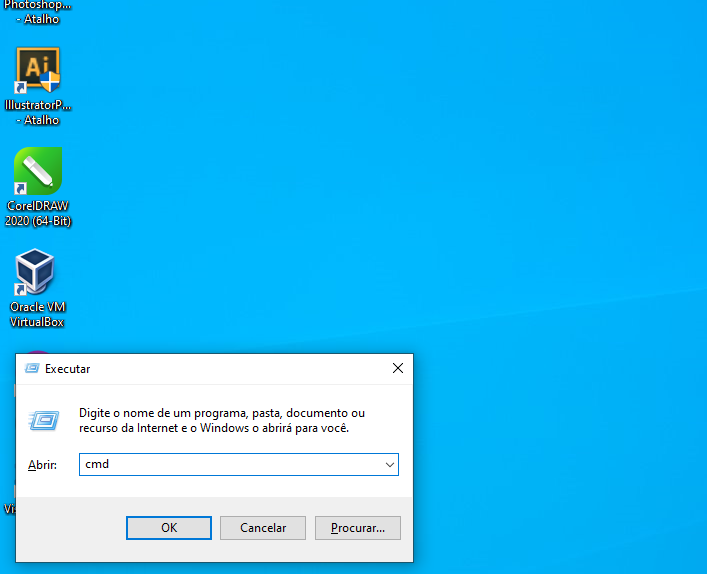
Nome: Guilherme Antonio Martos 2°DS-A Prof.: Ronan

Link do projeto no GitHub:

Vamos criar um projeto utilizando Ionic. Para começar vamos abrir o prompt de comando do Windows.

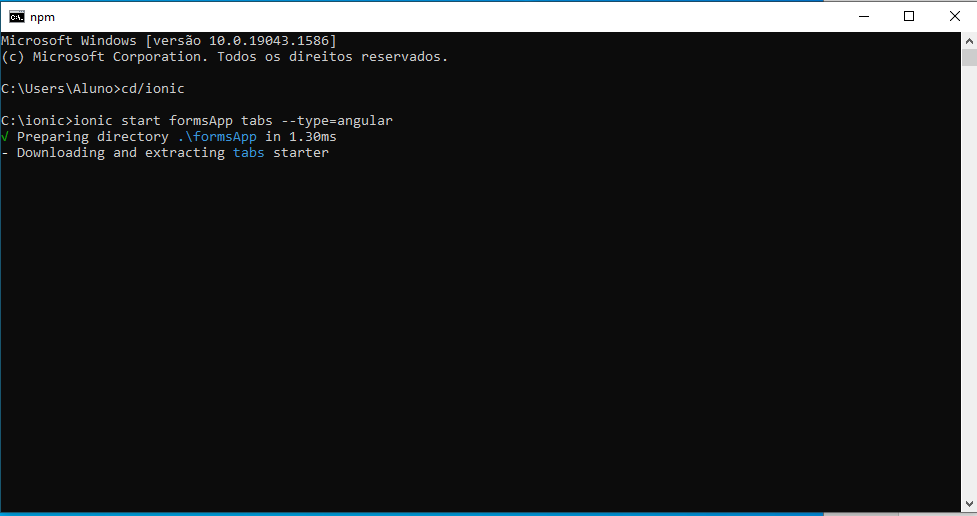
Utilizamos a tecla Windows+R e digitamos cmd e damos “Enter”.

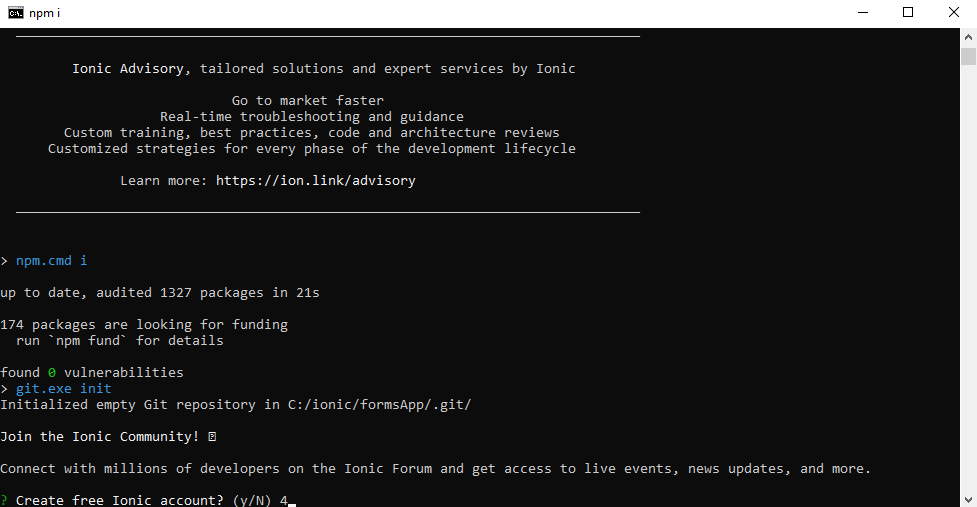


Depois no promp digitamos o seguinte comando:

