

```

246 // Validacion de Periodos
247 this.validarPeriodosMeta('DISCRETO ACUMULADO');
248 } else if (this.registerForm.value.comportamiento == 'DISCRETO POR PERIODO') {
249 this.validarPeriodosMeta('DISCRETO POR PERIODO');
250 } else if (this.registerForm.value.comportamiento == 'CONTINUO') {
251 this.validarPeriodosMeta('CONTINUO');
252 } else {
253 // No a selecciona comportamiento
254 Swal.fire(
255 'Revisar Formulario',
256 'Seleccione el Comportamiento del Indicador',
257 'warning'
258 );
259 }
260 }
261
262 validarPeriodosMeta(comportamiento: string) {
263 // variable se suma totad los periodos
264 let total = 0;
265 let habilitarPeriodo = false;
266 // Validacion de Periodos Mensual, trimestral...
267 switch (this.opc) {
268 case 'MENSUAL':
269 if (comportamiento == 'DISCRETO ACUMULADO') {
270 total =
271 +this.registerForm.value.enero +
272 +this.registerForm.value.febrero +
273 +this.registerForm.value.marzo +
274 +this.registerForm.value.abril +
275 +this.registerForm.value.mayo +
276 +this.registerForm.value.junio +
277 +this.registerForm.value.julio +
278 +this.registerForm.value.agosto +
279 +this.registerForm.value.septiembre +
280 +this.registerForm.value.octubre +
281 +this.registerForm.value.noviembre +
282 +this.registerForm.value.diciembre;
283 } else if (comportamiento == 'DISCRETO POR PERIODO') {

```

MANUAL DE USUARIO

Proyecto empresarial Indicadores

Grupo empresarial relacionado a documentación

23/10/2021

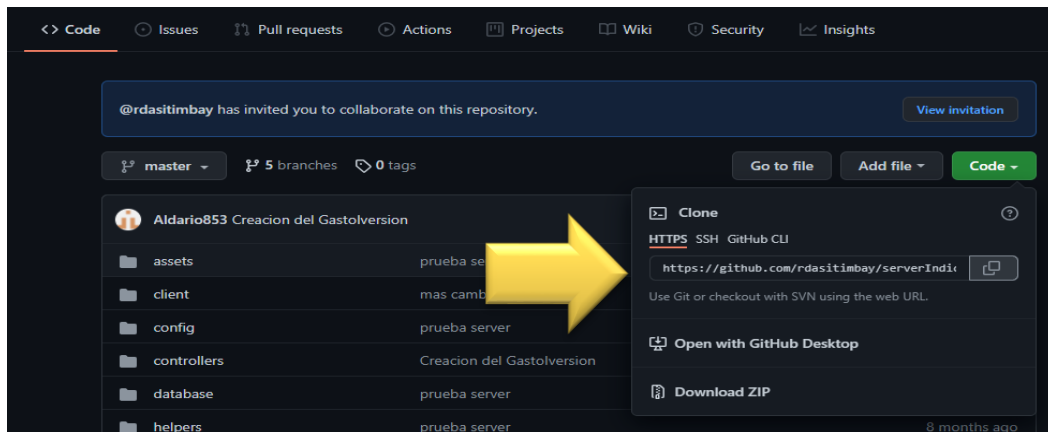


TABLA DE CONTENIDO

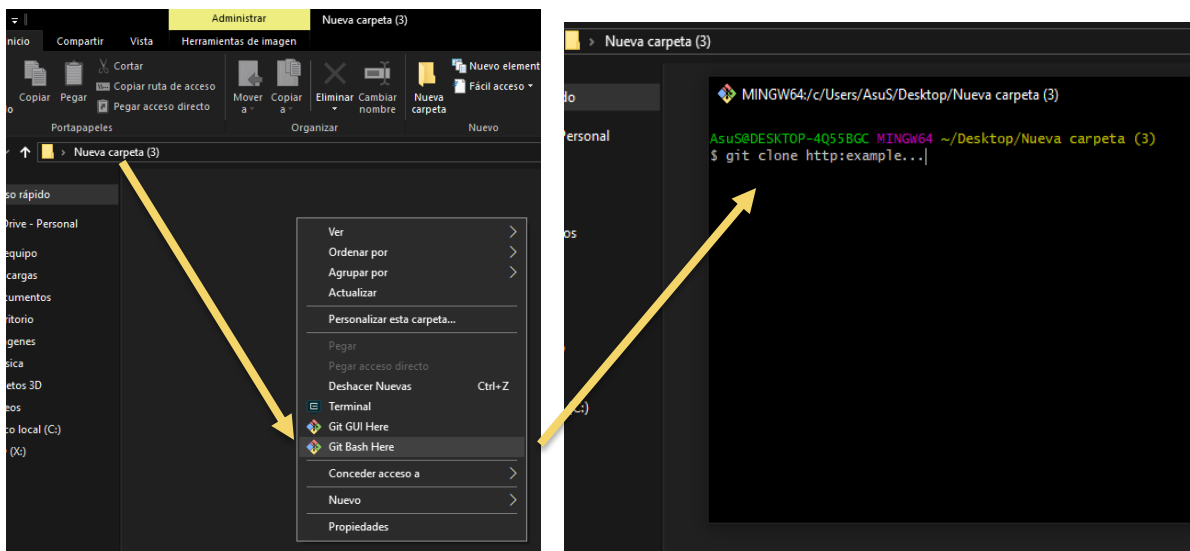
Desarrollo técnico	pag.3
Objetivo general a desarrollar	pag.5
Desarrollo del problema (Fronnd End)	pag.6
Declaración de la función en el servidor (Fronnd End)	pag.8
Desarrollo del problema (Back End)	pag.10
Modelo	pag.12
Parte visual	pag.15

DESARROLLO TECNICO

1. Primero clonaremos el proyecto de un repositorio en GitHub.



2. Crearemos una carpeta donde ira nuestro proyecto clonado, aremos clic derecho y abriremos una consola en la carpeta, dentro de la consola ejecutaremos el comando `git clone` junto con el url del proyecto que queramos clonar.



