EXAMEN DE TALLERES 4

CIGARROA HERNÁNDEZ LUISA FERNANDA, 6° “M”

APP.CONTROLLER.TS

En este fragmento de este código, se realizó la definición de un controlador utilizando el framework NestJS. Se creó una clase llamada **`AppController`** y se decoró con el decorador **`@Controller('api/v1/bm')`,** lo que indica que las rutas manejadas por este controlador comenzarán con `/api/v1/bm`.

En el constructor de la clase **`AppController`**, se inyecta una instancia del servicio **`AppService`**, que se utiliza para realizar la lógica de negocio de la aplicación.

Dentro de la clase **`AppController`**, se define un método **`getHello()`** decorado con **`@Get()`**. Este método se encarga de manejar las solicitudes HTTP GET que llegan a la ruta definida por el controlador. En este caso, al recibir una solicitud GET en la ruta base del controlador, el método **`getHello()`** se ejecuta y devuelve un saludo obtenido del método **`getHello()`** del servicio **`AppService`**.

Los cambios realizados en este apartado incluyen la definición de un controlador en NestJS, la inyección de dependencias de un servicio y la definición de un método para manejar las solicitudes GET en una ruta específica.

src > TS app.controller.ts > ...

```ts
import { Controller, Get } from '@nestjs/common';
import { AppService } from './app.service';

@Controller('api/v1/bm')
export class AppController {
  constructor(private readonly appService: AppService) {}

  @Get()
  getHello(): string {
    return this.appService.getHello();
  }
}

```

APP.SERVICE.TS

El código introduce un servicio denominado `**AppService**` con el decorador `**@Injectable**`, indicando su capacidad de ser inyectado en otras partes del código. Este servicio ofrece un método `**getHello()**` que primero resuelve la ruta del directorio del proyecto usando el módulo `**path**`. Luego, emplea el módulo `**fs**` para leer el contenido del archivo `**index.html**` situado en el directorio `**src**`, devolviendo el contenido como una cadena de texto. En síntesis, el servicio facilita la lectura del contenido de un archivo HTML dentro del proyecto, lo cual puede ser útil para servir contenido estático en una aplicación NestJS.

INDEX.HTML

Se crea un **index.htlm** para poder direccionarlo con nuestro **app.service.ts** en el **index.htm** solo indicamos el título que desea que aparezca y le damos el diseño que nosotros queramos.

Index.html | app.controller.ts | app.service.ts

src > Index.html > html > body

```html
<!DOCTYPE html>
<html lang="es">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>EXAMEN DE TALLERES</title>
    <style>
        body {
            background-color: rgb(197, 175, 192);
            text-align: center;
            color: black;
        }

        h1 {
            color: black;
        }
    </style>
</head>
<body>
    <h1>EXAMEN DE TALLERES</h1>
    <p>LUISA FERNANDA CIGARROA HERNANDEZ, 6º M</p>
</body>
</html>
```

RESULTADO FINAL