cd/ionic - para entrar na pasta ionic

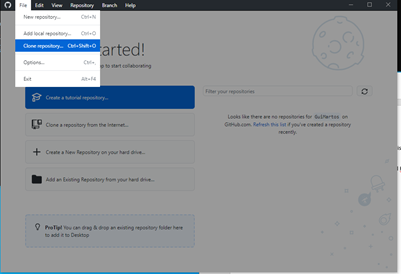
Depois  
ionic start formsApp tabs –type=angular – para criar o projeto

Dessa maneira e damos “Enter”. Agora é só esperar criar o projeto

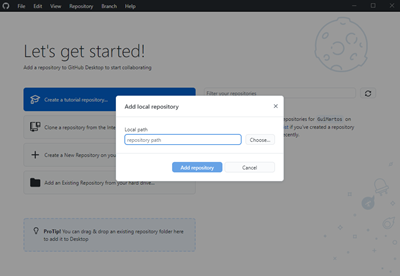
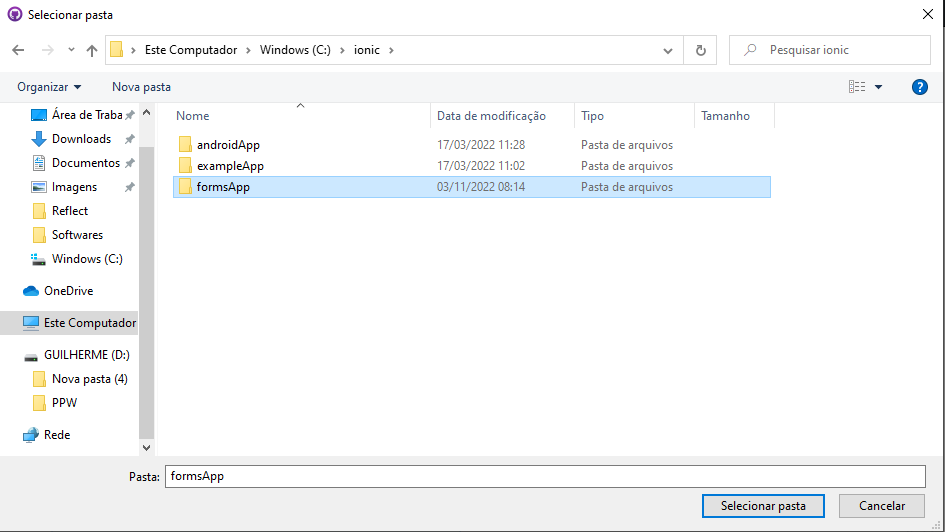
Se você chegou aqui, é porque deu tudo certo. Aperte a tecla “N” e de “Enter”

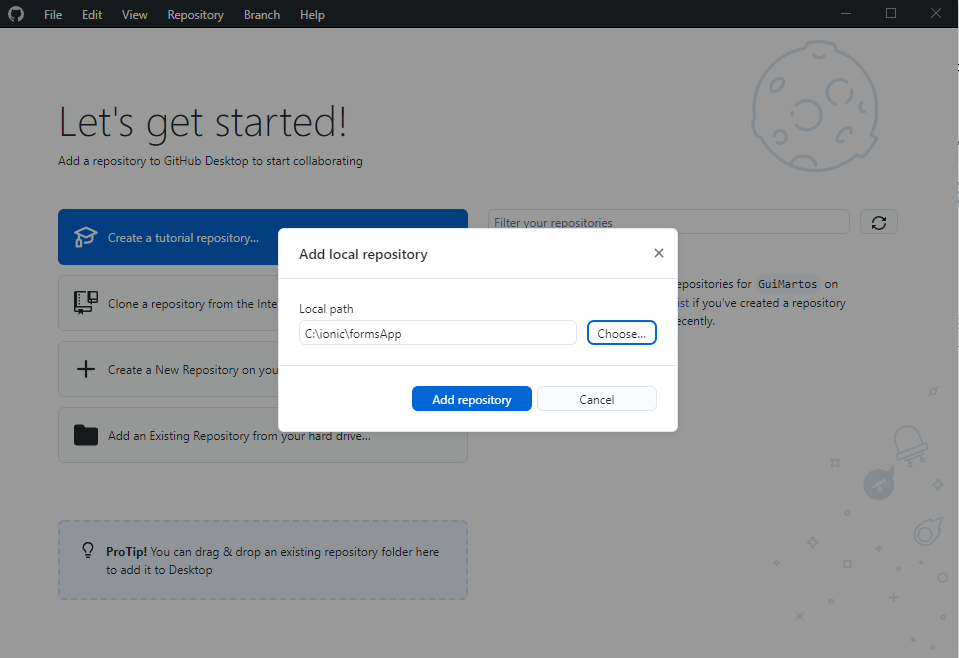
Logo após isso vamos enviar para o GitHub, para isso você já deve estar logado em sua conta no GitHub Desktop.