3. Una vez clonado el proyecto ejecutaremos en consola los siguientes comandos que instalaran y levantarán el proyecto.
 - 1.- Ejecutaremos el comando `npm install` para la instalación de los `node_modules`.

```
Asu@DESKTOP-4Q55BGC MINGW64 ~/Desktop/Nueva carpeta (3)
$ npm install
```

- 2.- Ejecutaremos el comando `npm run start` para iniciar el programa.

```
Asu@DESKTOP-4Q55BGC MINGW64 ~/Desktop/Nueva carpeta (3)
$ npm run start
```

- 3.- Ingresamos el comando `code .` para levantar el código en VSC.

```
Asu@DESKTOP-4Q55BGC MINGW64 ~/Desktop/Nueva carpeta (3)
$ code .
```

4. Una vez levantado el proyecto en VSC entraremos a la consola del editor y nos podemos percatar de que estamos trabajando en la rama **master** para entrar a una rama en la que podamos trabajar de forma más tranquila.

1.- Ejecutaremos el comando **git branch indicadoresAvance** para verificar en la rama que estamos trabajando:

```
AsuS@DESKTOP-4Q55BGC MINGW64 ~/Desktop/proyecto empresarial clonado/seguimientoIndicadores (master)
$ git branch
ModuleIndicadores
* master
```

2.-Comprobando que la rama sea la master ejecutaremos el comando **git checkout indicadoresAvance** para cambiarnos de rama a la más modificable:

```
AsuS@DESKTOP-4Q55BGC MINGW64 ~/Desktop/proyecto empresarial clonado/seguimientoIndicadores (master)
$ git checkout ModuleIndicadores
Switched to branch 'ModuleIndicadores'
M    package-lock.json
```

3.- Una vez cambiados de rama ejecutaremos el comando **git branch** para verificar que nos hemos cambiado de rama:

```
AsuS@DESKTOP-4Q55BGC MINGW64 ~/Desktop/proyecto empresarial clonado/seguimientoIndicadores (ModuleIndica
dores)
$ git branch
* ModuleIndicadores
master
```

4.- De esta forma podemos trabajar en una rama donde los errores se pueden corregir sin dañar el código principal.

5. Podemos ejecutar distintos comandos para exportar o importar código en la fase practica como pueden ser:

- Comando **git pull origin indicadoresAvance** para importar código que nuestro equipo de trabajo a realizado.

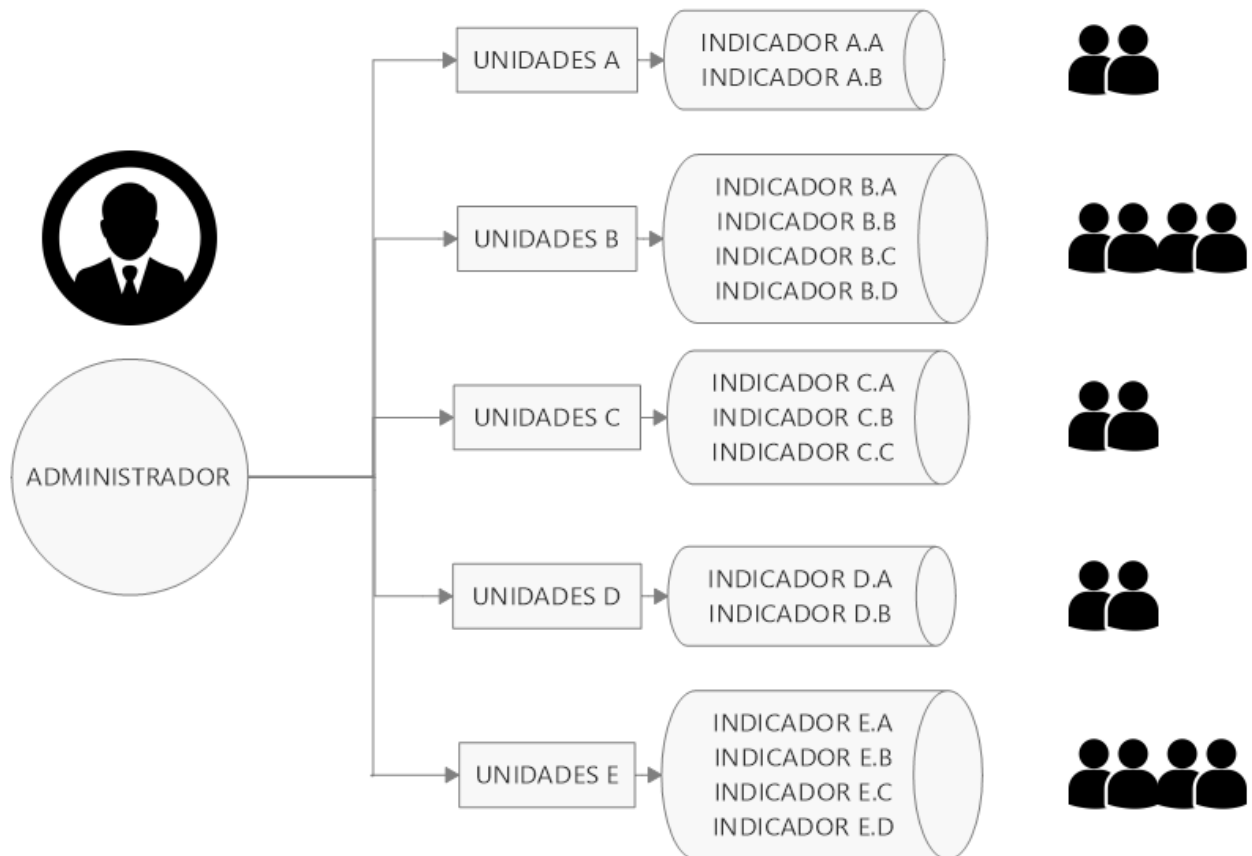
```
AsuS@DESKTOP-4Q55BGC MINGW64 ~/Desktop/proyecto empresarial clonado/seguimientoIndicadores (ModuleIndica
dores)
$ git pull origin ModuleIndicadores
From https://github.com/rdasitimbay/seguimientoIndicadores
* branch      ModuleIndicadores -> FETCH_HEAD
Updating 2522d8a..57d4bd1
Fast-forward
 src/app/models/departamento.ts | 2 +-
 1 file changed, 1 insertion(+), 1 deletion(-)
```











