



TÉCNICO EN DESARROLLO DE APLICACIONES WEB
Departamento de Informática
PROYECTO

[MyDiary] **Manual Técnico**

Autor: Rodrigo García Luengo.

Curso Académico: Desarrollo de Aplicaciones Web (DAW)

Índice

| | |
|---|-----------|
| 1 Introducción | 2 |
| 2 Objetivos | 3 |
| 3 Tecnologías involucradas | 4 |
| 4 Proceso de desarrollo | 7 |
| 5 Proceso de Despliegue | 9 |
| 6 Propuesta de mejora o trabajos futuros | 10 |
| 7 Bibliografía | 11 |

1 Introducción

Una agenda escolar virtual con la que un profesor, podrá enviar deberes o tareas a sus alumnos, sin preocuparse porque estos puedan olvidarse de apuntar o anotar en sus agendas físicas la tarea mandada.

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: *MyDiary*



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

2 Objetivos

El objetivo de esta aplicación web consiste en que un usuario “Profesor” pueda enviar una tarea a varios usuarios “Alumnos” de una misma clase.

3 Tecnologías involucradas

Esta aplicación web está formada por:

Angular 9

Angular es un framework diseñado para la creación de SPAs(Single Pages Applications),PWA(Aplicaciones Web Progresivas) y aplicaciones móviles, además de poder usarse en aplicaciones de escritorios.

Angular permite controlar la escalabilidad. Satisface requerimientos enormes de datos construyendo modelos en RxJS, Immutable.js o algún otro.

Renderiza el código para optimizarlo al máximo y provee de un enrutador propio para poder viajar entre componentes dentro de la aplicación.

Además, usa código TypeScript para el desarrollo de sus aplicaciones en el entorno cliente, permite usar HTML5 para el diseño gráfico y SCSS además de CSS3 para las hojas de estilos.

Más información sobre Angular: <https://docs.angular.lat/features>

Node.js, Express y NPM

Para la parte de Back-end de esta aplicación se ha utilizado Node.js con el framework que ofrece este, llamado Express.

Node.js es un entorno de ejecución para JavaScript construido con el motor JavaScript V8 de Chrome.

Express, es un framework escrito en JavaScript y alojado dentro de Node.js

NPM es el sistema de gestión de paquetes por defecto para Node.js.

MySQL,phpMyAdmin y XAMPP.

Para el almacenamiento de datos, se ha usado Xampp, MySql, usando el entorno gráfico de phpMyAdmin.

VSCode.

En cuanto a los IDEs usados para desarrollar esta aplicación, se ha usado únicamente VisualStudioCode.

VSCode es un editor de código fuente desarrollado por Microsoft, permite trabajar de manera muy fluida con los frameworks y entornos de ejecución usados para esta

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: *MyDiary*



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

aplicación, ya que Angular a su vez, es un producto de Google, entidad es dueña de Microsoft.

DIA.

La creación de diagramas entidad de relación se ha usado el editor Dia.

Bootstrap4 con CSS3

Con respecto al diseño de la aplicación, además de las tecnologías mencionadas anteriormente, se ha usado el framework Bootstrap4.

- **Modelo Relacional de la base de datos.**

Este modelo está adjunto en la ruta:

‘AppWebMyDiary/BBDD/MyDiaryDiagrama.dia’.

- **Guía de estilos.**

Dentro de cada componente existe un fichero “.scss” que recoge cada hoja de estilos asociados a este.

Además, existe un fichero styles.scss en la ruta ‘AppWebMyDiary/src/’ en donde se recogen los estilos generales para toda la aplicación.

- **Estructura del Proyecto.**

Front-end.

Dentro de ‘AppWebMyDiary’, observamos que existe una carpeta ‘src’.

Dentro de esta, se encuentra las siguientes subcarpetas:

app: Recoge todos los componentes creados y usados por la aplicación.

Son los siguientes:

‘Admin’: módulo y componentes para la zona administrador.

‘Alumno’: módulo para el usuario alumno.