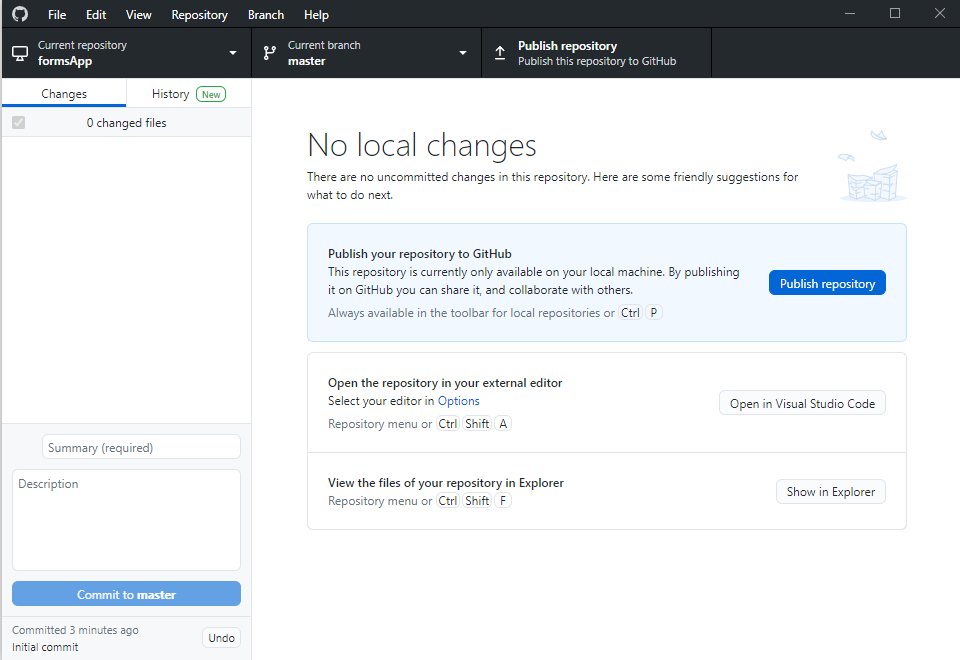
Abra o GitHub Desktop e vá até “File”, “Add Local Repository”



Depois clicamos em “Choose” e fazemos o caminho até a pasta onde está o nosso projeto



Em seguida clique no botão azul

Depois clique em “Publish repository”

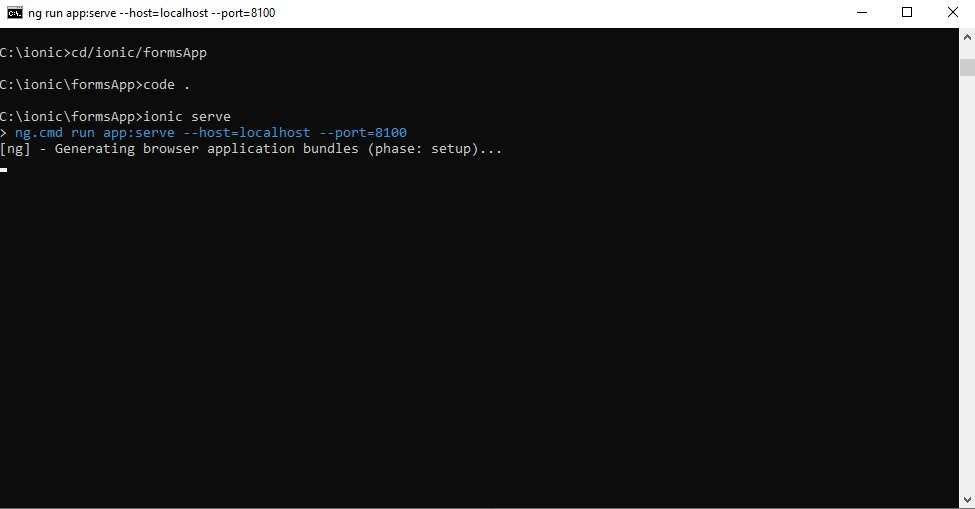
Agora vamos começar a abrir o projeto.

