

Anexo IV

Manual del programador

TRABAJO DE FIN DE GRADO
GRADO EN INGENIERÍA INFORMÁTICA



VNiVERSIDAD
D SALAMANCA

Marzo de 2023

Autor

Rodrigo de la Calle Alonso

Tutor/a

Álvaro Lozano Murciego

Tabla de contenido

1 INTRODUCCIÓN	1
2 CÓDIGO.....	2
2.1 ESTRUCTURA	2
2.2 REPOSITORIO Y CONTROL DE VERSIONES	3
2.3 DOCUMENTACIÓN	5
3 SOFTWARE DE DESARROLLO	6
3.1 SOFTWARE PARA LA PWA	6
3.2 SOFTWARE PARA EL BACK-END.....	6
3.3 ENTORNO DE DESARROLLO INTEGRADO (IDE).....	7
4 BILIOGRAFÍA	8

Tabla de Ilustraciones

ILUSTRACIÓN 1 DIRECTORIO PWA EN GITHUB	3
ILUSTRACIÓN 2 "README.MD"	4
ILUSTRACIÓN 3 JSDOC FOREMANDATA STORE.....	5
ILUSTRACIÓN 4 JSDOC WORK OBJECT.....	5
ILUSTRACIÓN 5 VISUAL STUDIO CODE Y VOLAR	7
ILUSTRACIÓN 6 IDE ARDUINO.....	7

1 Introducción

En este manual, se proporciona orientación y documentación detallada sobre el desarrollo, implementación y mantenimiento de un proyecto de software específico. Está diseñado para ayudar a los programadores y desarrolladores a comprender y utilizar eficazmente el código fuente, las herramientas, las tecnologías y los procesos asociados con el proyecto.

2 Código

2.1 Estructura

El código fuente del proyecto se divide en dos directorios. Por un lado, el directorio “PWA” que contiene, como bien su nombre indica, todos los archivos relacionados con el desarrollo de la PWA (*Progressive Web App*) [1] basada en vue.js, estos se organizan de la siguiente manera:

- Carpeta "**src**": esta es la carpeta principal que contiene todo el código fuente del proyecto.
 - Carpeta "**assets**": aquí se almacenan los archivos estáticos, como imágenes, fuentes, iconos, etc.
 - Carpeta "**components**": contiene los componentes Vue reutilizables de la aplicación.
 - Carpeta "**views**": aquí se encuentran las vistas principales de la aplicación, que se componen utilizando componentes.
 - Carpeta "**router**": contiene los archivos relacionados con la configuración del enrutador, como las rutas y las configuraciones de navegación.
 - Carpeta "**store**": aquí se encuentra la configuración y los módulos de pinia, que se utiliza para administrar el estado de la aplicación.
 - Carpeta "**services**" o "**api**": contiene los archivos que manejan las llamadas a la API y las interacciones con servicios externos.
 - Archivo "**main.js**": Es el punto de entrada de la aplicación, donde se configura y se inicializa Vue.js.
- Carpeta "**public**": aquí se encuentran los archivos estáticos que se copian directamente en la carpeta de salida final durante el proceso de construcción, como el archivo "index.html" principal, favicon, etc.
- Carpeta "**locale**": contiene los diccionarios del proyecto.
- Archivo "**package.json**": este archivo contiene la configuración del proyecto, las dependencias, los scripts de construcción, las versiones de las dependencias, etc.
- Archivo "**vite.config.js**": se utiliza para personalizar la configuración de Vite en el proyecto. Al crear este archivo, puedes especificar diferentes opciones y ajustes para el entorno de desarrollo y producción como *plugins* o rutas de alias.

Por otro lado, el directorio “Backend” contiene dos elementos. El primero es el archivo “index.js” que contiene las funciones implementadas en *firebase cloud functions*. El segundo, es el directorio “Arduino” que contiene el proyecto con el código referente a la programación del dispositivo M5 Stick-CPlus [2].

2.2 Repositorio y control de versiones

Para almacenar el proyecto de la PWA y su correspondiente control de versiones se ha utilizado la plataforma GitHub [3]. El proyecto está ubicado en la dirección: <https://github.com/rodriCalleA/TimeCheck>, como se puede apreciar en la Ilustración 1. También cuenta con archivo “README.md” que contiene los logos y una breve descripción del proyecto (Ilustración 2).