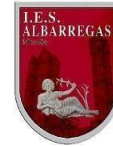
‘Auth’: módulos y servicio para hacer ‘login’ y registrarse.

‘Home’: módulo para la vista principal de la aplicación.

‘Models’: diferentes interfaces que recogen los modelos de datos que necesita la aplicación.

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: *MyDiary*



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

‘Profesor’: módulo y componentes para la zona del profesor.

‘Services’: servicios necesarios para la realización de diversas funcionalidades, en ellos se encuentran los servicios para, alumnos, profesores, tareas y usuarios.

‘Shared’: esta carpeta recoge los componentes compartidos para toda la aplicación, como el ‘header’.

Assets: contiene la siguiente carpeta.

- Una carpeta ‘images’:
En ella se encuentran las imágenes necesarias para la aplicación.

Back-end.

Dentro de la carpeta separada, APIRestMyDiary, se encuentra la parte servidora de la aplicación, que permite relacionar la zona cliente con la servidora, extrayendo datos de una base de datos en MySQL.

4 Proceso de desarrollo

Desarrollo del Front-end.

Se han creado la parte cliente usando Angular9, este framework ofrece la oportunidad de crear componentes, servicios, módulos y un enrutamiento entre estos de manera asíncrona.

Para cada rol de la aplicación se ha desarrollado un módulo, estos contienen 6 ficheros diferentes. Cada uno de estos ficheros hace referencia a:

- componente ‘.html’
Este es un HTML5 donde se generan las vistas de dicho modulo.
- módulo .module.ts
En él se recogen las importaciones necesarias para dicho modulo, además de la exportación e importación de los componentes o módulos externos que se deseen implementar. Es opcional, pero recomendable, usar aquí los componentes o módulos exclusivos a dicho módulo, ya que existe un fichero general llamado ‘app.module’, este ofrece, al importar en el cualquier fichero, su visualización o utilización en toda la aplicación.
- componente spect.ts
No se ha realizado ninguna lógica o metodología adicional en este fichero, está tal y como Angular nos lo ofrece.
- componente .ts
Realiza la lógica necesaria para el componente del módulo.
- componente .scss
Aquí se recogen los distintos estilos que se deseen implementar, tanto para el módulo como para, si los hubiera, sus componentes. No es obligatorio usar aquí todas las hojas de estilos que necesiten los componentes internos a este módulo, ya que cada uno de sus componentes internos posee una hoja de estilos propia .SCSS.
- enrutamiento routing.module.ts
En esta zona se recoge la ruta propia de dicho módulo. Sin embargo no hay que olvidar que al igual que los archivos .module.ts propios a cada módulo, existe un fichero routing.module.ts general a toda la aplicación, llamado ‘app-routing.module.ts’, en el se declaran todas las rutas de los módulos de la aplicación y es quien gestiona dichas rutas.

Desarrollo del Back-end.

Se ha desarrollado dentro del fichero index.js la programación necesaria para poder usar el framework de Express y la conexión con la base de datos de MySQL.

Autor/es: Rodrigo García Luengo

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: *MyDiary*



En 'index.js'. se encuentran las rutas necesarias para poder acceder a la base de datos y devolver los datos solicitados.

Al inicio de este fichero se encuentra la lógica necesaria para la conexión con la base de datos y el uso de express.

Después encontraremos las rutas necesarias para poder acceder a la base de datos y devolver los datos solicitados.

5 Proceso de Despliegue

Para poder ejecutar esta aplicación de manera local es necesario seguir los siguientes pasos.

- Primero debemos instalar Node.js, para ello iremos a su página oficial (<https://nodejs.org/es/>) y descargaremos la última versión LTS que aparece, ya que es la recomendada para la mayoría de los usuarios, puesto que está testeada y no tiene funcionalidades en modo beta.

Una vez instalado, debemos usar el terminal de nuestro pc, y debemos actualizar 'npm' de la siguiente manera: `'npm install -g npm@latest'`.