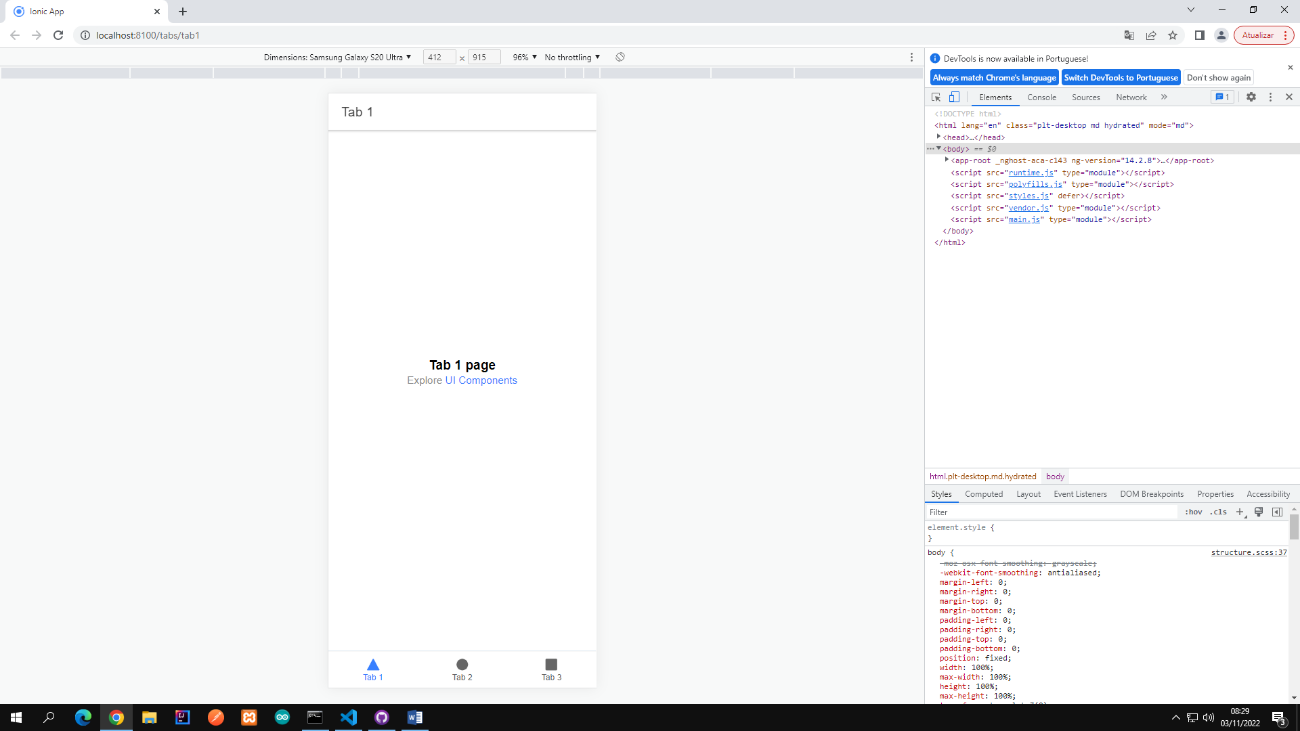
Para isso voltamos ao prompt e digitamos o comando