- Comando **git push** para exportar nuestros cambios al equipo de trabajo.

```
AsuS@DESKTOP-4Q55BGC MINGW64 ~/Desktop/proyecto empresarial/seguimientoIndicadores (ModuleIndicadores)
$ git push -u origin ModuleIndicadores
To https://github.com/rdasitimbay/seguimientoIndicadores.git
```

OBJETIVO GENERAL A DESARROLLAR

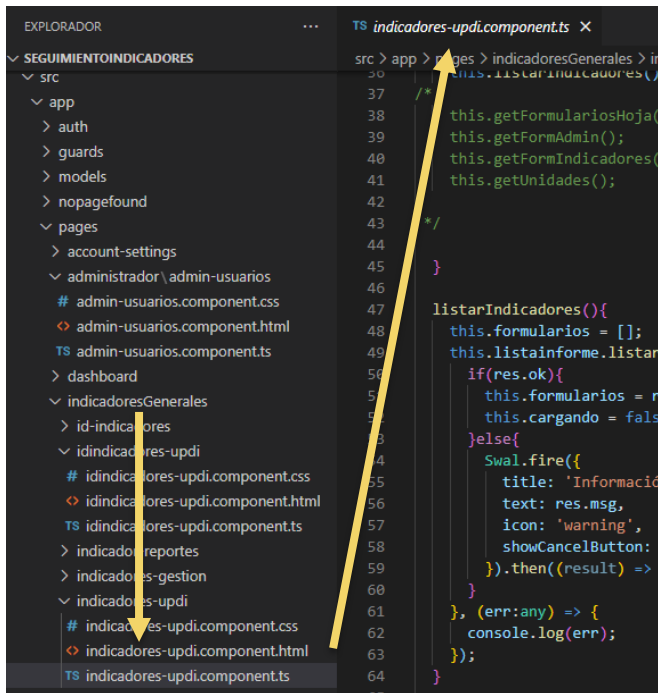
Desarrollar un filtro que permita que cada unidad solo visualice los indicadores que corresponden a dicha unidad, teniendo en claro que solo el administrador pueda visualizar todas las unidades existentes.



<input type="checkbox"/> Todo	UNIDAD	RESPONDE A	MACROPROCESO	FECHA CREACIÓN
<input type="checkbox"/> 1	PACAL	GESTIÓN DE ASESORÍA JURÍDICA	26-05-2021	 
<input type="checkbox"/> 2	PACAL	GESTIÓN DE DIRECCIÓN ESTRATÉGICA	26-05-2021	 
<input type="checkbox"/> 3	PACAL	GESTIÓN DE ASESORÍA JURÍDICA	26-05-2021	 
<input type="checkbox"/> 4	PACAL	GESTIÓN DE COORDINACIÓN ACADÉMICA GENERAL	26-05-2021	 
<input type="checkbox"/> 5	PACAL	GESTIÓN DE ASESORÍA JURÍDICA	26-05-2021	 

DESARROLLO DEL PROBLEMA (FRONT END)

Para desarrollar este filtro primero vamos a trabajar en la herramienta Angular destinada a Frond End y nos ubicaremos en: `indicadores-updi-component.ts`.



En `indicadores-updi-component.ts` vamos a declarar una función llamada: `listarIndicadores`.

```
listarIndicadores(){
  this.formularios = [];
  this.listainforme.listarIndicadoresRol(this.role).subscribe((res: any) => {
    if(res.ok){
      this.formularios = res.datos;
      this.cargando = false;
    }else{
      Swal.fire({
        title: 'Información',
        text: res.msg,
        icon: 'warning',
        showCancelButton: false,
      }).then((result) => {});
    }
  }, (err:any) => {
    console.log(err);
  });
}
```

En esta función vamos a llamar al servicio de los indicadores con **listainforme** y a consumir el servicio con **listarIndicadoresRol** enviando el **role**.

```
listarIndicadores(){
  this.formularios = [];
  this.listainforme.listarIndicadoresRol(this.role).subscribe((res: any) => {
```



Una vez especificada la función el **if** se encargará de evaluar las condiciones, si tal petición cumple las características el **if** apertura el **res.ok** que en este caso la variable de formularios poblara con los datos de respuesta (**res.datos**).

La variable **this.cargando = false**; permitirá ver el estado si está cargando o no.