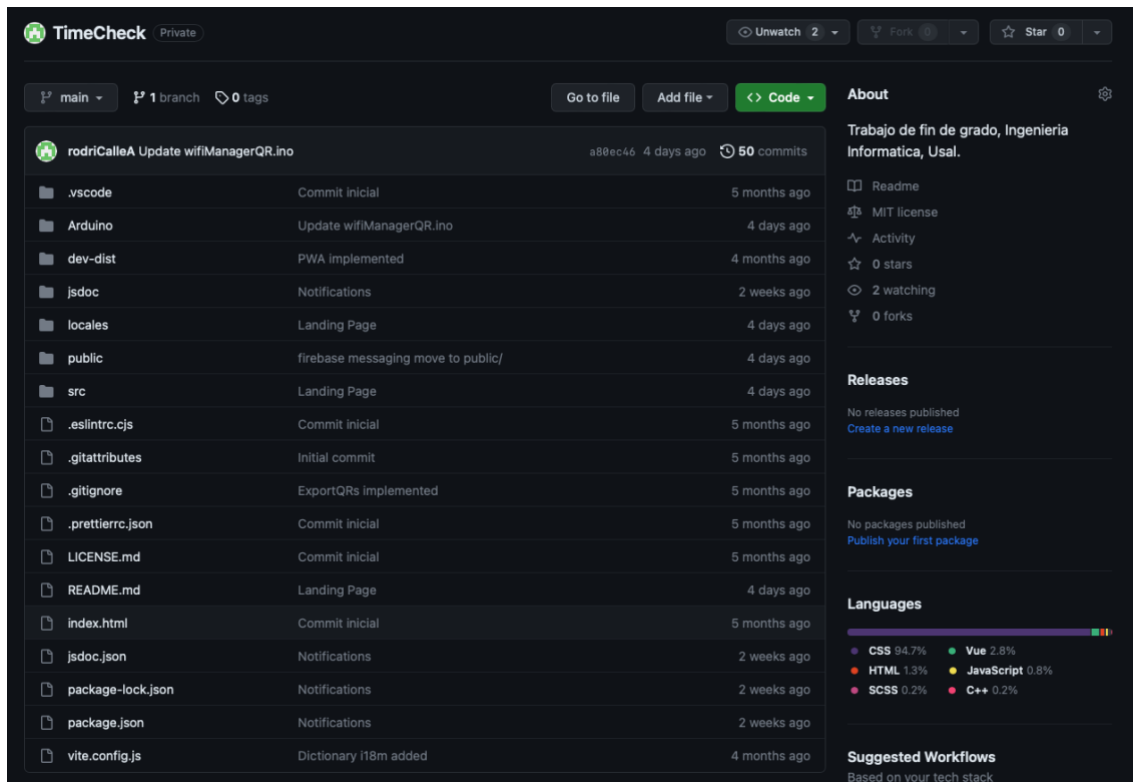
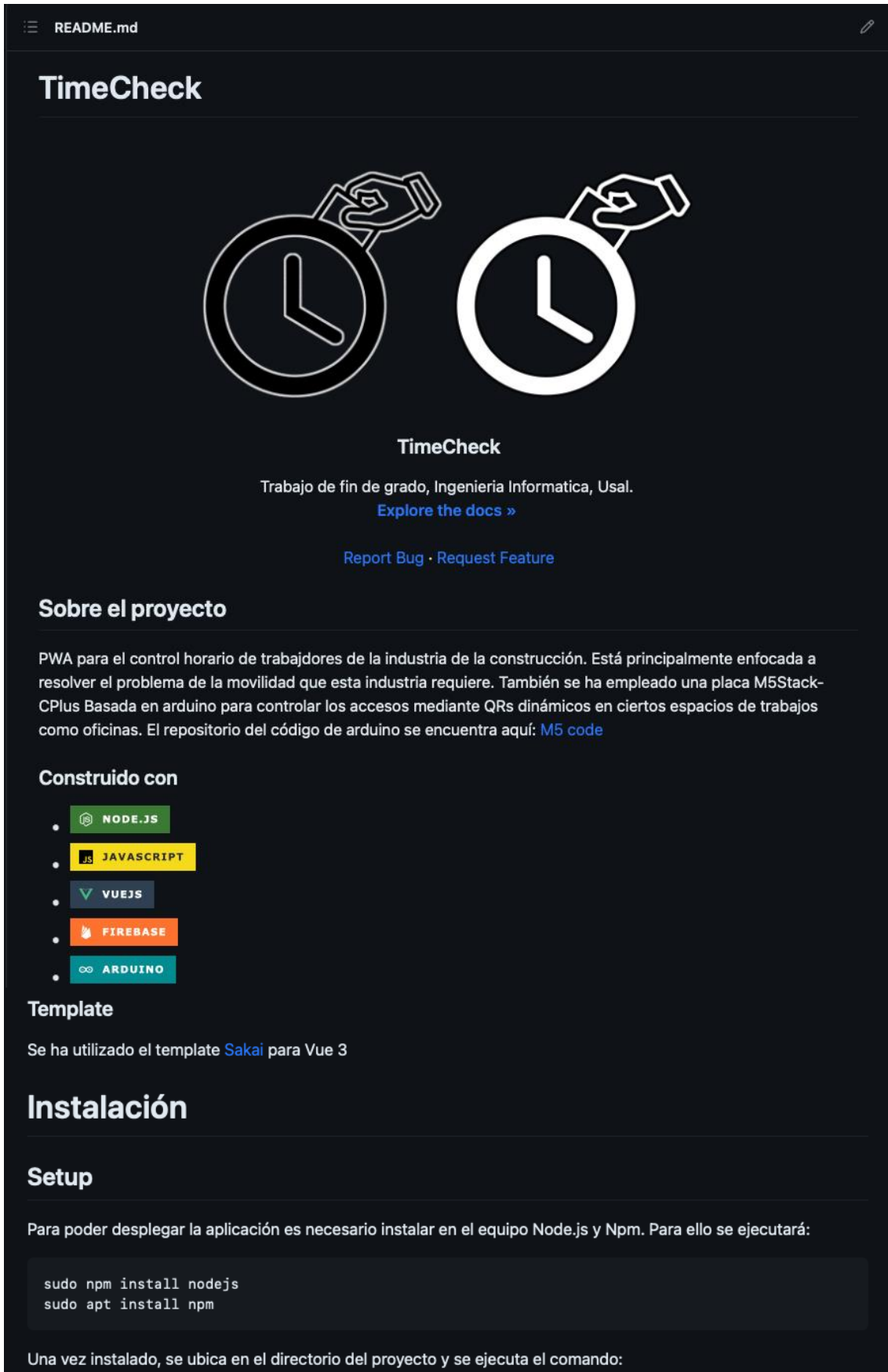