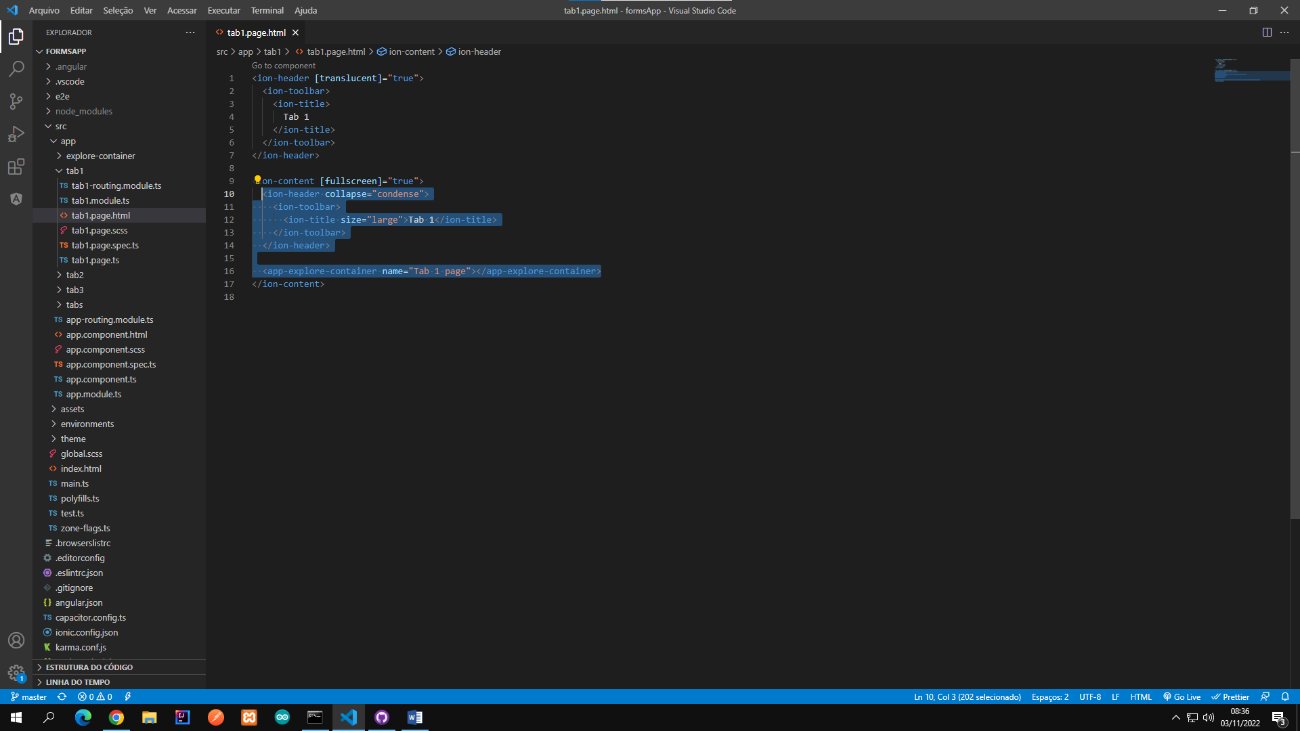
cd/ionic/fromsApp - para ir para a pasta do projeto

Code . – para abrir o VSCode

E depois

ionic serve - para abrir o nosso projeto no navegador para ver como ele está ficando.

Se apareceu isso, é porque deu tudo certo

Agora no VSCode, vamos apagar tudo o que está dentro do “ion-content”

Agora no prompt vamop apertar a tecla Ctrl+C para parar o server, pois vamos criar duas páginas, a “login” e a “registro”

