

Anexo V: Documentación técnica

FitApp: Aplicación para controlar el plan de nutrición y entrenamiento

Trabajo de Fin de Grado

Ingeniería Informática



**VNiVERSiDAD
D SALAMANCA**

Septiembre de 2023

Autor

María Pérez Morales

Tutor/a

André Filipe Sales Mendes

ÍNDICE DE CONTENIDO

5.1 INTRODUCCIÓN	4
5.2. ESTRUCTURA DEL SISTEMA.....	4
5.3 ESPECIFICACIÓN DE LA DOCUMENTACIÓN.....	6
5.4. DESPLIEGUE DE LA APLICACIÓN.....	8
5.5 REFERENCIAS.....	9

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Interfaz gráfica documentación técnica	8
---	---

5.1 INTRODUCCIÓN

Este anexo proporciona una guía detallada sobre el código software desarrollado, con el fin de simplificar su comprensión y asegurar que cualquier modificación futura se realice de manera eficiente.

La organización del documento es la siguiente:

- Estructura del sistema.
- Especificación de la documentación.
- Despliegue de la aplicación.

A continuación, ofreceremos una descripción de las carpetas y ficheros, detallando su función y propósito.

5.2. ESTRUCTURA DEL SISTEMA.

La funcionalidad del sistema se define mediante su estructura. Esta se divide en:

- **Interfaz:** los usuarios interactúan con el sistema a través de una aplicación móvil.
- **Desarrollo:** el sistema ha sido desarrollado utilizando el framework de Vue basado en HTML, CSS y JavaScript.
- **Base de Datos:** se emplea Firebase para guardar y recuperar datos.

La estructura del proyecto es la siguiente:

- **App.vue:** es el archivo principal en una aplicación Vue.js que actúa como componente raíz. Define y estructura el contenido visual y funcional para una ruta específica, sirviendo como punto de anclaje para otros componentes secundarios.
- **main.js:** es el archivo de arranque para proyectos basados en Vue.js. Se encarga de inicializar y configurar el entorno del framework, incorporando plugins necesarios y estableciendo el punto de entrada a través de App.vue, donde se inicia la construcción de la aplicación Vue.
- **node_modules:** es un directorio que almacena las bibliotecas y dependencias requeridas por una aplicación Node.js. Estas dependencias son definidas en el archivo package.json y se instalan a través de NPM (Node Package Manager). Estas bibliotecas facilitan el desarrollo al proveer funcionalidades preexistentes.
- **package.json:** es un archivo fundamental para proyectos basados en Node.js, ya que describe las propiedades y dependencias del proyecto. Por su parte, **package-lock.json** garantiza la consistencia de las dependencias instaladas al registrar versiones específicas. Ambos son utilizados por NPM para administrar y asegurar el correcto funcionamiento del proyecto y sus dependencias asociadas.
- **ionic.config.json:** es esencial para proyectos desarrollados con Ionic, pues actúa como un centro de configuración y gestión. A través de este, es posible definir y ajustar parámetros específicos del proyecto y sus dependencias, garantizando que la aplicación se ejecute de manera óptima.
- **firebase.js:** archivo que incluye la configuración esencial para establecer una conexión con Firebase, la plataforma de desarrollo web y móvil de Google. Este archivo es crucial para que las aplicaciones interactúen correctamente con los servicios de Firebase, ya que contiene las credenciales necesarias.

