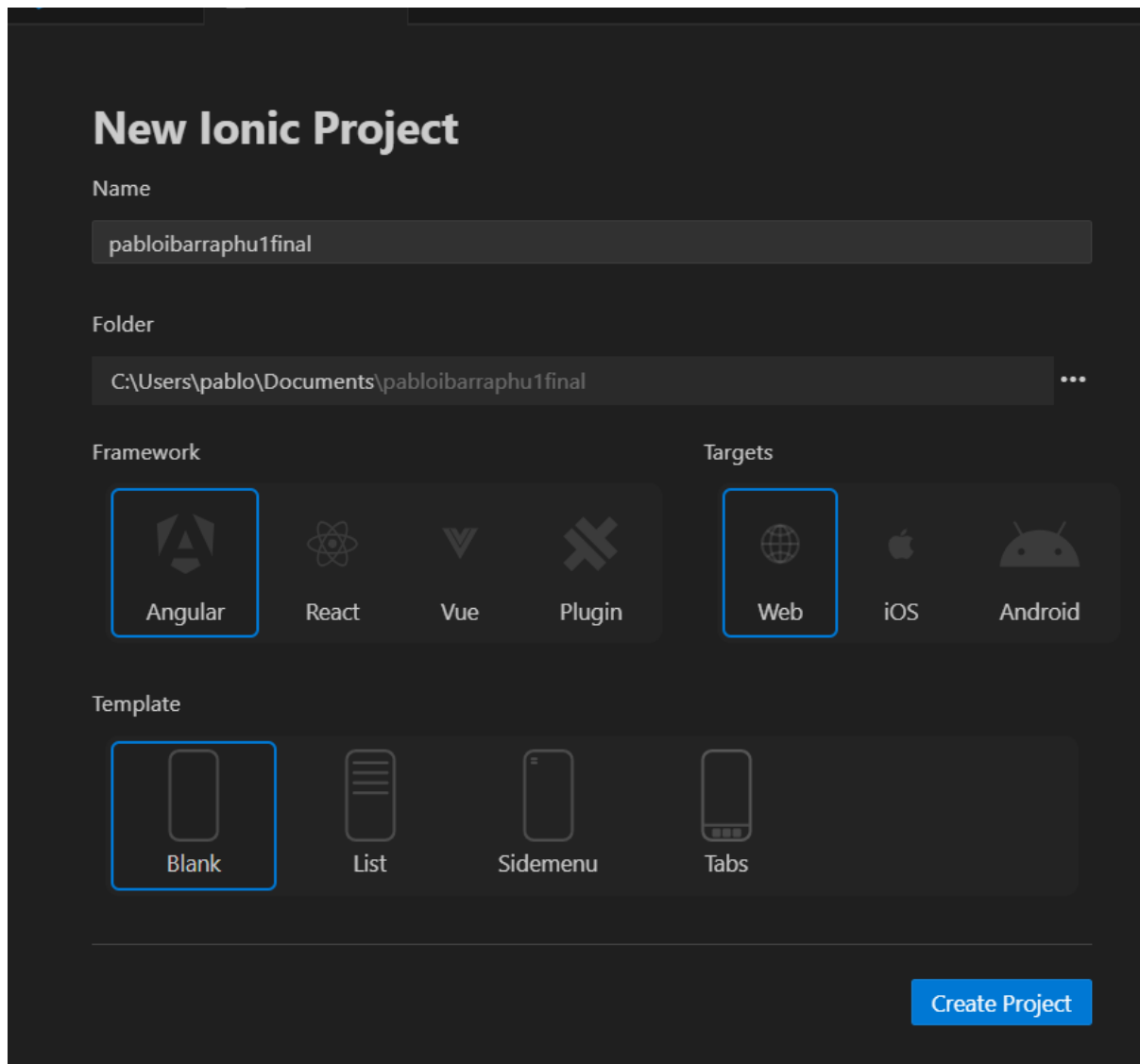
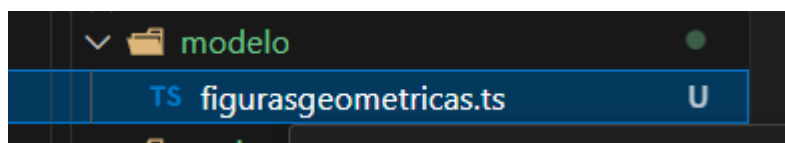


