

MANUAL

de

Despliegue

Una guía sobre
el despliegue
del aplicativo.



1. PROCEDIMIENTO DE DESPLIEGUE (CI/CD)

El despliegue del aplicativo "CuidaDos" se basa en un modelo de Integración y Despliegue Continuo (CI/CD). Este enfoque automatiza el paso del código desde el entorno de desarrollo a producción, minimizando errores manuales y asegurando que la aplicación esté siempre disponible.

1.1. Infraestructura de Hosting: Vercel

Se ha seleccionado Vercel como plataforma de despliegue debido a su optimización nativa para aplicaciones Angular y su red de distribución de contenidos (Edge Network), que garantiza tiempos de respuesta mínimos.

1.2. Flujo de Trabajo y Automatización

El proceso de despliegue sigue un flujo estructurado:

1. Control de Versiones: El código se gestiona en un repositorio de GitHub.
2. Disparador (Trigger): Cada vez que se realiza un *push* a la rama principal (main), Vercel detecta el cambio e inicia automáticamente un nuevo *build*.
3. Proceso de Compilación (Build):
 - o Se instalan las dependencias mediante npm install.
 - o Se ejecuta el comando ng build, que compila TypeScript a JavaScript altamente optimizado, aplicando *tree-shaking* y *minificación*.
4. Generación de la PWA: Durante el build, se generan los Service Workers y el manifiesto necesarios para que la aplicación funcione como una Progressive Web App.

1.3. Configuraciones Específicas del Proyecto

Para garantizar el correcto funcionamiento en el entorno de producción, se realizaron los siguientes ajustes técnicos:

- Manejo de Rutas (vercel.json): Al ser una SPA (Single Page Application), se configuró una regla de reescritura de rutas. Esto evita errores 404 al recargar la página, redirigiendo todas las peticiones al index.html para que el enrutador de Angular tome el control.
- Root Directory: Se ajustó el directorio raíz en la configuración de Vercel para asegurar que la plataforma localice correctamente los archivos de salida dentro de la estructura de carpetas del proyecto.
- Variables de Entorno: Las claves de conexión con Supabase (URL y Anon Key) se han configurado como variables de entorno cifradas en el panel de Vercel, manteniendo la seguridad y evitando que datos sensibles se suban al repositorio público.

1.4. Sincronización con el Backend

El despliegue del frontend se complementa con la persistencia en Supabase. Al tratarse de una arquitectura *Serverless*, no es necesario desplegar un servidor físico; sin embargo, se verifica que:

- Las tablas y el esquema relacional en PostgreSQL estén actualizados.
- Las políticas de RLS (Row Level Security) estén activas para proteger los datos de los usuarios en el entorno real.

1.5 Resolución de Incidencias en el Despliegue

Durante el proceso de puesta en producción, se identificaron y resolvieron los siguientes conflictos técnicos para asegurar la disponibilidad del aplicativo:

1. **Error de Enrutamiento (404 en Refresh):** Al ser una Single Page Application (SPA), el servidor de Vercel no reconocía las rutas internas de Angular al recargar la página. Se solucionó mediante la creación de un archivo `vercel.json` en la raíz del proyecto, configurando una regla de *rewrites* para redirigir todas las peticiones al `index.html`.
2. **Conflicto con el Directorio Raíz (Root Directory):** El despliegue fallaba inicialmente porque la plataforma no localizaba correctamente los archivos compilados. Se procedió a ajustar el **Root Directory** en el panel de configuración de Vercel para que apuntara a la carpeta específica del proyecto, permitiendo que el comando `ng build` se ejecutara en el contexto correcto.
3. **Configuración de la Carpeta de Salida (Output Directory):** Se verificó y corrigió la ruta de salida de los archivos "traducidos" (compilados) para que Vercel pudiera servirlos correctamente desde la carpeta `dist`, asegurando que todos los activos (imágenes, fuentes y scripts) se cargaran sin errores.

1.6. Verificación Final

Tras cada despliegue, se realiza una validación en la URL pública (<https://tfg-cuidados.vercel.app/>), comprobando:

- La correcta carga de los Signals de Angular.
- La persistencia de la sesión de usuario.
- La respuesta de los servicios en menos de 200ms mediante las Chrome DevTools.

The screenshot shows a Vercel deployment page for the project 'CuidaDos'. At the top left is the project logo, which includes a heart icon and the text 'CuidaDos'. To the right of the logo are sections for 'Deployment', 'Domains', 'Status', and 'Source'. The 'Deployment' section shows the URL 'tfg-cuidados.vercel.app'. The 'Domains' section shows a single domain 'tfg-cuidados.vercel.app'. The 'Status' section indicates the app is 'Ready' and was created 4 minutes ago by user 'egilp04'. The 'Source' section shows a GitHub repository with the main branch checked out and a commit hash '82d68cf' that updated the README.md file.

Deployment
tfg-cuidados-a2def8q7r-eves-projects-b28db611.vercel.app

Domains [+](#)
tfg-cuidados.vercel.app [🔗](#)

Status Created
● Ready 4m ago by egilp04 [🔗](#)

Source
↳ main
-o 82d68cf Update README.md

> Deployment Settings [4 Recommendations](#)

Puedes clicar en la imagen para acceder al aplicativo desplegado.