Ilustración 1 Directorio PWA en GitHub








The image shows a screenshot of a GitHub README.md file for a project named "TimeCheck". The page has a dark theme. At the top, it says "README.md" in the upper left corner and has a pencil icon in the upper right. The main heading is "TimeCheck" in a large, bold font. Below the heading are two identical white line-art icons of a hand holding a clock. Underneath the icons, the title "TimeCheck" is repeated in a smaller font, followed by the text "Trabajo de fin de grado, Ingeniería Informática, Usal." and a blue link "Explore the docs »". Below that are two more blue links: "Report Bug" and "Request Feature".

Sobre el proyecto

PWA para el control horario de trabajadores de la industria de la construcción. Está principalmente enfocada a resolver el problema de la movilidad que esta industria requiere. También se ha empleado una placa M5Stack-CPlus Basada en arduino para controlar los accesos mediante QRs dinámicos en ciertos espacios de trabajos como oficinas. El repositorio del código de arduino se encuentra aquí: [M5 code](#)

Construido con

-  NODE.JS
-  JAVASCRIPT
-  VUEJS
-  FIREBASE
-  ARDUINO

Template

Se ha utilizado el template [Sakai](#) para Vue 3

Instalación

Setup

Para poder desplegar la aplicación es necesario instalar en el equipo Node.js y Npm. Para ello se ejecutará:

```
sudo npm install nodejs
sudo apt install npm
```

Una vez instalado, se ubica en el directorio del proyecto y se ejecuta el comando:

Ilustración 2 "README.md"

2.3 Documentación

Para la documentación del código se ha usado la herramienta JSDoc para VueJS [4], que es una herramienta que convierte los comentarios en el código fuente de JavaScript en documentación HTML (Ilustración 3, Ilustración 4).

Para el uso de JSDoc se han seguido los siguientes pasos:

- **Instalación:** `npm install --save-dev jsdoc jsdoc-vuejs`.
- **Ejecución:** `sudo npm run jsdoc`.