De no ser correcto lo solicitado el **if** apertura la condición **false** que enviará una alerta con el error.

```
listarIndicadores(){
  this.formularios = [];
  this.listainforme.listarIndicadoresRol(this.role).subscribe((res: any) => {
    if(res.ok){
      this.formularios = res.datos;
      this.cargando = false;
    }else{
      Swal.fire({
        title: 'Información',
        text: res.msg,
        icon: 'warning',
        showCancelButton: false,
      }).then((result) => {});
    }
  }, (err:any) => {
    console.log(err);
  });
}
```

True

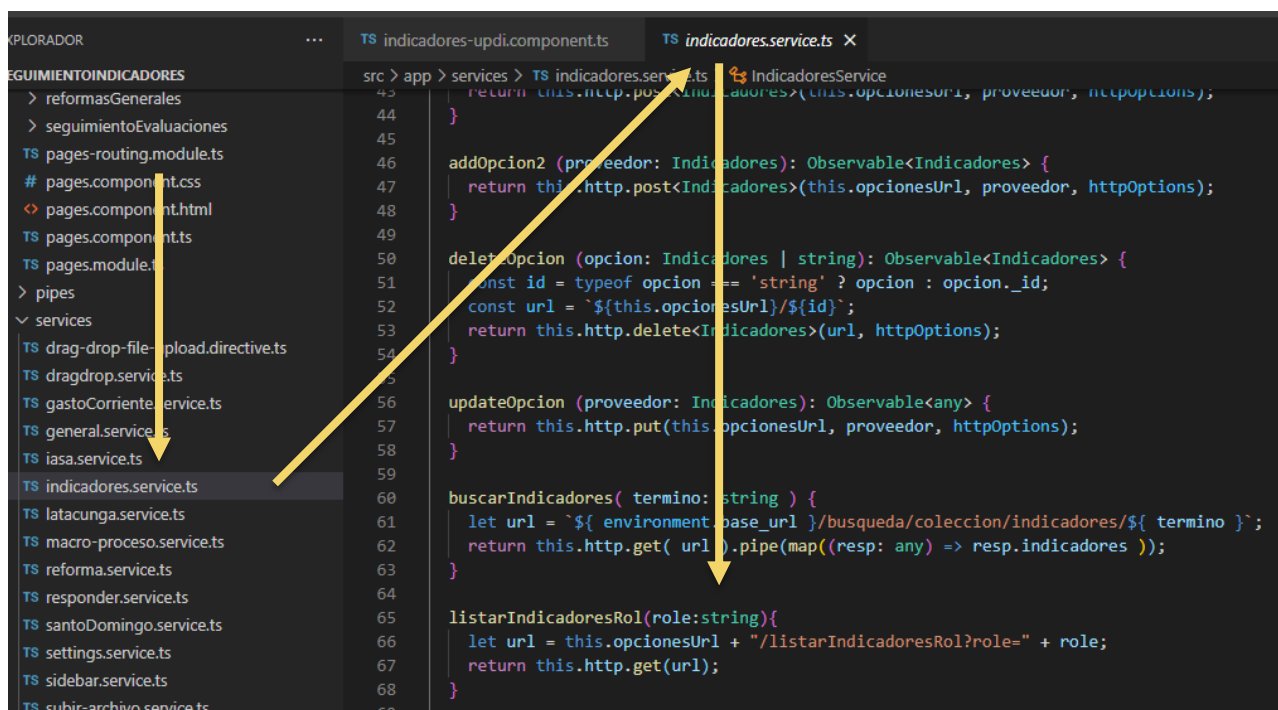
False

También declararemos un **ngOnInit** para llamar a la función cuando el servicio este encendido de la siguiente manera:

```
ngOnInit() {
    this.listarIndicadores();
    /*
    this.getFormulariosHoja();
    this.getFormAdmin();
    this.getFormIndicadores();
    this.getUnidades();
    */
}
```

Declaración de la función en el servidor (Front End)

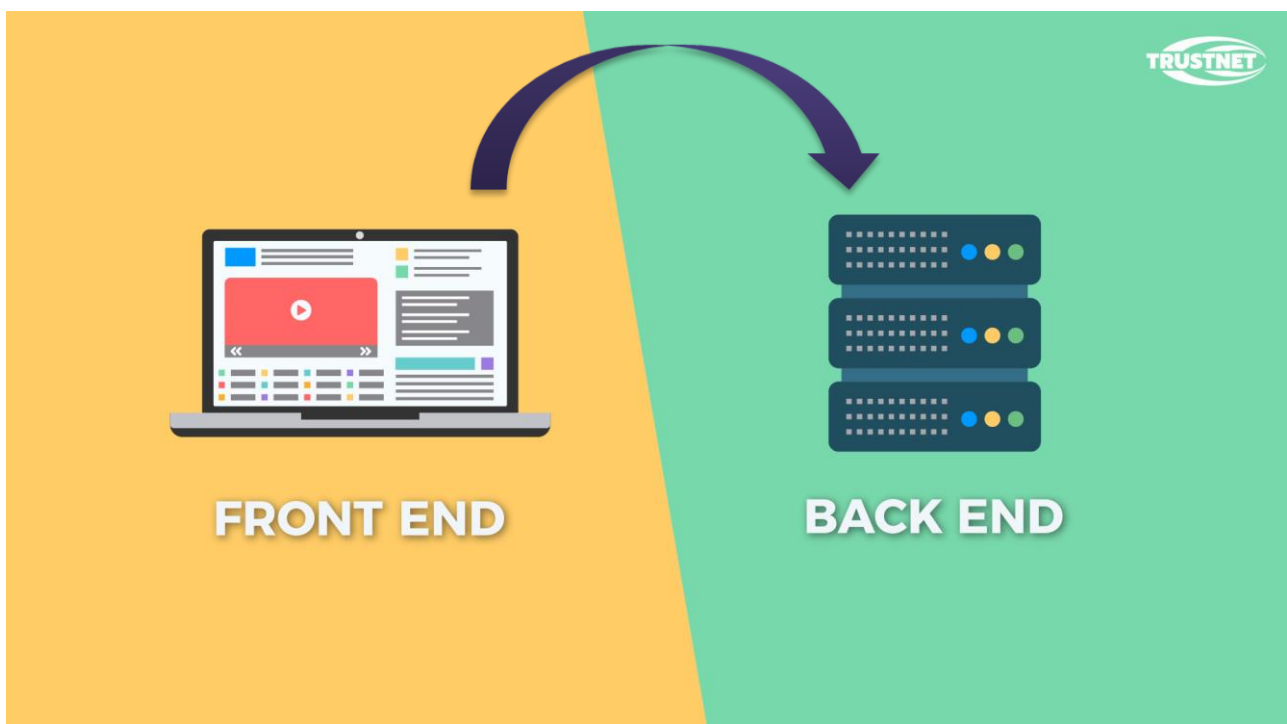
Una vez hallamos declarado la función en **indicadores-updi-component.ts** nos dirigimos al servidor de indicadores llamado **indicadores.service.ts** para llamar a la función destinada en **Back End**.