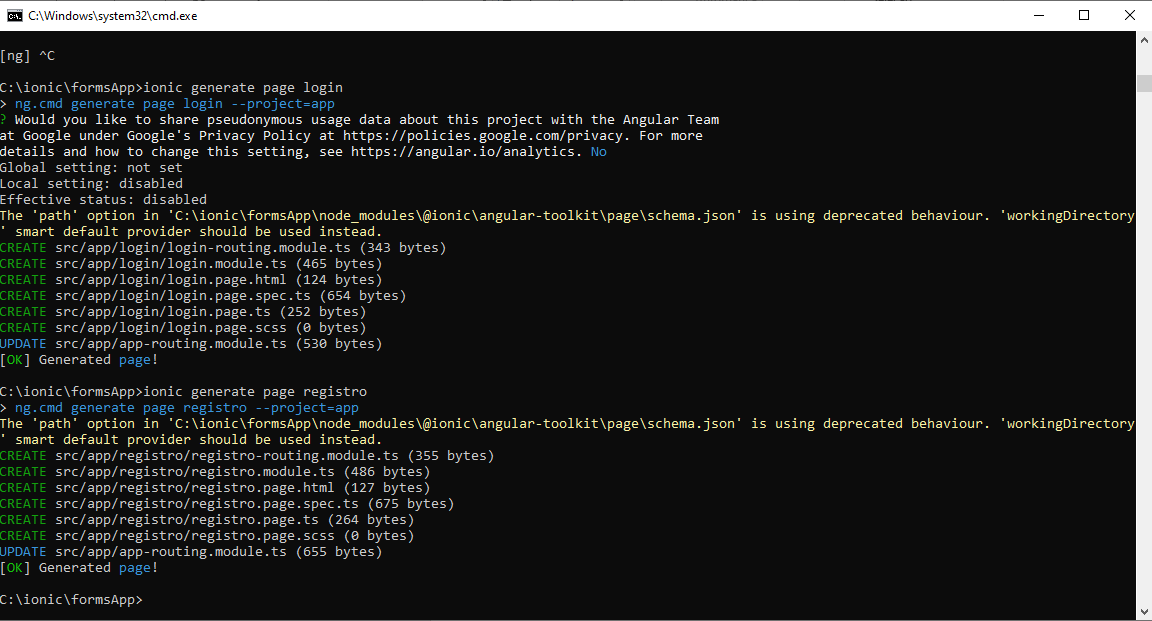
Digitamos no prompt

Ionic generate page login – para criar a página “login”

Damos “Enter”

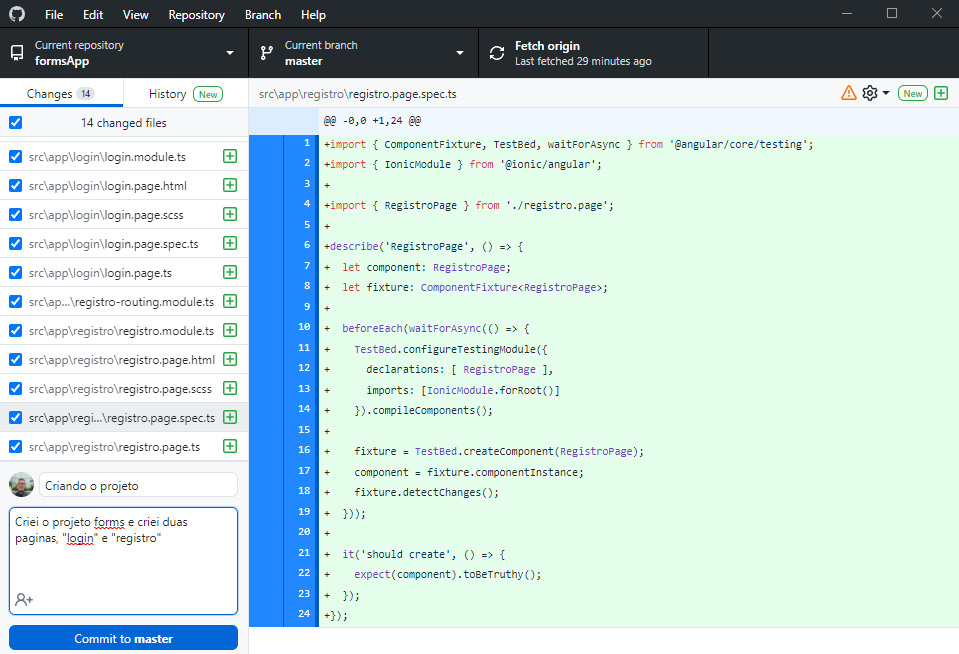
Apertamos “N”

Damos “Enter”

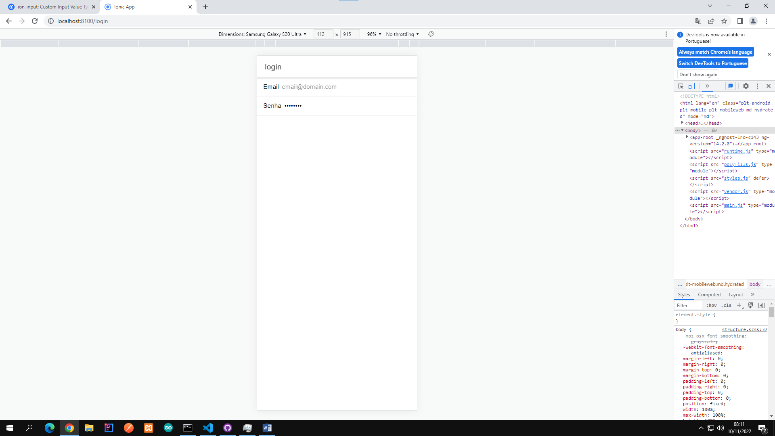
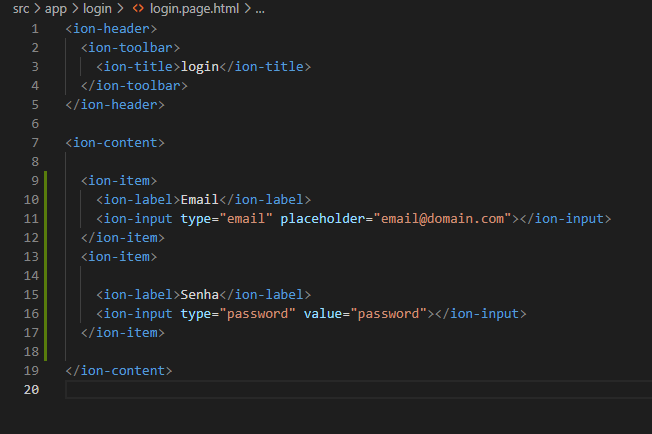
Ionic generate page registro – para criar a página “registro”