De esta manera actualizaremos 'npm' de manera global en nuestro pc.

- Segundo, debemos tener instalado Angular9 y por ende su 'CLI', para ellos iremos a la documentación de Angular (<https://angular.io/guide/setup-local>).

En su documentación nos indica paso a paso como crear un nuevo proyecto angular, pero en nuestro caso tan solo debemos instalar de manera global dicho CLI, para ello:

`'npm install -g @angular/cli'`

Esto instalará la última versión del CLI de Angular, para asegurarnos de nuestra versión y de que el CLI ha sido instalado, haremos `'ng version'` y aparecerá un mensaje de Angular indicando dicha versión.

Ahora en en la consola del pc, dentro del directorio de la aplicación, es decir, dentro de 'AppWebMyDiary/', debemos ejecutar el siguiente comando para que se instalen todos los módulos de node usados en esta aplicación. Para ello ejecutaremos el siguiente comando.

`'npm install '`

Una vez acabado el proceso podremos desplegar de manera local la aplicación usando los comandos que Angular CLI nos ofrece, para ello, ejecutaremos el comando `'ng serve -o'`.

Con este comando se abrirá la aplicación directamente en nuestro navegador.

La consola de nuestro pc no debemos cerrarla y debe estar ejecutándose en modo administrador.

Proyecto “Desarrollo de Aplicaciones Web”

Título del Proyecto: *MyDiary*



JUNTA DE EXTREMADURA

Consejería de Educación y Empleo

- Cuarto, debemos tener instalado MySQL en nuestro pc, en mi caso he usado XAMPP, pero obligatorio correr MySQL, solo que se esté ejecutando en el puerto 3306 y esté activo.

Debemos importar la base de datos en MySQL, se encuentra en /AppWebMyDiary/BBDD/mydiarybd.sql

En la siguiente dirección explican cómo importar una base de datos:

<https://clouding.io/hc/es/articles/360011334200-C%C3%B3mo-importar-la-base-de-datos-MySQL-desde-la-l%C3%ADnea-de-comandos>

Ahora abriremos una nueva consola en nuestro pc, para poder instalar el paquete de node necesario en la zona servidora.

Para ellos iremos a la ruta donde se encuentre la carpeta APIRestMyDiary y ejecutaremos de nuevo el comando ‘**npm install**’, para que instales todos los módulos y dependencias que necesite.

Una vez esté corriendo la base de datos, tan solo tendremos que ejecutar en la consola donde hemos abierto la ruta de APIRestMydiary, ‘node.’

- Por último, pero no menos importante, debemos instalar la siguiente extensión en nuestro navegador para que no afecte la política de CORS a nuestra aplicación. Para ello buscaremos en las extensiones de Google Chrome Store la siguiente extensión:

‘Allow CORS: Access-Control-Allow-Origin’

o ir directamente a la siguiente ruta:

‘<https://chrome.google.com/webstore/detail/allow-cors-access-control/lhobafahddgcelffkeicbaginieeejlf?hl=es>’

Ahora ya está todo listo para poder usar la Aplicación.

6 Propuesta de mejora o trabajos futuros

Hay varias propuestas de mejora en esta aplicación:

La principal y más importante es la parte de Back-end, ya que no está bien estructurada y debería realizar más validaciones que la parte de Front-end.

Otra mejora dentro de la parte Cliente, sería añadir nuevas funcionalidades a los Roles de Profesor, Alumno y Administrador.



7 Bibliografía

Angular: <https://angular.io/>

Node.js: <https://nodejs.org/es/>

XAMMP: <https://www.apachefriends.org/es/index.html>

Express: <https://expressjs.com/es/guide/routing.html>

Bootstrap: <https://getbootstrap.com/>

Canal de Youtube: https://www.youtube.com/channel/UC3QuZuJr2_EOUak8bWUd74A

(En él, se encuentran múltiples tutoriales, sus videos sobre cómo usar el framework express junto con la documentación de este han sido claves en el desarrollo de la parte back-end).