En `indicadores.service.ts` añadimos un nuevo servicio llamado `listarIndicadoresRol` para solicitar un rol (`role:string`), en el `Url` declaramos consumir el servicio `"/listarIndicadoresRol?role="` enviándome como parámetro el rol (`role`) desde el **Back End**.

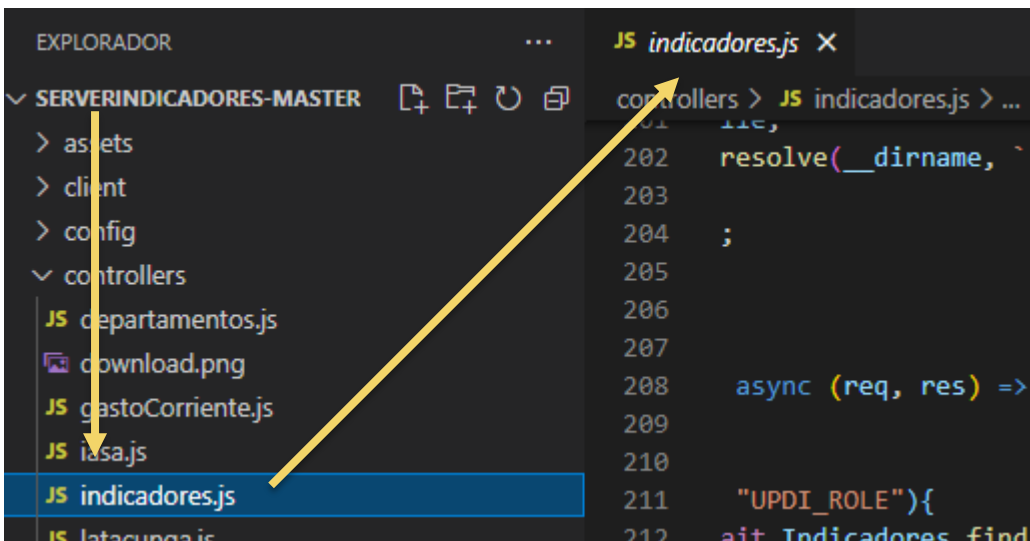
```
listarIndicadoresRol(role:string){  
  let url = this.opcionesUrl + "/listarIndicadoresRol?role=" + role;  
  return this.http.get(url);  
}
```

En las siguientes paginas se mostrará el desarrollo del **Back End**.



DESARROLLO DEL PROBLEMA (FRONT END)

Primero nos dirigimos a los controladores del **Back End** donde empezaremos a programar.



En **indicadores.js** declaramos una nueva función llamada **listarIndicadoresRol** donde va hacer de tipo asincrónico (**async**) y tendrá un **req** para enviar una solicitud y un **res** para devolver una respuesta.

Declaramos el método **try** seguido declaramos la variable "**var indicadores;**" para retornar el listado de indicadores.

El **catch** actuará si el **try** obtiene un error.

```
const listarIndicadoresRol = async (req, res) => {
  try {
    var indicadores;
    if(req.query.role == "UPDI_ROLE"){
      indicadores = await Indicadores.find({}).exec();
    }else{
      indicadores = await Indicadores.find({ role: req.query.role }).exec();
    }
    res.status(200).send({ ok: true, cantidad: indicadores.length ,datos: indicadores });
  } catch (error) {
    res.status(500).send({ ok: false, msg: "Error inesperado \n" + error.toString() });
  }
};
```

Con el **if** hacemos una comparación de que el parámetro **role** va hacer igual a “UPDI_ROLE” ya que este rol es el único que puede observar todas las unidades correspondientes.

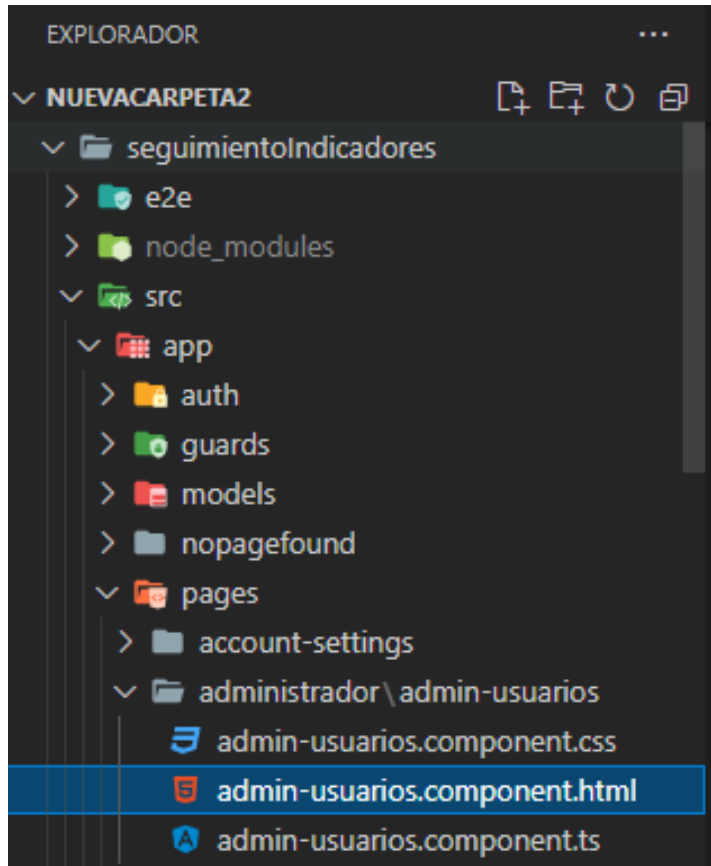
El **if** actuará si la comparación del **rol** si es **true** procederá hacer una consulta y traer a todos los roles existentes, pero si la comparación es **false** el **if** enviara a los indicadores filtrados por rol (**role**) y a los indicadores enviados por consulta.

```
if(req.query.role == "UPDI_ROLE"){  
  indicadores = await Indicadores.find({}).exec();  
}else{  
  indicadores = await Indicadores.find({ role: req.query.role }).exec();  
}
```

MODELO

Nos dirigimos al Frontend y abrimos

`seguimientoIndicadores\src\app\pages\administrador\admin-usuarios\` y en `admin-usuarios.component.html` trabajaremos la parte de agregar opciones al administrador.