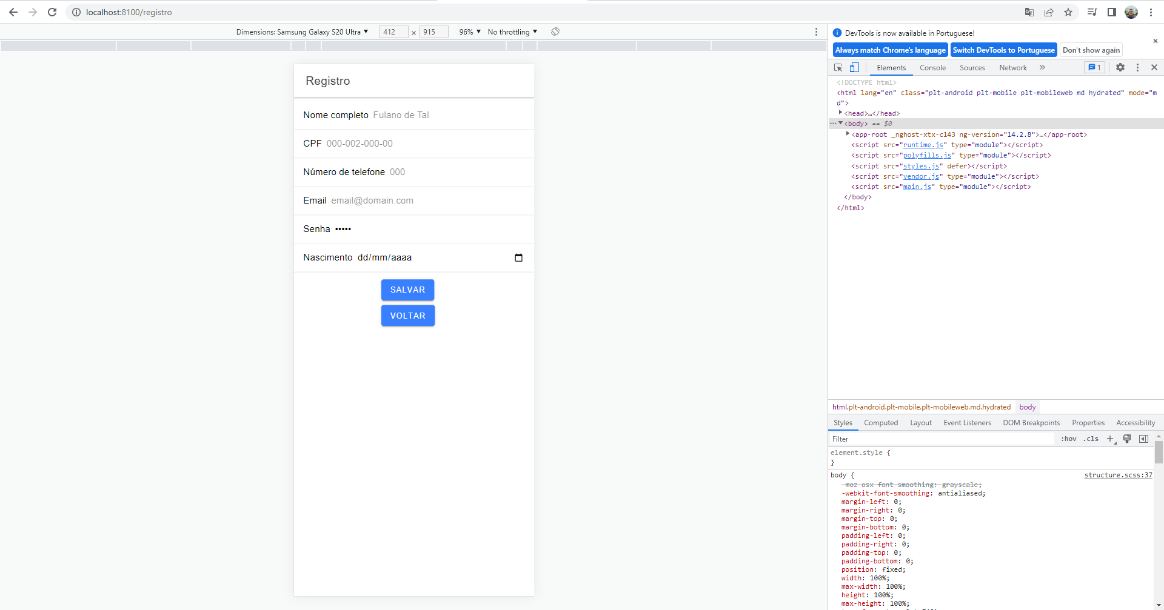
Agora vamos no GitHub.

Vamos dar o “Comit” para isso digitamos no campo “Comit” tudo o que nós fizemos até agora, podemos dar um dar um título e fazer uma descrição. Exemplo:



Apertamos o botão azul e depois clicamos em “Push origin”

Para começar a página de login, vamos começar pelo input que serve para colocar informações colocamos dois de nossa preferência, um para email e outro para senha. Depois só personalizar da forma que achar melhor.

Para a página de registro devemos colocar todos os campos da imagem e depois personalizarmos da forma que acharmos melhor.

Texto

Descrição gerada automaticamenteInterface gráfica do usuário, Aplicativo

Descrição gerada automaticamenteNo tab 1 colocaremos botões para “Login” e “Registrar-se”

Devemos colocar o o link para podermos acessar as outras págias do app, para isso fazemos dessa forma:

Utilizamos o routerLink = “caminho”

Tela de celular com aplicativo aberto

Descrição gerada automaticamente

No meu caso ficou dessa forma, depois é só ir aplicando a cada link que tiver no app.

Agora vamos fazer o formulário funcionar.

Acesse este link: <https://github.com/ionic-team/ionic-storage>

Rode os dois códigos de instalação deste link

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Depois de rodados e instalados no prompt de comando, vamos no vsCode e procuramos a página app.module.ts

E colocamos a seguinte importação:

import { IonicStorageModule } from '@ionic/storage-angular';

Depois mais para baixo no código vamos organizar os imports para ficar mais fácil de localizarmos as coisas. Um a baixo do outro.