- **firebase.json:** es un archivo de configuración utilizado por Firebase que define propiedades y ajustes para diversos servicios dentro del proyecto. Facilita la gestión y despliegue de aplicaciones en la plataforma Firebase.
- **index.js:** gestiona las rutas de la aplicación, determinando qué componente se despliega en cada una. Es el núcleo del enrutamiento del proyecto.
- **Carpeta assets:** son archivos o recursos utilizados en una aplicación para mejorar su interfaz y experiencia del usuario. Estos incluyen gráficos, imágenes, fondos y emoticonos, que son esenciales para el diseño visual y la interactividad de la aplicación.
- **Carpeta views:** son las representaciones gráficas de la interfaz de usuario en una aplicación. Incluyen distintas pantallas y los métodos asociados que determinan su comportamiento y funcionalidad.
 - **Archivo RegistrarUsuario.vue:** es un formulario destinado a los usuarios no registrados. Aquí, el usuario introduce datos personales como nombre, apellidos, edad, peso, entre otros esenciales para garantizar su adecuado registro y acceso a la aplicación.
 - **Archivo CuentanosDeTi.vue:** es un formulario destinado a usuarios no registrados. Aquí pueden ingresar datos esenciales para la aplicación, como su preferencia de dieta o el deporte de su interés.
 - **Archivo InicioSesion.vue:** este archivo contiene un formulario que solicita al usuario su email y contraseña. Al introducir correctamente estos datos, el usuario puede iniciar sesión y acceder a todas las funcionalidades de la aplicación.
 - **Archivo NuevaContrasena.vue:** es un archivo destinado al restablecimiento de contraseñas olvidadas. Incluye un formulario para ingresar el email del usuario, y tras su envío, remite un enlace de restablecimiento al correo proporcionado.
 - **Archivo PaginaAlimentacion.vue:** es un archivo diseñado para mostrar la dieta seleccionada por el usuario, ofreciendo diversas opciones de comidas para incorporar al registro. Cuenta con varias funcionalidades:
 - **Pestaña principal:** visualiza las comidas que se han seleccionado como favoritas para que el usuario pueda iterar con ellas, ya sea añadiendo o eliminando alimentos al registro.
 - **Pestaña de comidas favoritas:** permite agregar/eliminar y visualizar las comidas predilectas que se mostrarán en la pantalla principal del archivo.
 - **Pestaña de organización calórica:** ofrece al usuario la capacidad de planificar y organizar las calorías que consumirá a lo largo del día, promoviendo una gestión nutricional más eficiente.
 - **Archivo PaginaDiario.vue:** este archivo gestiona la visualización de las estadísticas de calorías, tanto ingeridas por alimentos como gastadas en actividades físicas. Además, integra una pestaña de historial que resume las acciones del usuario desde su registro.
 - **Archivo PaginaEjercicios.vue:** es un archivo destinado a presentar diversos ejercicios para actividades deportivas. Cuenta con varias funcionalidades:
 - **Pestaña principal:** visualiza los ejercicios que se han seleccionado como favoritos para que el usuario pueda iterar con ellos, iniciando una actividad deportiva.
 - **Pestaña de ejercicios favoritos:** permite agregar/eliminar y visualizar los ejercicios predilectos que se mostrarán en la pantalla principal del archivo.
 - **Pestaña de información:** ofrece al usuario información útil que pueda tener en cuenta a la hora de registrar sus actividades dentro de la aplicación.

- **Archivo PaginaPerfil.vue:** es un archivo diseñado para visualizar y editar los datos del usuario. Cualquier modificación realizada aquí puede influir en el comportamiento general de la aplicación.

5.3 ESPECIFICACIÓN DE LA DOCUMENTACIÓN

Para documentar el código de la aplicación, se utilizará **JSDoc**, una herramienta que permite generar documentación del código. Es importante señalar que JSDoc es una herramienta de documentación específica para JavaScript, y su principal ventaja es que genera documentación técnica directamente a partir de los comentarios introducidos en el código fuente.

Instalación:

1. Instalar JSDoc y el plugin para Vue con el comando:

```
npm install --save-dev jsdoc jsdoc-vuejs
```

2. Instalar el compilador de Vue que corresponda a su versión:

```
npm install --save-dev @vue/compiler-sfc
```

Generación de Documentación: Una vez que el código esté adecuadamente comentado, ejecute:

1. yarn docs
2. Al ejecutar este proceso, se creará un sitio web en formato HTML. Se puede encontrar la documentación generada en la carpeta *Documentación Técnica*.

Vemos un ejemplo de documentación técnica con la vista de página de ejercicios en la que se detallarán las diferentes funciones junto con una breve descripción de cada una de ellas:

Module: PaginaEjercicios

Default exported object containing component's data and methods.

Source: [PaginaEjercicios.vue, line 11](#)

Methods

"añadirEjercicio"()

Adds a new exercise to the user's exercise list.

Source: [PaginaEjercicios.vue, line 123](#)

actualizarKcalFavoritosEjercicios()

Updates the list of favorite exercises and their kcal.