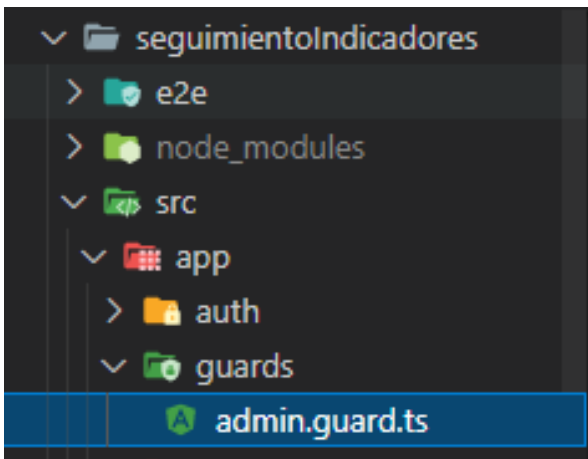


Los roles fueron creados de acuerdo al nombre de las unidades porque de estas se hará una relación.

```
<!--indicadores roles-->
<option value="FINANCIERO_ROLE">FINANCIERO_ROLE </option>
<option value="PLANIFICACION_ROLE">PLANIFICACION_ADMINISTRACION_ROLE </option>
<option value="ESPE_LATACUNGA_ROLE">ESPE_LATACUNGA_ROLE</option>
<option value="ESPE_STO_DOMINGO_ROLE">ESPE_SANTO_DOMINGO_ROLE</option>
<option value="ADMISION_REGISTRO_ROLE">ADMISION_REGISTRO_ROLE</option>
<option value="VICERECTOR_INVESTIGACION_INNOVACION_ROLE">VICERECTOR_INVESTIGACION_INNOVACION_ROLE</option>
<option value="GESTION_INVESTIGACION_ROLE">GESTION_INVESTIGACION_ROLE</option>
<option value="TECNOLOGIAS_INFORMACION_COMUNICACIONES_ROLE">TECNOLOGIAS_INFORMACION_COMUNICACIONES_ROLE</option>
<option value="TECNOLOGIAS_DE_LA_INFORMACION_Y_COMUNICACIONES_ROLE">TECNOLOGÍAS_DE_LA_INFORMACIÓN_COMUNICACIONES_ROLE</opt
<option value="COMUNICACION_SOCIAL_ROLE">COMUNICACIÓN_SOCIAL_ROLE</option>
<option value="DEPARTAMENTO_DE_Ciencias_DE_LA_TIERRA_Y_LA_CONSTRUCCION_CARRERAS_ROLE">DEPARTAMENTO_DE_Ciencias_DE_LA_TIERR
<option value="DEPARTAMENTO_DE_ELECTRICA_Y_ELECTRONICA_CARRERAS_ROLE">DEPARTAMENTO_DE_ELECTRICA_Y_ELECTRONICA_CARRERAS_ROL
<option value="DEPARTAMENTO_DE_Ciencias_DE_LA_VIDA_Y_AGRICULTURA_CARRERAS_ROLE">DEPARTAMENTO_DE_Ciencias_DE_LA_VIDA_Y_AGRIC
<option value="DEPARTAMENTO_DE_Ciencias_DE_LA_VIDA_Y_LA_AGRICULTURA_ROLE">DEPARTAMENTO_DE_Ciencias_DE_LA_VIDA_Y_LA_AGRICUL
<option value="DEPARTAMENTO_DE_Ciencias_ECONOMICAS_ADMINISTRATIVAS_Y_DE_COMERCIO_ROLE">DEPARTAMENTO_DE_Ciencias_ECONOMICAS
<option value="DEPARTAMENTO_DE_Ciencias_ECONOMICAS_ADMINISTRATIVAS_Y_DE_COMERCIO_CARRERAS_ROLE">DEPARTAMENTO_DE_Ciencias_E
```

Después nos dirigimos a la ruta de

`seguimientoIndicadores\src\app\guards\admin.guard.ts`



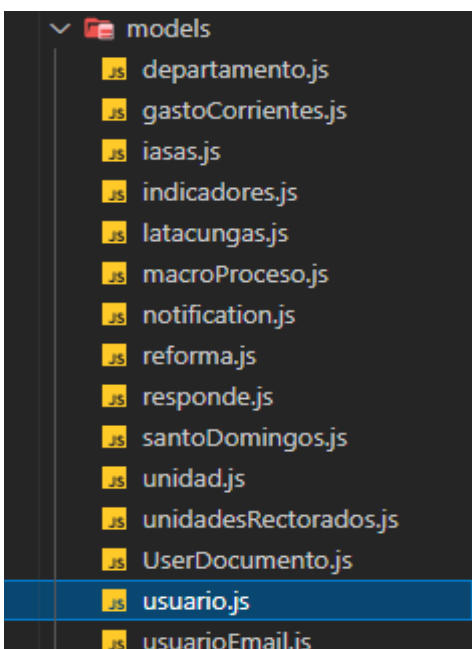
En esta parte añadiremos los roles porque de no hacerlo los **guards** no nos permitirán que el rol que se añadió tenga acceso a su ruto correspondiente.

```
if (this._usuarioService.usuario.role === 'VICERECTOR_ROLE') {
  return true;
}

if (this._usuarioService.usuario.role === 'DIRECTOR_ROLE') {
  return true;
}

if (this._usuarioService.usuario.role === 'PLANIFICADOR_ROLE') {
  return true;
}
```