Prueba Programación Híbrida Unidad 1 Pablo Ibarra

Primeramente, se crea el proyecto utilizando la plantilla en blanco

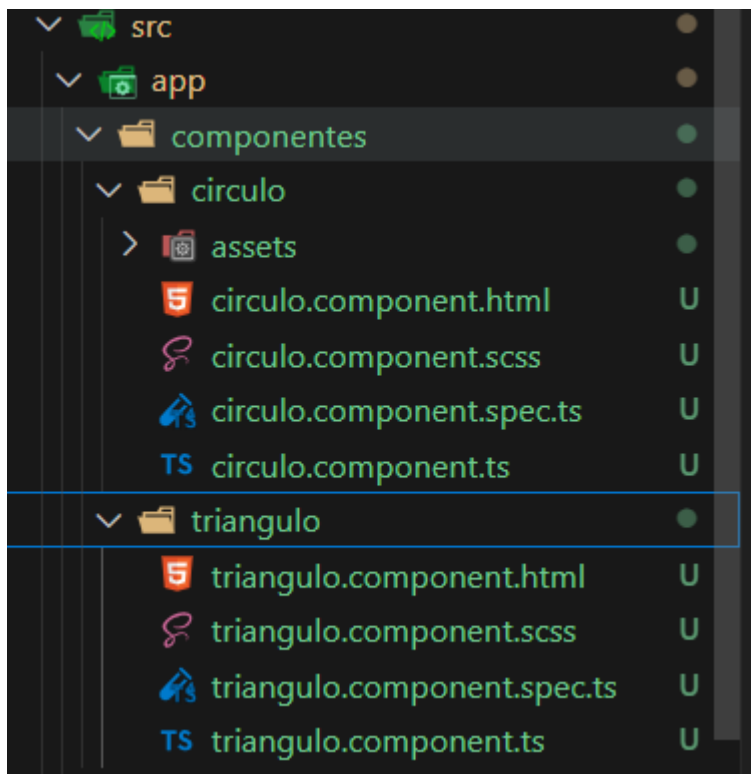


Se crea la carpeta que llamaremos Modelo y dentro creamos la jerarquía de la figuras geométricas.



```
TS figurasgeometricas.ts U X  circulo.component.html U  TS circulo.component.ts U
src > app > modelo > TS figurasgeometricas.ts > TrianguloEscaleno > constructor
1  export class figurasgeometricas {
2      constructor(public radio:number,) {}
3  }
4
5  export class circulo extends figurasgeometricas {
6      constructor(radio:number,){
7          super(radio);
8      }
9
10
11      calcularPerimetro(): number {
12          return 2 * Math.PI * this.radio;
13      }
14  }
15
16  export class TrianguloEscaleno extends figurasgeometricas {
17      constructor(
18          public ladoA: number,
19          public ladoB: number,
20          public ladoC: number
21      ) {
22          super(ladoA);
23          super (ladoB)
24          super (ladoC)
25      }
26
27      calcularPerimetro(): number {
28          return this.ladoA + this.ladoB + this.ladoC;
29      }
30  }
31
32
```

Aquí creamos la clase figuras geométricas que define una figura geométrica básica , luego creamos la clase circulo que es heredada de figurasgeometricas y utiliza el radio y ejecuta un método para calcular el perímetro del círculo, y por último la clase trianguloescaleno que definimos el triángulo con tres lados y tiene un método para calcular el perímetro.



Creamos los componentes circulo y triangulo en una carpeta llamada componentes
Y luego procedemos a modificar .

```
src > app > componentes > circulo > circulo.component.html > ...
Go to component
1  <ion-list>
2  | <ion-card>
3  | | <ion-img src="assets/perimetrocirculo.png" style="width: 300px; height: auto;"></ion-img>
4  | </ion-card>
5
6
7
8  | <ion-item>
9  | | <ion-input type="number" label="radio" [(ngModel)]="radio" placeholder="Ingrese el radio"></ion-input>
10 | </ion-item>
11
12
13
14 | <ion-item>
15 | | <ion-button expand="full" (click)="calcularPerimetro()">Calcular Perimetro</ion-button>
16 | </ion-item>
17
18 | <ion-item *ngIf="resultado !== null">
19 | | <ion-label>El perímetro es: {{ resultado }}</ion-label>
20 | </ion-item>
21 </ion-list>
22
```

En este código creamos una interfaz de usuario interactiva utilizando componentes de Ionic lo que nos permite como usuario ingresar el radio de un círculo, calcular su perímetro al hacer clic en un botón y mostrar el resultado en pantalla. Además, colocamos una imagen referencial al cálculo del área de un círculo.

```

import { Component, OnInit } from '@angular/core';
import {IonicModule} from '@ionic/angular'
import { FormsModule } from '@angular/forms';

@Component({
  selector: 'app-circulo',
  templateUrl: './circulo.component.html',
  styleUrls: ['./circulo.component.scss'],
  standalone: true,
  imports: [IonicModule,FormsModule,]
})
export class CirculoComponent implements OnInit {
  calcularPerimetro() {
    throw new Error('Method not implemented.');
```



```

    radio:number = 20;
    resultado:number | null = null;

    constructor() { }





    ngOnInit() {}

  }

