

ANEXO V: Documentación técnica

Desarrollo de una plataforma o aplicación web para la gestión de un gimnasio

Trabajo fin de grado
Grado en Ingeniería Informática



**VNiVERSIDAD
D SALAMANCA**

Julio de 2021

Autor

Alberto Martín Peralejo

Tutores

**André Filipe Sales Mendes
Gabriel Villarrubia González
Juan Francisco de Paz Santana**

Índice general

1. Introducción	1
2. Estructura del sistema	2
2.1. Frontend	2
2.2. Backend	2
3. Especificación de la documentación	4
3.1. Frontend	4
3.2. Backend	4
4. Bibliografía	5

1. Introducción

La finalidad que tiene este documento es la de facilitar a otros posibles programadores la comprensión del código de la aplicación.

La programación del sistema se divide en 2 partes: **Frontend** y **Backend**.

La estructura del Anexo V: Documentación técnica es la siguiente:

- Estructura del sistema.
- Especificación de la documentación.

2. Estructura del sistema

2.1. Frontend

El código fuente de la aplicación web se encuentra dentro de la carpeta “./Código Fuente/Frontend”. Este código se encuentra distribuido en varios archivos. Cabe destacar que las clases y los nombres corresponden con los asignados en la fase de diseño.

En esta carpeta de código se ha seguido la siguiente estructura:

- **“./App.vue”**: Raíz de la vista de la aplicación GBAApp desarrollada con Vue.js.
- **“./main.js”**: Script de inicialización del Frontend y dónde se resuelven las direcciones.
- **“./Store”**: Módulo Vuex que maneja un estado centralizado a lo largo de toda la aplicación.
- **“./Plugins”**: Módulo Vuetify que es accedido por todos los componentes.
- **“./Components”**: Subcarpeta que contiene los componentes que se van a utilizar por otros elementos en la aplicación.
- **“./Services”**: Subcarpeta que contiene los scripts que llamarán a las funciones del Backend.
- **“./Models”**: Subcarpeta que contiene los scripts con los modelos de datos utilizado en el sistema.

Para desplegar la aplicación web se deben instalar las dependencias declaradas en el package.json mediante npm install.

2.2. Backend

El código fuente de la aplicación web se encuentra dentro de la carpeta “./Código Fuente/Backend”. Este código se encuentra distribuido en varios archivos. Cabe destacar que las clases y los nombres corresponden con los asignados en la fase de diseño.

En esta carpeta de código se ha seguido la siguiente estructura:

- **“./server.js”**: Es el fichero principal del Backend. Para inicializar el subsistema se ejecuta dicho fichero cuyo objetivo es cargar los parámetros de configuración del subsistema, establecer conexión con la base de datos y comenzar a escuchar peticiones en el puerto 4000.

- **“./Routes”**: Subcarpeta que contiene el script donde se encuentran las rutas de acceso al sistema.
- **“./Middlewares”**: Subcarpeta que contiene los scripts que comprueban ciertas condiciones según la petición que llega al sistema. Se encuentran divididos según los datos del sistema que se quieren tratar.
- **“./Controllers”**: Subcarpeta que contiene los scripts con las acciones del servidor según la petición que llega al sistema. Se encuentran divididos según los datos del sistema que se quieren tratar.
- **“./Services”**: Subcarpeta que contiene el script que llamarán los controladores cuando un usuario solicite resetear su contraseña.
- **“./Utils”**: Subcarpeta que contiene los scripts y las templates para enviar los emails cuando un usuario solicita resetar su contraseña.
- **“./Config”**: Subcarpeta que contiene los scripts con configuraciones de la base de datos y de la autenticación.
- **“./Upload”**: Subcarpeta donde se guardarán todas las imágenes de los usuarios y actividades.
- **“./Models”**: Subcarpeta que contiene los scripts con los modelos de datos utilizado en el sistema. Son modelos de mongoose para el uso con MongoDB.

Para desplegar la aplicación web se deben instalar las dependencias declaradas en el package.json mediante npm install.

3. Especificación de la documentación

Para mayor documentación del código se ha generado código HTML a partir de generadores de código que explica más detalladamente cada fichero.

3.1. Frontend

Para la generación de documentación del código del Frontend se ha utilizado dos herramientas diferentes:

- **Vue Styleguidist:** Facilita la documentación de proyectos en el framework Vue.js [3], ha sido utilizado para la generación de la documentación de de todos los componentes con extensión “.vue” de la aplicación web. Para el resto se ha usado JSDoc.

La documentación generada se encuentra en el directorio “./Documentación/Frontend/dist” y se accede a ella mediante index.html.

- **JSDoc:** Se ha utilizado en el resto de ficheros con extensión “.js” [1]. La documentación generada se encuentra en el directorio “./Documentación/Frontend/docs” y se encuentra un html para cada uno de los scripts utilizados.

3.2. Backend

Para la generación de documentación del código del Frontend se ha utilizado la herramienta JSDoc.

La documentación generada se encuentra en el directorio “./Documentación/Backend/docs” y se encuentra un html para cada uno de los scripts utilizados.

Cabe destacar que se ha hecho uso de Swagger.io [2] para documentar el fichero router.js de la carpeta ./routes.

Swagger ayuda a crear un documento Swagger desde un navegador web proporcionando una vista en paralelo del documento Swagger y de las definiciones de API REST resultantes.

Dicha documentación se encuentra en el directorio “./Documentación/Backend/docs/Swagger UI.pdf”.

4. Bibliografía

- [1] Jsdoc. URL <https://jsdoc.app/>.
- [2] Swagger.io. URL <https://swagger.io/resources/articles/documenting-apis-with-swagger/>.
- [3] Vue styleguidist. URL <https://vue-styleguidist.github.io/docs/GettingStarted.html>.