Ahora abrimos el **Backend** y para que los roles sean válidos se los añadió en la siguiente ruta `serverIndicadores\models\usuario.js`



Una vez aquí se los añadió en la variable que ya estaba creada llamada **rolesValidos**, tomando en cuenta que sea el mismo nombre de las opciones creadas en **admin-usuarios.component.html**

```
var rolesValidos = {
  values: ['ADMIN_ROLE', 'USER_ROLE', 'RECTOR_ROLE', 'VICERECTOR_ROLE', 'DIRECTOR_ROLE', 'PLANIFICADOR_ROLE',
    'UPDI_ROLE', 'INACTIVO', 'REQUIRENTE_ROLE', 'PRESUPUESTO_ROLE', 'DIRECTOR_UPDI_ROLE', 'VICERECTOR_ADMINISTRATIVO_ROLE', 'VICERECTOR_ACADEMICO_GENERAL_ROLE',
    //indicadores roles
    'PLANIFICACION_ROLE', 'FINANCIERO_ROLE', 'ESPE_LATACUNGA_ROLE', 'ESPE_STO_DOMINGO_ROLE', 'ADMISION_REGISTRO_ROLE', 'VICERECTOR_INVESTIGACION_INNOVACION_ROLE',
    'INSTITUTO_DE_IDIOMAS_ROLE', 'CICTE_ROLE', 'TALENTO_HUMANO_ROLE', 'SECRETARIA_GENERAL_ROLE', 'IASA_1_ROLE', 'ASESORIA_JURIDICA_ROLE', 'CENTRO_DE_NANOCIENCIA_ROLE',
    'VICERRECTORADO_ACADEMICO_GENERAL_ROLE', 'DEPARTAMENTO_DE_Ciencias_MEDICAS_ROLE', 'VICERRECTORADO_DE_DOCENCIA_ROLE',
    'DEPARTAMENTO_DE_Ciencias_HUMANAS_Y_SOCIALES_ROLE', 'DEPARTAMENTO_DE_Ciencias_DE_LA_VIDA_Y_DE_LA_AGRICULTURA_ROLE', 'DEPARTAMENTO_DE_Ciencias_DE_LA_COMUNICACION_ROLE',
    'DEPARTAMENTO_DE_Ciencias_ECONOMICAS_ADMINISTRATIVAS_DE_COMERCIO_ROLE',
    'DEPARTAMENTO_DE_Ciencias_DE_TIERRA_Y_LA_CONSTRUCCION_ROLE',
    'DEPARTAMENTO_DE_ENERGIA_Y_MECAICA_ROLE',
    'DEPARTAMENTO_DE_ELECTRICA_ELECTRONICA_Y_TELECOMUNICACIONES_ROLE',
    'DEPARTAMENTO_DE_Ciencias_EXACTAS_ROLE']
  ,
  message: '{VALUE} no es un rol permitido'
};
```

Para que los usuarios puedan visualizar la página principal se los agrego de la siguiente manera en la siguiente ruta **serverIndicadores-master\routes\login.js**

El siguiente código hace una comunicación entre el **fontend** y el **backend**, y el **backend** le da las vistas al **frontend**.

