**PARTE2-CREACIÓN DE LA APLICACIÓN DEL CLIMA**

**+) Importar librerías para llamadas http en el archivo app.module.ts:**

import { HttpClientModule } from '@angular/common/http';

@NgModule({

  declarations: [AppComponent],

  entryComponents: [],

  imports: [

    BrowserModule,

    IonicModule.forRoot(), AppRoutingModule,

    AngularFireModule.initializeApp(environment.firebaseConfig),

    AngularFireAuthModule,

    HttpClientModule],

  providers: [

    StatusBar,

    SplashScreen,

    { provide: RouteReuseStrategy, useClass: IonicRouteStrategy }

  ],

  bootstrap: [AppComponent]

})

export class AppModule {}

**+) Crear paginas para las ciudades de las que se va a mostrar el clima:**

Dentro de la carpeta del proyecto ionic abrir una terminal y ejecutar:

ionic generate page ciudades/quito

ionic generate page ciudades/londres

ionic generate page ciudades/paris

ionic generate page ciudades/madrid

ionic generate page ciudades/new york

**+) En archivo weather.page.html crear menú codificar los siguiente:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <ion-card-header>Lista de Ciudades</ion-card-header>

  <ion-list>

    <ion-list-header>

      Ciudades:

    </ion-list-header>

    <ion-item class="user-list">

      <ion-item lines="none" button routerLink='/quito' detail>

        <ion-label>

          <h5>

            <ion-icon name="globe"></ion-icon>QUITO

          </h5>

        </ion-label>

      </ion-item>

    </ion-item>

    <ion-item class="user-list">

      <ion-item lines="none" button routerLink="/londres" detail>

        <ion-label>

          <h5>

            <ion-icon name="globe"></ion-icon>LONDRES

          </h5>

        </ion-label>

      </ion-item>

    </ion-item>

    <ion-item class="user-list">

      <ion-item lines="none" button routerLink="/paris" detail>

        <ion-label>

          <h5>

            <ion-icon name="globe"></ion-icon>PARIS

          </h5>

        </ion-label>

      </ion-item>

    </ion-item>

    <ion-item class="user-list">

      <ion-item lines="none" button routerLink="/madrid" detail>

        <ion-label>

          <h5>

            <ion-icon name="globe"></ion-icon>MADRID

          </h5>

        </ion-label>

      </ion-item>

    </ion-item>

    <ion-item class="user-list">

      <ion-item lines="none" button routerLink="/newyork" detail>

        <ion-label>

          <h5>

            <ion-icon name="globe"></ion-icon>NEW YORK

          </h5>

        </ion-label>

      </ion-item>

    </ion-item>

  </ion-list>

</ion-content>

**+) Crear servicio para controlador de carga de los datos de la api:**

Dentro de la carpeta del proyecto ionic abrir terminal y ejecutar:

ionic g service serviceCommon/commonservice

**+) Abrir archivo common.service.ts, y codificar debe quedar asi:**

import { Injectable } from '@angular/core';

import { LoadingController } from '@ionic/angular';

@Injectable({

  providedIn: 'root'

})

export class CommonService {

  constructor(private loadingController: LoadingController) { }

  async loadingPresent(message: string = null, duration: number = null) {

    const loading = await this.loadingController.create({ message, duration });

    return await loading.present();

}

async loadingDismiss() {

    setTimeout(() => {

        return this.loadingController.dismiss();

    }, 1000);

}

}

**+) Codificar quito.page.ts:**

import { Component, OnInit } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { CommonService } from '../../servicesCommon/common.service';

@Component({

  selector: 'app-quito',

  templateUrl: './quito.page.html',

  styleUrls: ['./quito.page.scss'],

})

export class QuitoPage implements OnInit {

  currentWeather: any;

  weatherData: any;

  summaryIcon: string;

  weatherResult = false;

  loading: any;

  constructor(private http: HttpClient,

              private common: CommonService) {}

              ngOnInit() {

              }

  async ionViewWillEnter() {

    this.common.loadingPresent();

    this.http.get('https://cors-anywhere.herokuapp.com/https://api.darksky.net/forecast/0e12efde401f70a2ccb80be3d60cccb5/-0.225219,-78.5248')