Module: foremanData

Auth Module

Source: [foremanData.js, line 7](#)

Methods

(inner) module:foremanData#addNewAbsence Añade una nueva ausencia a un empleado.(qr, absence)

Parameters:

Name	Type	Description
qr	String	uid del empleado a registrar
absence	String	tipo de ausencia

Source: [foremanData.js, line 221](#)

Home

Modules

- Auth
- companyData
- foremanData
- sessionData

Global

Ilustración 3 JSDoc foremanData Store

Work

Type:

- Object

Properties:

Name	Type	Description
reference	number	
name	string	
location	string	

Source: [companyData.js, line 8](#)

Ilustración 4 JSDoc Work Object

3 Software de desarrollo

3.1 Software para la PWA

En esta sección se explica como abrir el proyecto mediante VueJS, Firebase y distintos *plugins* instalados mediante npm.

Lo primero es la instalación de NodeJS y npm [5] mediante su instalador o con la terminal (`sudo apt install nodejs` y `sudo apt install npm`). Después, desde el directorio “PWA”, se instalan las dependencias. Algunas de las opciones útiles se pueden encontrar en “package.json” son:

- **npm install:** installation.
- **npm run dev:** ejecución de la aplicación en modo de desarrollo de forma local.
- **npm run build:** construcción del proyecto para producción.
- **npm run preview:** se utiliza para ejecutar una aplicación ya construida como una vista previa de la compilación de producción.

Para poder usar Firebase y sus utilidades se tiene que instalar la utilidad de consola denominada Firebase CLI [6], usando npm, y asociándose con un proyecto de Firebase creado. Pasos:

- **npm install -g firebase-tools:** Instalación.
- **firebase login:** Inicio sesión en Firebase.
- **firebase init:** Inicio proyecto.

3.2 Software para el back-end

En esta parte se aloja la PWA con Firebase Hosting [6], y se trata el *backend* mediante Firebase Cloud Functions [6]. Al igual que en la parte de la PWA, se tiene que instalar Firebase CLI.

- **npm install -g firebase-tools:** Instalación.
- **firebase login:** Inicio sesión en Firebase.
- **firebase init (Seleccionando Hosting):** Inicio proyecto de Firebase Hosting.
- **firebase init functions:** Inicio proyecto de Firebase Cloud Functions.
- **firebase deploy:** Despliegue de sitio web y funciones.
- **firebase deploy --only functions:** Despliegue de únicamente las funciones.

3.3 Entorno de desarrollo integrado (IDE)

Como IDE para la edición del código se ha utilizado Visual Studio Code [7], por ser multiplataforma, relativamente ligero e incluir soporte para la depuración y el control integrado de Git. Además, está muy extendido, lo que se refleja en la gran comunidad activa que tiene en internet donde poder consultar dudas y problemas. Se ha instalado la extensión Vue Language Features (Volar) [8], que añade la sintaxis de Vue al IDE (Ilustración 5).