```

En este código definimos un componente llamado CirculoComponent utilizando módulos de Ionic y Angular Forms. Este componente está preparado para que nos permita a nosotros como usuario ingresar el radio de un círculo y calcular su perímetro. Aunque el método calcularPerimetro aún no está implementado y lanza un error que no pudimos solucionar, las propiedades radio y resultado están configuradas para manejar los valores necesarios para el cálculo y que nos muestre el resultado en pantalla. El componente también incluye un método ngOnInit que se ejecutará al inicializarse.

```

src > app > componentes > triangulo >  triangulo.component.html >  ion-card >  ion-card-content >  ion-text
Go to component
1 <ion-card>
2   <ion-card-header>
3     <ion-card-title>Triángulo Escaleno</ion-card-title>
4   </ion-card-header>
5   <ion-card-content>
6     <form (ngSubmit)="calcularPerimetro()">
7       <ion-card>
8         <ion-img src="assets/perimetrotriangulo.png" style="width: 300px; height: auto;"></ion-img>
9       </ion-card>
10      <ion-item>
11        <ion-label>Lado A</ion-label>
12        <ion-input type="number" [(ngModel)]="ladoA" name="ladoA"></ion-input>
13      </ion-item>
14      <ion-item>
15        <ion-label>Lado B</ion-label>
16        <ion-input type="number" [(ngModel)]="ladoB" name="ladoB"></ion-input>
17      </ion-item>
18      <ion-item>
19        <ion-label>Lado C</ion-label>
20        <ion-input type="number" [(ngModel)]="ladoC" name="ladoC"></ion-input>
21      </ion-item>
22
23      <ion-button expand="full" type="submit">Calcular Perímetro</ion-button>
24    </form>
25    <ion-text *ngIf="resultado">
26      <p>El perímetro es: {{ resultado }}</p>
27    </ion-text>
28  </ion-card-content>
29 </ion-card>
30

```

Aquí este código creamos una interfaz de usuario interactiva utilizando componentes de Ionic que nos permite como usuario ingresar los valores de los lados de un triángulo escaleno y calcular su perímetro al hacer clic en un botón. Pusimos una imagen referencial sobre el cálculo del perímetro del triángulo para mejorar la experiencia visual. El resultado del cálculo se muestra en pantalla solo si colocamos los datos de cada lado.

```

src > app > componentes > triangulo > TS triangulo.component.ts > TrianguloComponent
1  import { IonicModule } from '@ionic/angular';
2  import { Component } from '@angular/core';
3  import { FormsModule } from '@angular/forms';
4  import { TrianguloEscaleno } from 'src/app/modelo/figurasgeometricas';
5
6
7  @Component({
8    selector: 'app-triangulo',
9    templateUrl: './triangulo.component.html',
10   styleUrls: ['./triangulo.component.scss'],
11   standalone: true,
12   imports: [IonicModule, FormsModule, TrianguloComponent]
13 })
14 export class TrianguloComponent {
15   ladoA!: number;
16   ladoB!: number;
17   ladoC!: number;
18   resultado: number | undefined;
19
20   calcularPerimetro() {
21     const triangulo = new TrianguloEscaleno(this.ladoA, this.ladoB, this.ladoC);
22     this.resultado = triangulo.calcularPerimetro();
23   }
24 }

```

Aquí definimos un componente de Angular llamado TrianguloComponent utilizando módulos de Ionic y Angular Forms. Este componente nos permite ingresar los valores de los tres lados de un triángulo escaleno y calcular su perímetro. La función calcularPerimetro crea una instancia de TrianguloEscaleno con los valores ingresados y utiliza su método calcularPerimetro para determinar el perímetro, almacenando el resultado en la propiedad resultado.

Creamos una carpeta llamada páginas y también una nueva página llamada inicio por lo que el enrutamiento lo cambiamos borramos la página home.page.

```

└─ paginas \ inicio
   ├── inicio.page.html
   ├── inicio.page.scss
   ├── inicio.page.spec.ts
   └── TS inicio.page.ts

