>Ionic start navegación blank

>ionic g page principal

Creamos carpete pagina3 con ficheros pagina3.html y página3.ts

Pagina3.html

<ion-header>

<ion-navbar>

<ion-title>

Página 3

</ion-title>

</ion-navbar>

</ion-header>

<ion-content padding>

</ion-content>

Pagina3.ts

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

@Component({

selector: 'page-pagina3',

templateUrl: 'pagina3.html'

})

export class Pagina3Page {

constructor() {

}

}

Genero automáticamente ajustes, ajustes2, modal y tabs

En app.module.ts import y

PrincipalPage,

Principal2Page,

Pagina3Page,

AjustesPage,

Ajustes2Page,

ModalPage,

TabsPage,

En @NgModule y en entryComponents

En app.components.ts cambiamos la página de inicio

rootPage:any = PrincipalPage;

Borramos página home

\*\*\*\*\*\*\*\*\*\*\*\*\*\*COMMIT Con todas las paginas \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

En principal.ts el método

NavegarPagina(){

this.navCtrl.push(Principal2Page);

}

Y en principal.html

<button ion-button

block

color ="primary"

(click)="NavegarPagina()">

Ir a página 2

</button>

Ahora solo con html

Necesitamos tener en el principal.ts

pagina2:any = Principal2Page;

y en el principal.html

[navPush]="pagina2">

Pasando parámetros de página Principal2 a Página3. En principal2

irPagina3(p:any){

console.log(p);

this.navCtrl.push(Pagina3Page, {"pais":p})

}

<ion-list>

<ion-item \*ngFor="let p of paises"

(click)="irPagina3(p)" >

{{p.pais}}

</ion-item>

</ion-list>

En página3

pais: any = {};

constructor(private navParams:NavParams) {

console.log(this.navParams);

this.pais = this.navParams.get("pais");

}

<ion-header>

<ion-navbar>

<ion-title>

{{pais.pais}}

</ion-title>

</ion-navbar>

</ion-header>

<ion-content padding>

<h3 text-center>{{pais.capital}}</h3>

</ion-content>

Para ir a la principal (root)

IrRoot(){

this.navCtrl.popToRoot();

}

Por html la etiqueta

navPop>

\*\*\*\*\*\*\*\*\*\*\*\*\*commit NavController y NavParams

TABS

tabs.ts

import { AjustesPage } from './../ajustes/ajustes';

import { PrincipalPage } from './../principal/principal';

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

import { IonicPage} from 'ionic-angular';

@IonicPage()

@Component({

selector: 'page-tabs',

templateUrl: 'tabs.html',

})

export class TabsPage {

tab1:any;

tab2:any;

constructor() {

this.tab1 = PrincipalPage;

this.tab2 = AjustesPage;

}

Para tabs.htm copio de DOCU COMPONENTS TABS y reorganizo

<ion-tabs>

<ion-tab tabIcon="alarm" tabTitle="Principal" [root]="tab1"></ion-tab>

<ion-tab tabIcon="hammer" tabTitle="Ajustes" [root]="tab2"></ion-tab>

</ion-tabs>

En app.components.ts variamos la página de inicio

rootPage:any = TabsPage;

la siguiente propiedad marca la tab en uso

selectedIndex= "2"

Para ir por código a otro tab, podemos poner un botón y asociar el evento clic por ejemplo a

activarPrincipal(){

this.navCtrl.parent.select(0);

}

PAGINA MODAL

Creamos un botón en ajustes.html y programos evento click en mostrarModal

(inyectar ModalController) y el método

let modal = this.modalCtrl.create(ModalPage, {nombre:"Dolores", edad:24})

modal.present();

En el constructor de modal.ts recuperamos valores con

this.nombre\_mio=this.navParams.get("nombre");

this.edad\_mia= this.navParams.get("edad");

Y los mostramos en el html con

<h1>{{nombre\_mio}} {{edad\_mia}}</h1>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*COMMIT Modal con parámetros de ida \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Cerrar sin parámetros

cerrar\_sin\_parametros(){

this.viewCtrl.dismiss();

}

Cerrar con parámetros

cerrar\_con\_parametros(){

let dato = {

nombre:"Lara",

edad: 4,

ubucicion:{

longitud: 10,

latitud: 43.444

}

}

this.viewCtrl.dismiss(dato);

}

Puedo acceder a los parámetros de vuelta desde la página invocante (Ajustes) a través de

modal.onDidDismiss( (parametros)=>{

console.log ("Datos de vuelta");

console.log (parametros);

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*COMMIT modal con parametros vuelat\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

MENUS LATERALES

En app.html copio lo que tiene la DOCU COMPONENTS TABS, pero dejo la última ninea

<ion-menu [content]="content">

<ion-header>

<ion-toolbar>

<ion-title>Menu</ion-title>

</ion-toolbar>

</ion-header>

<ion-content>

<ion-list>

<button ion-item (click)="abrirPagina(tabs)">

Tabs

</button>

<button ion-item (click)="abrirPagina(ajustes2)">

Ajustes2

</button>

</ion-list>

</ion-content>

</ion-menu>

<ion-nav [root]="rootPage" #content></ion-nav>

Enlazo la última línea con un #content con el encabezado del ion-menu.

Y establezco un método para atender el evento click y los parámetros

En app.components.ts creo el método y las dos propiedades de página

tabs=TabsPage;

ajustes2 = Ajustes2Page;

abrirPagina(pagina:any){

this.rootPage = pagina;

}

Creo el botón de menú en la parte superior izquierda de principal (con evento) y ajustes (sin evento)

\*\*\*\*\*\*\*\*\*\*\*\*\*\*COMMIT final \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*