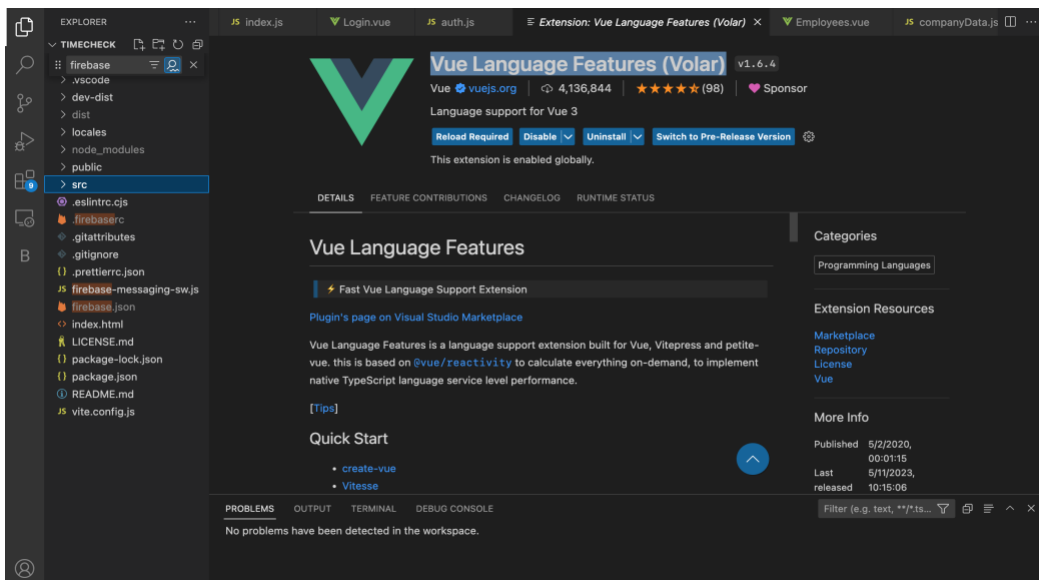


Ilustración 5 Visual Studio Code y VOLAR

De manera adicional se ha utilizado el IDE propio de Arduino [9] (Ilustración 6) para programar el dispositivo M5 Stick-CPlus [2]. Se ha optado por este programa por razones similares al anterior, multiplataforma, gran comunidad y soporte, alta compatibilidad con placas Arduino y buena gestión de *plugins* y bibliotecas externas. Este código se puede encontrar en el directorio de GitHub del proyecto, dentro de la carpeta “Arduino”.

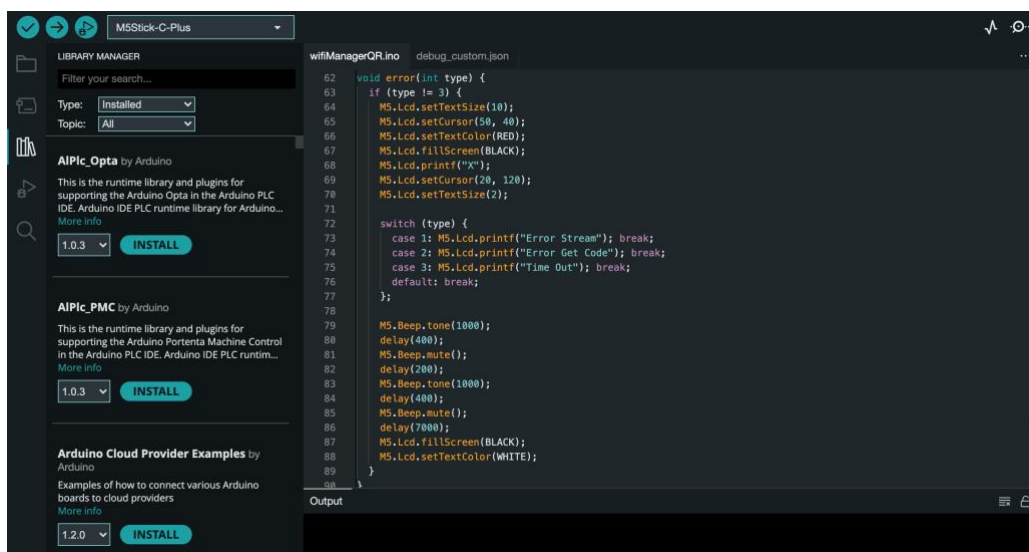


Ilustración 6 IDE Arduino

4 Bibliografía

- [1] Iván Ramírez, “¿Qué es una Aplicación Web Progresiva o PWA?,” <https://www.xataka.com/basics/que-es-una-aplicacion-web-progresiva-o-pwa>.
- [2] m5stack, “M5StickC PLUS ESP32-PICO Mini IoT Development Kit.” <https://shop.m5stack.com/products/m5stickc-plus-esp32-pico-mini-iot-development-kit> (accessed Apr. 27, 2023).
- [3] GitHub Inc, “GitHub.” Accessed: May 27, 2023. [Online]. Available: <https://github.com>
- [4] “JSDoc.” Accessed: Jun. 18, 2023. [Online]. Available: <https://jsdoc.app/index.html>
- [5] Zadhid Powell, “Cómo Instalar Node.js y npm en Windows, macOS y Linux.” Accessed: May 27, 2023. [Online]. Available: <https://kinsta.com/es/blog/como-instalar-node-js/#cmo-instalar-nodejs-en-macos>
- [6] Google for Developers, “Referencia de Firebase CLI.” Accessed: May 27, 2023. [Online]. Available: <https://firebase.google.com/docs/cli?hl=es-419>
- [7] Microsoft, “Visual Studio Code.” Accessed: May 27, 2023. [Online]. Available: <https://code.visualstudio.com>
- [8] StackBlitz, “Vue Language Features (Volar).” Accessed: May 27, 2023. [Online]. Available: <https://marketplace.visualstudio.com/items?itemName=Vue.volar>
- [9] Arduino, “Arduino IDE.” Accessed: May 27, 2023. [Online]. Available: <https://www.arduino.cc/en/software>