     .subscribe((weatherData: any) => {

      this.common.loadingDismiss();

      this.weatherResult = true;

       this.weatherData = weatherData;

       this.currentWeather = weatherData.currently;

       if (this.currentWeather.summary.toLowerCase() === 'cloudy') {

         this.summaryIcon = 'cloudy';

       } else if (this.currentWeather.summary.toLowerCase() === 'rainy') {

         this.summaryIcon = 'rainy';

       } else if (this.currentWeather.summary.toLowerCase() === 'clear') {

         this.summaryIcon = 'sunny';

       } else if (this.currentWeather.summary.toLowerCase() === 'thunderstorm') {

         this.summaryIcon = 'thunderstorm';

       }

     });

  }

}

**+) En el archivo quito.page.html el código queda de la siguiente manera:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather QUITO</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <div \*ngIf="weatherResult" class="ion-padding">

    <ion-card>

      <ion-card-content>

        <ion-item class="activated" lines="none">

          <img src="../../../assets/img/quito.jpg">

          <ion-icon name="{{ summaryIcon }}" slot="start"></ion-icon>

        </ion-item>

        <ion-item-divider>

          <ion-card-subtitle>{{ weatherData.daily.summary }}°</ion-card-subtitle>

        </ion-item-divider>

        <ion-item-divider>

        </ion-item-divider>

        <br>

        <div class="currentInfo">

          <p>Zona: {{ weatherData.timezone }}</p><br>

          <p>El día esta a: {{ currentWeather.temperature }} grados</p><br>

          <p>Punto de rocío: {{ currentWeather.dewPoint }}</p><br>

          <p>Humedad: {{ currentWeather.humidity }}%</p><br>

          <p>Visibilidad: {{ currentWeather.visibility }} mi</p><br>

          <p>Presión: {{ currentWeather.pressure }} mb</p><br>

          <p>Daily: {{ weatherData.daily }}</p>

        </div>

      </ion-card-content>

    </ion-card>

  </div>

</ion-content>../

  <ion-fab vertical="start" horizontal="end" edge slot="fixed">

    <ion-fab-button routerLink="../weather" color="warning">

      <ion-icon name="book"></ion-icon>

    </ion-fab-button>

  </ion-fab>

**+) Codificar madrid.page.ts:**

import { Component, OnInit } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { CommonService } from '../../servicesCommon/common.service';

@Component({

  selector: 'app-madrid',

  templateUrl: './madrid.page.html',

  styleUrls: ['./madrid.page.scss'],

})

export class MadridPage implements OnInit {

  currentWeather: any;

  weatherData: any;

  summaryIcon: string;

  weatherResult = false;

  loading: any;

  constructor(private http: HttpClient,

              private common: CommonService) {}

              ngOnInit() {

              }

  async ionViewWillEnter() {

    this.common.loadingPresent();

    this.http.get('https://cors-anywhere.herokuapp.com/https://api.darksky.net/forecast/0e12efde401f70a2ccb80be3d60cccb5/40.4165001,-3.7025599')

     .subscribe((weatherData: any) => {

      this.common.loadingDismiss();

      this.weatherResult = true;

       this.weatherData = weatherData;

       this.currentWeather = weatherData.currently;

       if (this.currentWeather.summary.toLowerCase() === 'cloudy') {

         this.summaryIcon = 'cloudy';

       } else if (this.currentWeather.summary.toLowerCase() === 'rainy') {

         this.summaryIcon = 'rainy';

       } else if (this.currentWeather.summary.toLowerCase() === 'clear') {

         this.summaryIcon = 'sunny';

       } else if (this.currentWeather.summary.toLowerCase() === 'thunderstorm') {

         this.summaryIcon = 'thunderstorm';

       }

     });

  }

}

**+) En el archivo madrid.page.html el código queda de la siguiente manera:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather MADRID</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <div \*ngIf="weatherResult" class="ion-padding">

