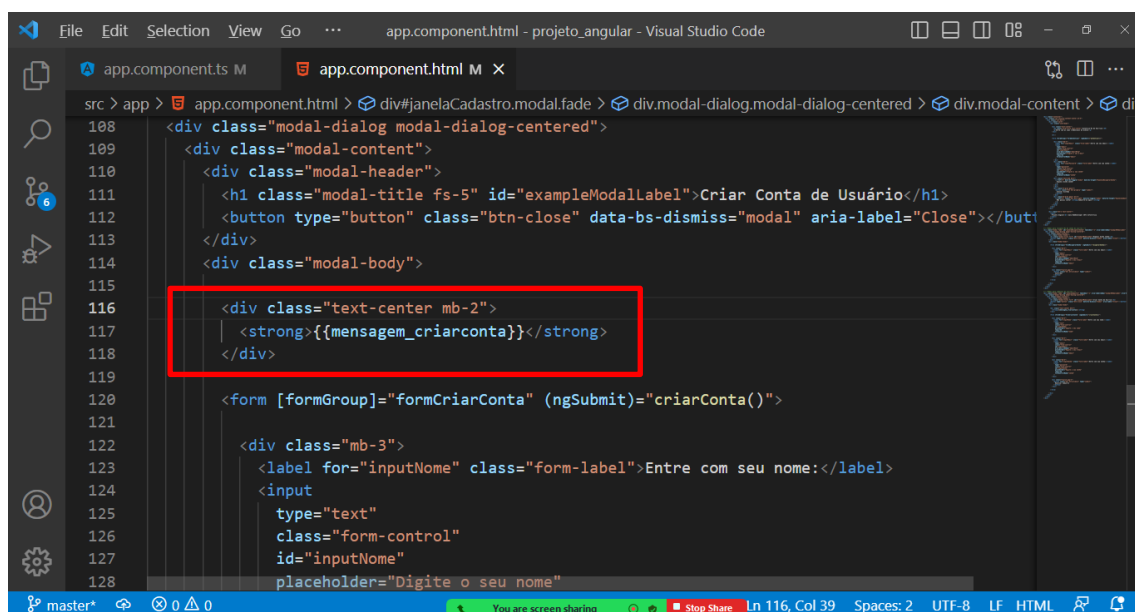
```

        alert(JSON.stringify(e.error));
    }

    });
}
}

```

Na página:

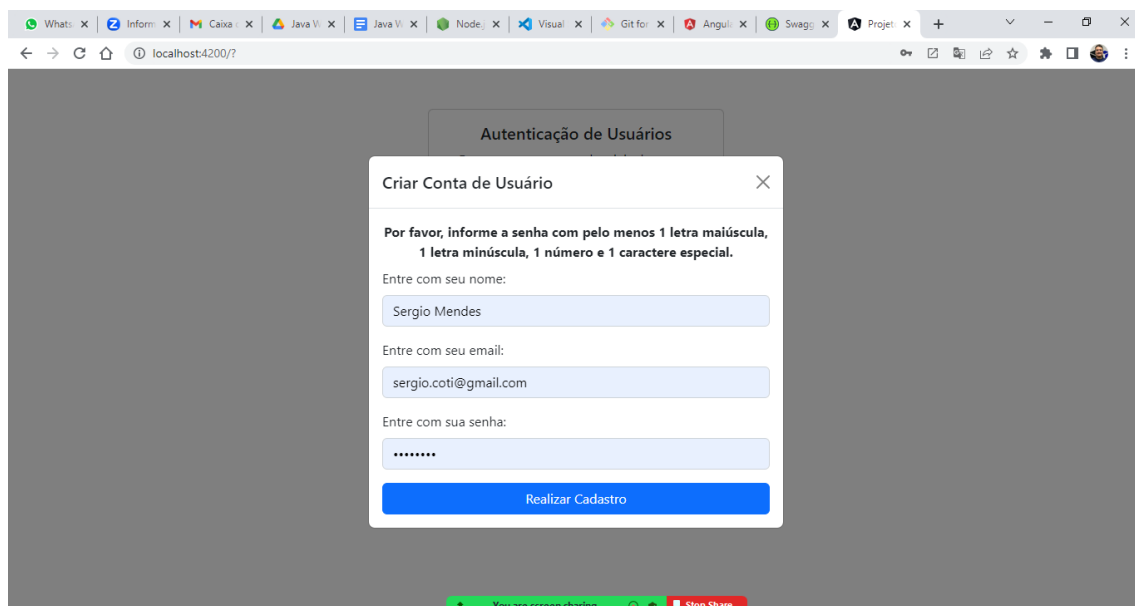


```

108 <div class="modal-dialog modal-dialog-centered">
109   <div class="modal-content">
110     <div class="modal-header">
111       <h1 class="modal-title fs-5" id="exampleModalLabel">Criar Conta de Usuário</h1>
112       <button type="button" class="btn-close" data-bs-dismiss="modal" aria-label="Close"></button>
113     </div>
114     <div class="modal-body">
115
116       <div class="text-center mb-2">
117         <strong>{{mensagem_criarconta}}</strong>
118       </div>
119
120       <form [formGroup]="formCriarConta" (ngSubmit)="criarConta()">
121
122         <div class="mb-3">
123           <label for="inputNome" class="form-label">Entre com seu nome:</label>
124           <input
125             type="text"
126             class="form-control"
127             id="inputNome"
128             placeholder="Digite o seu nome"

```

<http://localhost:4200/>



What's: x Inform: x Caixa: x Java Vi: x Java Vi: x Node: x Visual: x Git for: x Angul: x Swag: x Proj: x + - x

← → ↻ 🏠 ⓘ localhost:4200/?

Autenticação de Usuários
Entre com as suas credenciais de acesso

Criar Conta de Usuário

O email informado já está cadastrado. Tente outro.

Entre com seu nome:

Entre com seu email:

Entre com sua senha:

Realizar Cadastro

You are screen sharing Stop Share

What's: x Inform: x Caixa: x Java Vi: x Java Vi: x Node: x Visual: x Git for: x Angul: x Swag: x Proj: x + - x

← → ↻ 🏠 ⓘ localhost:4200/?

Autenticação de Usuários
Entre com as suas credenciais de acesso

Criar Conta de Usuário

Usuário cadastrado com sucesso

Entre com seu nome:

Entre com seu email:

Entre com sua senha:

Realizar Cadastro

You are screen sharing Stop Share