



***Apuntes de Angular. Juan Carlos Fernández García.
09 – Angular – Control de errores en la parte de angular***

Este documento va a ser bastante corto, ya que vamos a incluir únicamente el control de errores en la parte de Angular.

Hasta el documento 7, hacíamos las operaciones crud de forma que no nos dices error; creábamos alumnos, editábamos alumnos, etc, informando de todos los campos necesarios.

En el documento 8, cambiamos los métodos del servidor, para controlar cuando falta algún dato o cuando se actualiza algún registro, etc.

```
import { Injectable } from '@angular/core';
import { HttpClient, HttpHeaders } from '@angular/common/http';
import { Alumno } from '../components/alumnos/alumno';
import { catchError } from 'rxjs/operators';
import { throwError } from 'rxjs';
import Swal from 'sweetalert2';
import { Router } from '@angular/router';

@Injectable({
  providedIn: 'root'
})
export class AlumnoService {

  private urlEndPoint = 'http://localhost:8080/api/alumnos';

  private httpHeaders = new HttpHeaders({'Content-Type':
'application/json'});

  constructor(private http: HttpClient, private router: Router) { }

  getAlumnos() {
    return this.http.get( this.urlEndPoint);
  }

  creaAlumno( alumno: Alumno ) {
    return this.http.post(this.urlEndPoint, alumno, { headers:
this.httpHeaders})
    .pipe(
      catchError ( e => {
        console.log(e);
        Swal.fire(
          'Error al guardar',
          e.error.mensaje,
          'error'
        );
      })
    );
  }
}
```



*Apuntes de Angular. Juan Carlos Fernández García.
09 – Angular – Control de errores en la parte de angular*

```
    );  
    return throwError ( e );  
  
  })  
);  
}  
  
getAlumno ( id: number ) {  
  return this.http.get( `${this.urlEndPoint}/${id}` )  
    .pipe( catchError ( e => {  
      this.router.navigate(['/alumnos']);  
      // console.log(e);  
      Swal.fire(  
        'Error al editar',  
        e.error.mensaje,  
        'error'  
      );  
      return throwError ( e );  
    }  
  ));  
}  
  
updateAlumno( alumno: Alumno ) {  
  return this.http.put( `${this.urlEndPoint}/${alumno.id}`, alumno,  
    {headers: this.httpHeaders})  
    .pipe(  
      catchError ( e => {  
        console.log(e);  
        Swal.fire(  
          'Error al editar',  
          e.error.mensaje,  
          'error'  
        );  
        return throwError ( e );  
      }  
    )  
  );  
}  
  
deleteAlumno ( id: number ) {  
  return this.http.delete ( `${this.urlEndPoint}/${id}` );  
}
```



***Apuntes de Angular. Juan Carlos Fernández García.
09 – Angular – Control de errores en la parte de angular***

```
}
```

Únicamente probar a tener levantado el backend y el servidor del programa y debería funcionar.

Hasta aquí toda la parte del manejo de Angular sin seguridad y con un Crud con spring boot.