    <ion-card>

      <ion-card-content>

        <ion-item class="activated" lines="none">

          <ion-icon name="{{ summaryIcon }}" slot="start"></ion-icon>

        </ion-item>

        <ion-item-divider>

          <ion-card-subtitle>{{ weatherData.daily.summary }}°</ion-card-subtitle>

        </ion-item-divider>

        <ion-item-divider>

          <img src="../../../assets/img/madrid.jpg">

        </ion-item-divider>

        <br>

        <div class="currentInfo">

          <p>Zona: {{ weatherData.timezone }}</p><br>

          <p>El día esta a: {{ currentWeather.temperature }} grados</p><br>

          <p>Punto de rocío: {{ currentWeather.dewPoint }}</p><br>

          <p>Humedad: {{ currentWeather.humidity }}%</p><br>

          <p>Visibilidad: {{ currentWeather.visibility }} mi</p><br>

          <p>Presión: {{ currentWeather.pressure }} mb</p><br>

          <p>Daily: {{ weatherData.daily }}</p>

        </div>

      </ion-card-content>

    </ion-card>

  </div>

  <ion-fab vertical="start" horizontal="end" edge slot="fixed">

    <ion-fab-button routerLink="../weather" color="warning">

      <ion-icon name="book"></ion-icon>

    </ion-fab-button>

  </ion-fab>

</ion-content>

**+) Codificar londres.page.ts:**

import { Component, OnInit } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { CommonService } from '../../servicesCommon/common.service';

@Component({

  selector: 'app-londres',

  templateUrl: './londres.page.html',

  styleUrls: ['./londres.page.scss'],

})

export class LondresPage implements OnInit {

  currentWeather: any;

  weatherData: any;

  summaryIcon: string;

  weatherResult = false;

  loading: any;

  constructor(private http: HttpClient,

              private common: CommonService) {}

              ngOnInit() {

              }

  async ionViewWillEnter() {

    this.common.loadingPresent();

    this.http.get('https://cors-anywhere.herokuapp.com/https://api.darksky.net/forecast/0e12efde401f70a2ccb80be3d60cccb5/ 51.5072,-0.1275')

     .subscribe((weatherData: any) => {

      this.common.loadingDismiss();

      this.weatherResult = true;

       this.weatherData = weatherData;

       this.currentWeather = weatherData.currently;

       if (this.currentWeather.summary.toLowerCase() === 'cloudy') {

         this.summaryIcon = 'cloudy';

       } else if (this.currentWeather.summary.toLowerCase() === 'rainy') {

         this.summaryIcon = 'rainy';

       } else if (this.currentWeather.summary.toLowerCase() === 'clear') {

         this.summaryIcon = 'sunny';

       } else if (this.currentWeather.summary.toLowerCase() === 'thunderstorm') {

         this.summaryIcon = 'thunderstorm';

       }

     });

}

}

**+) En el archivo londres.page.html el código queda de la siguiente manera:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather LONDRES</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <div \*ngIf="weatherResult" class="ion-padding">

    <ion-card>

      <ion-card-content>

        <ion-item class="activated" lines="none">

          <ion-icon name="{{ summaryIcon }}" slot="start"></ion-icon>

        </ion-item>

        <ion-item-divider>

          <ion-card-subtitle>{{ weatherData.daily.summary }}°</ion-card-subtitle>

        </ion-item-divider>

        <ion-item-divider>

          <img src="../../../assets/img/londres.jpg">

        </ion-item-divider>

        <br>

        <div class="currentInfo">

          <p>Zona: {{ weatherData.timezone }}</p><br>

          <p>El día esta a: {{ currentWeather.temperature }} grados</p><br>

          <p>Punto de rocío: {{ currentWeather.dewPoint }}</p><br>

          <p>Humedad: {{ currentWeather.humidity }}%</p><br>

          <p>Visibilidad: {{ currentWeather.visibility }} mi</p><br>

          <p>Presión: {{ currentWeather.pressure }} mb</p><br>

          <p>Daily: {{ weatherData.daily }}</p>

        </div>

      </ion-card-content>

    </ion-card>

  </div>

  <ion-fab vertical="start" horizontal="end" edge slot="fixed">

    <ion-fab-button routerLink="../weather" color="warning">

      <ion-icon name="book"></ion-icon>

    </ion-fab-button>

  </ion-fab>

</ion-content>

**+) Codificar newyork.page.ts:**

import { Component, OnInit } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { CommonService } from '../../servicesCommon/common.service';

