

Guía Completa de Deploy

Método 1: GitHub Pages (Recomendado)

Paso 1: Preparar el proyecto

1. Edita `vite.config.js`:

```
javascript
base: '/nombre-exacto-de-tu-repo/',
```

2. Instala `gh-pages`:

```
bash
npm install --save-dev gh-pages
```

3. Añade scripts en `package.json`:

```
json
"scripts": {
  "predeploy": "npm run build",
  "deploy": "gh-pages -d dist"
}
```

Paso 2: Deploy

```
bash
# Compilar y desplegar
npm run deploy
```

Paso 3: Configurar GitHub

1. Ve a tu repositorio en GitHub
2. Settings > Pages
3. Source: **Deploy from a branch**
4. Branch: `gh-pages` / / (root)

5. Save

Tu sitio estará en: (<https://tu-usuario.github.io/nombre-repo/>)

Paso 4: Actualizar Firebase Config

Añade el dominio de GitHub Pages a Firebase:

1. Firebase Console > Authentication > Settings
 2. Authorized domains > Add domain
 3. Añade: (tu-usuario.github.io)
-

Método 2: Firebase Hosting

Paso 1: Instalar Firebase CLI

```
bash
npm install -g firebase-tools
firebase login
```

Paso 2: Inicializar Firebase Hosting

```
bash
firebase init hosting
```

Selecciona:

- Public directory: **dist**
- Single-page app: **Yes**
- GitHub auto-deploys: **No** (opcional)

Paso 3: Build y Deploy

```
bash
npm run build
firebase deploy --only hosting
```

Tu sitio estará en: (<https://tu-proyecto.web.app>)

Ventajas de Firebase Hosting:

- HTTPS automático
 - CDN global
 - Dominio personalizado gratis
 - Integración perfecta con Firebase
-

Método 3: Vercel

Paso 1: Instalar Vercel CLI

```
bash  
npm i -g vercel
```

Paso 2: Deploy

```
bash  
vercel
```

Sigue las instrucciones en pantalla.

Configuración automática:

Vercel detectará Vite automáticamente.

Método 4: Netlify

Opción A: Netlify CLI

```
bash  
npm install -g netlify-cli  
npm run build  
netlify deploy --prod --dir=dist
```

Opción B: Drag & Drop

1. Compila: `npm run build`

2. Ve a netlify.com/drop

3. Arrastra la carpeta `dist/`

Configuración de Variables de Entorno

GitHub Pages & Netlify & Vercel

Estas plataformas NO leen archivos `.env`. Debes:

Opción 1: Build con variables

```
bash
```

```
VITE_FIREBASE_API_KEY=xxx npm run build
```

Opción 2: Variables en la plataforma

- **Vercel:** Settings > Environment Variables
- **Netlify:** Site settings > Build & deploy > Environment

Opción 3: Hardcodear (no recomendado pero funciona) Edita `src/config.firebaseio.js` con valores directos.

Firebase Hosting

No necesita configuración especial, lee `.env` durante el build.

Troubleshooting

Error: "Page not found" en rutas

GitHub Pages: Crea `public/404.html`:

```
html
```

```
<!DOCTYPE html>
<html>
<head>
<script>
  sessionStorage.redirect = location.href;
</script>
<meta http-equiv="refresh" content="0;URL='/'>
</head>
</html>
```

Y en `[index.html]`, antes del `<script>`:

```
html

<script>
(function(){
  var redirect = sessionStorage.redirect;
  delete sessionStorage.redirect;
  if (redirect && redirect != location.href) {
    history.replaceState(null, null, redirect);
  }
})();
</script>
```

Firebase Hosting: Edita `[firebase.json]`:

```
json

{
  "hosting": {
    "public": "dist",
    "rewrites": [
      {
        "source": "**",
        "destination": "/index.html"
      }
    ]
  }
}
```

Firebase Auth no funciona

Añade tu dominio a Authorized domains:

PDFs no cargan (CORS)

Verifica las reglas de Storage. Deben permitir lectura pública.

Build falla

1. Borra `node_modules` y reinstala:

```
bash
```

```
rm -rf node_modules package-lock.json  
npm install
```

2. Verifica variables de entorno

3. Revisa la consola de errores

Comparativa de Plataformas

Plataforma	Gratis	HTTPS	Custom Domain	Deploy Time	CDN
GitHub Pages	✓	✓	✓	~2 min	✗
Firebase Hosting	✓	✓	✓	~1 min	✓
Vercel	✓	✓	✓	~30 seg	✓
Netlify	✓	✓	✓	~1 min	✓

Recomendación:

- **Firebase Hosting** si ya usas Firebase
- **Vercel/Netlify** para mejor performance
- **GitHub Pages** para proyectos sencillos

Actualizar después del primer deploy

GitHub Pages:

```
bash
```

```
npm run deploy
```

Firebase:

```
bash  
  
npm run build  
firebase deploy --only hosting
```

Vercel:

```
bash  
  
vercel --prod
```

Netlify:

```
bash  
  
netlify deploy --prod --dir=dist
```

Dominio personalizado

GitHub Pages:

1. Crea archivo `public/CNAME` con tu dominio
2. Configura DNS en tu proveedor:

```
A record: 185.199.108.153  
A record: 185.199.109.153  
A record: 185.199.110.153  
A record: 185.199.111.153
```

Firebase/Vercel/Netlify:

Sigue las instrucciones en el dashboard de cada plataforma.