```

En este código nos permite seleccionar una figura geométrica—ya sea un círculo o un triángulo escaleno—desde un menú desplegable (ion-select). Dependiendo de la

selección que hagamos, el componente correspondiente (app-circulo o app-triangulo) se muestra utilizando la directiva *ngIf.

```
src > app > paginas > inicio > inicio.page.html > ion-content
Go to component
1 <ion-header [translucent]="true">
2   <ion-toolbar>
3     <ion-title>Calculo Figuras Geometricas</ion-title>
4   </ion-toolbar>
5 </ion-header>
6
7 <ion-content [fullscreen]="true">
8   <ion-select [(ngModel)]="Seleccion" label="Figuras Geometricas" placeholder="Seleccione una figuras geometrica">
9     <ion-select-option value="circulo">circulo</ion-select-option>
10    <ion-select-option value="triangulo">triangulo</ion-select-option>
11  </ion-select>
12
13  <app-circulo *ngIf="Seleccion == 'circulo'" />
14  <app-triangulo *ngIf="Seleccion == 'triangulo'" />
15
16
17
18 </ion-content>
19
```

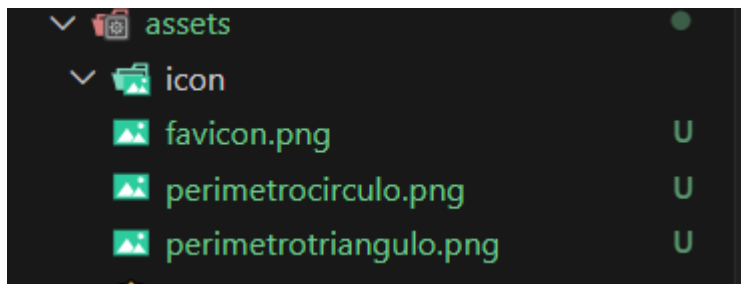
Aquí definimos el componente InicioPage, que nos sirve como la página de inicio de la aplicación. Nos permite seleccionar entre diferentes figuras geométricas, como un círculo o un triángulo escaleno, a través de un menú desplegable (ion-select). Dependiendo de la selección que hagamos, se muestra el componente correspondiente (CirculoComponent o TrianguloComponent) usando la directiva *ngIf.

```

src > app > paginas > inicio > TS inicio.page.ts > InicioPage
4 import { IonContent, IonHeader, IonTitle, IonToolbar, IonItem, IonSelect, IonSelectOption } from '@ionic/angular/standalone';
5 import { CirculoComponent } from 'src/app/componentes/circulo/circulo.component';
6 import { TrianguloComponent } from 'src/app/componentes/triangulo/triangulo.component';
7
8
9 @Component({
10   selector: 'app-inicio',
11   templateUrl: './inicio.page.html',
12   styleUrls: ['./inicio.page.scss'],
13   standalone: true,
14   imports: [IonSelect, IonSelectOption, IonItem, TrianguloComponent, CirculoComponent, IonContent, IonHeader, IonTitle, IonToolbar, C
15 })
16 export class InicioPage implements OnInit {
17
18   Seleccion:string = ""
19
20   constructor() { }
21
22   ngOnInit() {
23   }
24
25 }

```

importamos las imágenes referenciales para que queden visualizadas al momento de seleccionar una de las 2 figuras geométricas (circulo o triangulo).



Una vez realizado los códigos pertinentes para cada componente o página con ruta. Procedemos a iniciar la web (aunque uno fue probando antes), y se pudo realizar la selección de las figuras geométricas.

Calculo Figuras Geometricas

Figuras Geometricas

circulo ▲

$$P = \pi \cdot d$$

$$P = 2\pi \cdot r$$

$$A = \pi r^2$$

Donde:

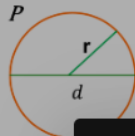
A = Área

P = Perímetro

d = diámetro

r = radio

$\pi \approx 3.1416$



Figuras Geometricas

radio 20

CALCULAR PERÍMETRO



circulo



triangulo

CANCEL

OK

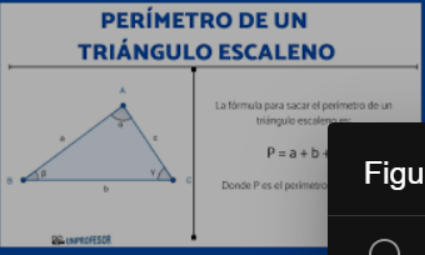
Calculo Figuras Geometricas

Figuras Geometricas

triangulo ▲

Triángulo Escaleno

PERÍMETRO DE UN TRIÁNGULO ESCALENO



La fórmula para sacar el perímetro de un triángulo escaleno es:
 $P = a + b + c$
Donde P es el perímetro

Figuras Geometricas

☐ circulo

☒ triangulo

CANCEL

OK

Lado A

Lado B

Lado C

CALCULAR PERÍMETRO

Como reflexión se puede decir falto darle más énfasis a la parte funcional de la página web ya que se generaron muchos errores que no me permitían dejar la página visible, pero se hará el mejor esfuerzo para superar.