@Component({

  selector: 'app-newyork',

  templateUrl: './newyork.page.html',

  styleUrls: ['./newyork.page.scss'],

})

export class NewyorkPage implements OnInit {

  currentWeather: any;

  weatherData: any;

  summaryIcon: string;

  weatherResult = false;

  loading: any;

  constructor(private http: HttpClient,

              private common: CommonService) {}

              ngOnInit() {

              }

  async ionViewWillEnter() {

    this.common.loadingPresent();

    this.http.get('https://cors-anywhere.herokuapp.com/https://api.darksky.net/forecast/0e12efde401f70a2ccb80be3d60cccb5/40.6643,-73.9385')

     .subscribe((weatherData: any) => {

      this.common.loadingDismiss();

      this.weatherResult = true;

       this.weatherData = weatherData;

       this.currentWeather = weatherData.currently;

       if (this.currentWeather.summary.toLowerCase() === 'cloudy') {

         this.summaryIcon = 'cloudy';

       } else if (this.currentWeather.summary.toLowerCase() === 'rainy') {

         this.summaryIcon = 'rainy';

       } else if (this.currentWeather.summary.toLowerCase() === 'clear') {

         this.summaryIcon = 'sunny';

       } else if (this.currentWeather.summary.toLowerCase() === 'thunderstorm') {

         this.summaryIcon = 'thunderstorm';

       }

     });

  }

}

**+) En el archivo newyork.page.html el código queda de la siguiente manera:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather NEW YORK</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <div \*ngIf="weatherResult" class="ion-padding">

    <ion-card>

      <ion-card-content>

        <ion-item class="activated" lines="none">

          <ion-icon name="{{ summaryIcon }}" slot="start"></ion-icon>

        </ion-item>

        <ion-item-divider>

          <ion-card-subtitle>{{ weatherData.daily.summary }}°</ion-card-subtitle>

        </ion-item-divider>

        <ion-item-divider>

          <img src="../../../assets/img/newyork.jpg">

        </ion-item-divider>

        <br>

        <div class="currentInfo">

          <p>Zona: {{ weatherData.timezone }}</p><br>

          <p>El día esta a: {{ currentWeather.temperature }} grados</p><br>

          <p>Punto de rocío: {{ currentWeather.dewPoint }}</p><br>

          <p>Humedad: {{ currentWeather.humidity }}%</p><br>

          <p>Visibilidad: {{ currentWeather.visibility }} mi</p><br>

          <p>Presión: {{ currentWeather.pressure }} mb</p><br>

          <p>Daily: {{ weatherData.daily }}</p>

        </div>

      </ion-card-content>

    </ion-card>

  </div>

  <ion-fab vertical="start" horizontal="end" edge slot="fixed">

    <ion-fab-button routerLink="../weather" color="warning">

      <ion-icon name="book"></ion-icon>

    </ion-fab-button>

  </ion-fab>

</ion-content>

**+) Codificar paris.page.ts:**

import { Component, OnInit } from '@angular/core';

import { HttpClient } from '@angular/common/http';

import { CommonService } from '../../servicesCommon/common.service';

@Component({

  selector: 'app-paris',

  templateUrl: './paris.page.html',

  styleUrls: ['./paris.page.scss'],

})

export class ParisPage implements OnInit {

  currentWeather: any;

  weatherData: any;

  summaryIcon: string;

  weatherResult = false;

  loading: any;

  constructor(private http: HttpClient,

              private common: CommonService) {}

              ngOnInit() {

              }

  async ionViewWillEnter() {

    this.common.loadingPresent();

    this.http.get('https://cors-anywhere.herokuapp.com/https://api.darksky.net/forecast/0e12efde401f70a2ccb80be3d60cccb5/48.8534088,2.3487999')