Dessa forma, sempre no final de tudo apertando as teclas Ctrl+shift+F para formatar o código de uma maneira melhor para enxergarmos/compreendermos.

Texto

Descrição gerada automaticamente

Como vamos utilizar o navegador para simular nosso app, vamos ter que fazer algumas importações.

IonicStorageModule.forRoot({

name: '\_\_mydb',

driverOrder: [Drivers.IndexedDB, Drivers.LocalStorage]

})

Deixando dessa forma aqui, caso de erro, vá até o final da palavra “Drivers” e aperte Ctrl+Tecla espaço e faça a importação dos drivers.

Texto

Descrição gerada automaticamenteTexto

Descrição gerada automaticamenteA parte da declaração de nome do Banco de dados vamos apagar, para que o Ionic utilize o nome padrão.

Agora vamos criar o serviço para que faça o armazenamento de dados, para isso vamos utiliza o seguinte comando no prompt de comando.

ionic generate service services/storage

Após isso verifique se foi criado a pasta services no VScode e vá até a storage.services.ts.

Depois disso vá até o git hub de onde estamos aproveitando os códigos.

private \_storage: Storage | null = null;

constructor(private storage: Storage) {

this.init();

}

async init() {

// If using, define drivers here: await this.storage.defineDriver(/\*...\*/);

const storage = await this.storage.create();

this.\_storage = storage;

}

// Create and expose methods that users of this service can

// call, for example:

public set(key: string, value: any) {

this.\_storage?.set(key, value);

}

Este será o código que colocaremos no storage.services.ts. após apagar o constructor.

Depois dará um erro falando que não podemos utilizar \_ antes de variáveis, para corrigir isso passamos com o mouse em cima do erro e clicamos em correção rápida e depois clicamos como na imagem.

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Vamos criar um outro método para buscarmos informações

Abaixo do método set vamos criar este

  public get(key: string) {

    this.\_storage?.get(key);

  }

}

Agora faremos outro import para que não de problema futuramente

import { Storage } from '@ionic/storage-angular';

Texto

Descrição gerada automaticamenteAgora vamos fazer um método para remover dados do banco de dados:

  public remove(key: string) {

    this.\_storage?.remove(key);

  }

Agora vamos fazer um método para buscar todos os dados do banco de dados de uma vez:

 public getAll(){

    const lista = [];

    this.\_storage.forEach((value, key, index) =>{

      lista.push(value);

    });

    return lista;

  }

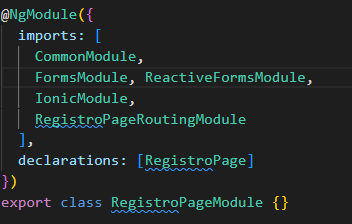
Vamos criar a página para registro de produtos para isso utilizamos o mesmo comando que vimos lá em cima, mas dessa vez vamos mudar o nome para produto.

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamenteNesta página deve ser feito o registro de um produto por tanto deve ter o nome do produto, descrição, valor e validade.

Deixei dessa forma.

Para fazer este formulário funcionar vamos começar as importações no registro.module.ts

Vamos completar este import assim e nos imports assim

Agora vamos para registro.page.ts

Fazemos o seguinte import

import {FormBuilder, FormGroup, Validators} from '@angular/forms';

Depois criamos a variável dentro do export class

  formsCadastro: FormGroup;

No constructor

export class RegistroPage implements OnInit {

  formsCadastro: FormGroup;

  constructor(private formBuilder : FormBuilder) {

    this.formsCadastro = this.formBuilder.group({

    });

   }

  ngOnInit() {

  }

}

Agora vamos começar o nosso BD dentro disso

Texto

Descrição gerada automaticamente

Agora vamos fazer as validações de alguns campos para isso vamos criar alguns métodos

Texto

Descrição gerada automaticamente

Para que isso funcione vamos fazer a ligação com o HTML

Para isso embaixo do ion-content deve ter a tag form e com fechamento antes do ion-content

<ion-content class="fundo">

  <form [formGroup]="formsCadastro">

Dessa forma.

Agora para fazer a ligação vamos utilizar dentro do input o seguinte comando

formControlName="nome que colocou no page.ts"

Texto

Descrição gerada automaticamente

No botão deixamos assim:

<ion-button class="button" expand="block" (click) ="salvarCadastro" color="primary">Salvar</ion-button>