Source: [PaginaEjercicios.vue, line 296](#)

alertaGuardado()

Shows an alert indicating successful saving.

Source: [PaginaEjercicios.vue, line 287](#)

calculoManual()

Manually calculates kcal spent and total time for an exercise.

Source: [PaginaEjercicios.vue, line 207](#)

(async) guardarDatos()

Updates Firestore with the user's daily data and exercise list.

Source: [PaginaEjercicios.vue, line 86](#)

iniciarCronometro()

Starts a timer.

Source: [PaginaEjercicios.vue, line 172](#)

pararCronometro()

Stops the timer and calculates the kcal spent.

Source: [PaginaEjercicios.vue, line 191](#)

porcentajeAgua()

Calculates the user's daily water consumption as a percentage.

Source: [PaginaEjercicios.vue, line 166](#)

(async) recuperarID()

Retrieves the authenticated user's ID and related data from Firestore.

Source: [PaginaEjercicios.vue, line 43](#)

reiniciarCronometro()

Resets the timer.

Source: [PaginaEjercicios.vue, line 230](#)

restarAgua()

Decrements the user's daily water consumption.

Source: [PaginaEjercicios.vue, line 154](#)

restartInterval()

Restarts the recurring task.

Source: [PaginaEjercicios.vue, line 280](#)

Home

Modules

- CuentanosDeTi
- PaginaDiario
- PaginaEjercicios
- PaginaPerfil
- RegistrarUsuario
- firebase/auth

Global

- calcularKcalNecesarias
- editar
- eleccionComidas
- logout

```
startInterval()
Starts a recurring task that checks and updates the kcal spent and displays alerts.
Source: PaginaEjercicios.vue, line 245

sumarAgua()
Increments the user's daily water consumption.
Source: PaginaEjercicios.vue, line 139
```

ILUSTRACIÓN 1: INTERFAZ GRÁFICA DOCUMENTACIÓN TÉCNICA

Esta no solo ofrece un desglose detallado de todas las funciones presentes en cada vista de la aplicación, sino que también proporciona una definición clara y concisa de cada función, facilitando así la comprensión y el mantenimiento del código.

5.4. DESPLIEGUE DE LA APLICACIÓN

El despliegue de una aplicación es una fase crucial que implica llevar la versión de desarrollo de la aplicación a un entorno de producción para que los usuarios finales puedan acceder y usarla.

Específicamente, este documento describe el proceso de despliegue de una aplicación móvil desarrollada con las tecnologías Vue e Ionic. Para ello se necesitará tener preinstalado Node.js y npm antes de comenzar con el despliegue:

- Node.js: es un entorno de ejecución que permite utilizar JavaScript para ejecutar código fuera de un navegador web, lo que lo convierte en una plataforma versátil para el desarrollo de aplicaciones y servidores. Es esencial para gestionar y ejecutar aplicaciones basadas en Vue e Ionic.
- npm: es la herramienta principal para gestionar paquetes en el entorno de Node.js. Facilita la descarga e instalación de bibliotecas y módulos que amplían las capacidades de las aplicaciones JavaScript.

Antes de iniciar el proceso de despliegue, es fundamental instalar las dependencias de la aplicación. Estas dependencias están detalladas en el archivo package.json.

Para instalarlas, se debe ejecutar el siguiente comando: `npm install`. Este comando descargará e instalará todas las bibliotecas y módulos esenciales para asegurar el adecuado desempeño de la aplicación.

Una vez finalizada la instalación de las dependencias, debes compilar y construir los archivos de la aplicación. Para hacerlo, utiliza el siguiente comando: `npm run build`.

Al ejecutarlo, este comando hará uso de la configuración establecida por Vue e Ionic para construir los archivos HTML, CSS y JavaScript necesarios.

Una vez finalizada la construcción, se generarán los archivos optimizados y listos para ser desplegados en un servidor web.

Siguiendo estos pasos, se consigue una aplicación móvil basada en Vue e Ionic disponible en línea para todos los usuarios.

5.5 REFERENCIAS

- JSDoc. Recuperado el 01/09/2023, de [jsdoc-vuejs - npm \(npmjs.com\)](https://npmjs.com/jsdoc-vuejs).