     .subscribe((weatherData: any) => {

      this.common.loadingDismiss();

      this.weatherResult = true;

       this.weatherData = weatherData;

       this.currentWeather = weatherData.currently;

       if (this.currentWeather.summary.toLowerCase() === 'cloudy') {

         this.summaryIcon = 'cloudy';

       } else if (this.currentWeather.summary.toLowerCase() === 'rainy') {

         this.summaryIcon = 'rainy';

       } else if (this.currentWeather.summary.toLowerCase() === 'clear') {

         this.summaryIcon = 'sunny';

       } else if (this.currentWeather.summary.toLowerCase() === 'thunderstorm') {

         this.summaryIcon = 'thunderstorm';

       }

     });

  }

}

**+) En el archivo paris.page.html el código queda de la siguiente manera:**

<ion-header>

  <ion-toolbar color="secondary">

    <ion-title>Weather PARIS</ion-title>

  </ion-toolbar>

</ion-header>

<ion-content>

  <div \*ngIf="weatherResult" class="ion-padding">

    <ion-card>

      <ion-card-content>

        <ion-item class="activated" lines="none">

          <ion-icon name="{{ summaryIcon }}" slot="start"></ion-icon>

        </ion-item>

        <ion-item-divider>

          <ion-card-subtitle>{{ weatherData.daily.summary }}°</ion-card-subtitle>

        </ion-item-divider>

        <ion-item-divider>

          <img src="../../../assets/img/paris.jpg">

        </ion-item-divider>

        <br>

        <div class="currentInfo">

          <p>Zona: {{ weatherData.timezone }}</p><br>

          <p>El día esta a: {{ currentWeather.temperature }} grados</p><br>

          <p>Punto de rocío: {{ currentWeather.dewPoint }}</p><br>

          <p>Humedad: {{ currentWeather.humidity }}%</p><br>

          <p>Visibilidad: {{ currentWeather.visibility }} mi</p><br>

          <p>Presión: {{ currentWeather.pressure }} mb</p><br>

          <p>Daily: {{ weatherData.daily }}</p>

        </div>

      </ion-card-content>

    </ion-card>

  </div>

  <ion-fab vertical="start" horizontal="end" edge slot="fixed">

    <ion-fab-button routerLink="../weather" color="warning">

      <ion-icon name="book"></ion-icon>

    </ion-fab-button>

  </ion-fab>

</ion-content>

**Estilos**

**+) En el archivo home.page.scss el código queda de la siguiente manera:**

ion-content {

    --background: url('../../assets/img/fondo\_login.jpg') no-repeat;

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

**+) En el archivo register.page.scss el código queda de la siguiente manera:**

ion-content {

    --background: url('../../assets/img/fondo\_register.png') no-repeat 10% 10%;

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

**+) En el archivo weather.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

**+) En el archivo quito.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

ion-card {

    --background: #ffffff;

    --color: #3880ff;

}

**+) En el archivo paris.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

ion-card {

    --background: #ffffff;

    --color: #3880ff;

}

**+) En el archivo londres.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

ion-card {

    --background: #ffffff;

    --color: #3880ff;

}

**+) En el archivo madrid.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

ion-card {

    --background: #ffffff;

    --color: #3880ff;

}

**+) En el archivo newyork.page.scss el código queda de la siguiente manera:**

ion-content {

    --background:url('../../../assets/img/fondo\_listaCiudades.jpg');

    --color: #3880ff;

    --keyboard-offset: 0px;

    --offset-bottom: 0px;

    --offset-top: 0px;

    --padding-bottom: 0px;

    --padding-end: 0px;

    --padding-start: 0px;

    --padding-top: 0px;

}

ion-card {

    --background: #ffffff;

    --color: #3880ff;

}

**+) Finalmente ejecutamos el proyecto ionic:**

**Dentro de la carpeta del proyecto abrir un terminal y ejecutar el comando: ionic serve**