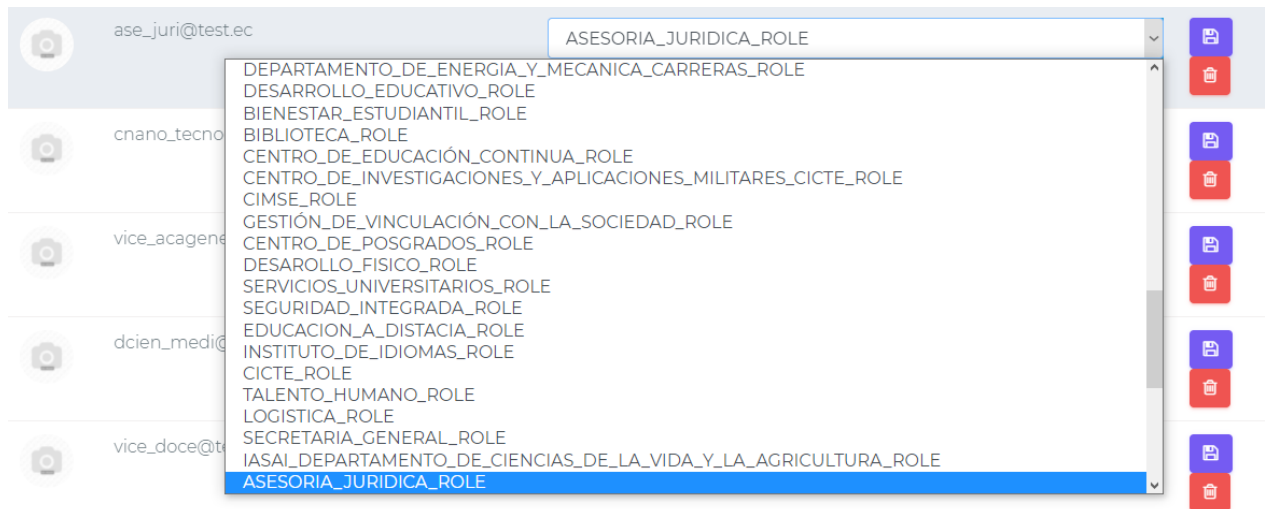
```
}
if (ROLE === 'DIRECTOR_UPDI_ROLE') {
  menu[0].submenu.unshift(
    { titulo : 'REFORMAS GENERALES', url: '/dashboard/reformasGenerales' },
    // { titulo : 'INDICADORES EVIADOS', url: '/dashboard/historial' }
  );
}

if (ROLE === 'VICERECTOR_ADMINISTRATIVO_ROLE') {
  menu[0].submenu.unshift(
    { titulo : 'REFORMAS GENERALES', url: '/dashboard/reformasGenerales' },
    // { titulo : 'INDICADORES EVIADOS', url: '/dashboard/historial' }
  );
}

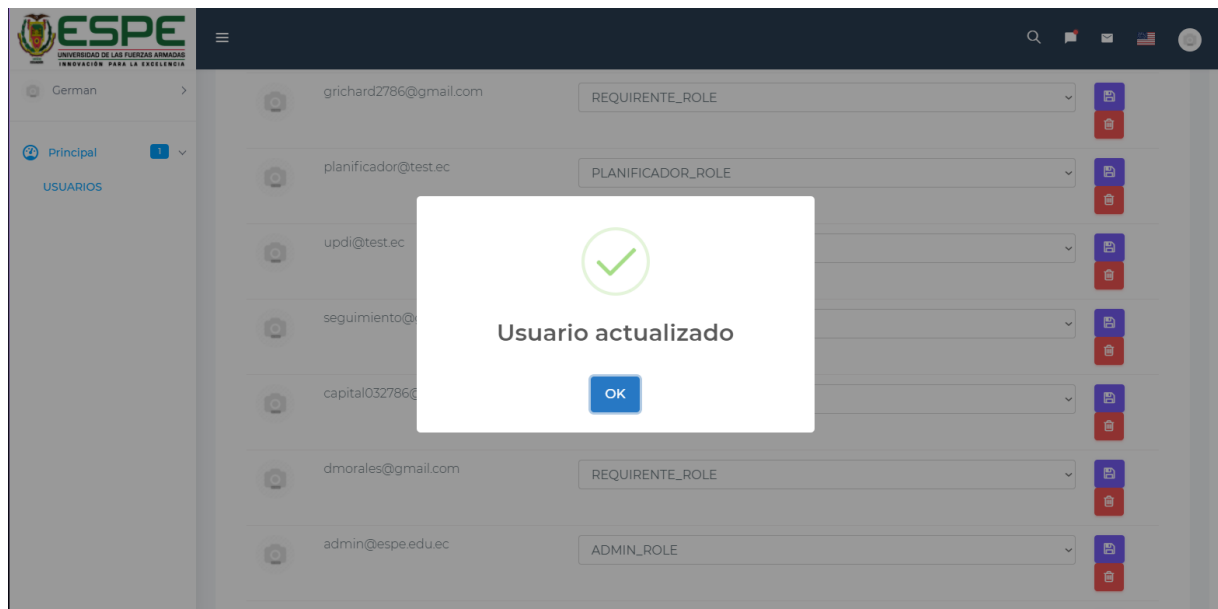
if (ROLE === 'VICERECTOR_ACADEMICO_GENERAL_ROLE') {
  menu[0].submenu.unshift(
    { titulo : 'REFORMAS GENERALES', url: '/dashboard/reformasGenerales' },
    // { titulo : 'INDICADORES EVIADOS', url: '/dashboard/historial' }
  );
}
```

PARTE VISUAL

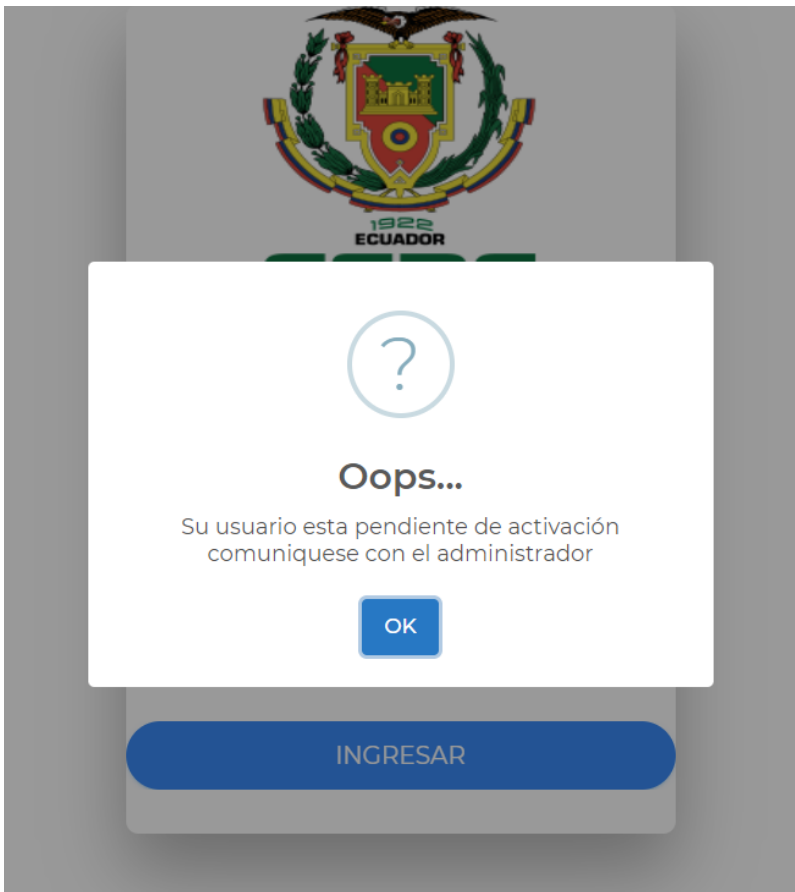
Para el administrador serán visibles todas las opciones creadas en `admin-usuarios.component.html`.



El administrador vera este mensaje al dar el rol a cada nuevo usuario que se registre.



El usuario que se vaya a iniciar sesión sin que el administrador le dé un rol le saldrá un mensaje que dice que el usuario está pendiente de activación.



FIN.