🚀 Manual de Despliegue a Producción: FinDriver Pro

Proyecto: FinDriver Pro (Sistema de Gestión Financiera para Conductores)

Fecha de Implementación: 6 de Diciembre de 2025

Estado: ✅ En Producción (Live)

📑 Índice

---

1. Arquitectura del Despliegue

El sistema opera en dos nubes separadas que se comunican entre sí:

* **Frontend (La Web):** Alojado en **Vercel**. Es lo que ve el usuario.
  * URL: `https://findriver-app.vercel.app/`
* **Backend (El Cerebro):** Alojado en **Render**. Procesa datos y conecta con la base de datos.

    * URL: `https://findriver-app.onrender.com`
* **Base de Datos:** MongoDB Atlas (Nube).
* **Autenticación:** Firebase Auth (Google).

---

2. Paso 1: Preparación del Código (Git)

Antes de subir nada, aseguramos la integridad del código:

1. **Protección de Secretos:** Se creó un archivo `.gitignore` para evitar subir contraseñas (`.env`) o carpetas pesadas (`node_modules`) a GitHub.
2. **Limpieza de Código:**
    * Se corrigieron errores de sintaxis en `AddIncomeScreen.jsx` (código duplicado).
    * Se limpiaron advertencias en `OnboardingModal.jsx` y `RealTimeChart.jsx`.
3. **Repositorio:** Se subió todo el código limpio a GitHub: `docenciainformatica2025/findriver-app`.

---

3. Paso 2: Backend en Render

Configuramos el servidor para que "escuche" en internet:

1. **Servicio:** Web Service en Render (Node.js).
2. **Comandos de Construcción:**
    * Build: `npm install`

* Start: `node server.js`

3. **Variables de Entorno Clave (Environment Variables):**

* `NODE_ENV`: `production` (Desactiva herramientas de desarrollo).

* `FIREBASE_PRIVATE_KEY`: La "llave maestra" para hablar con Firebase Admin.

* `FRONTEND_URL`: `https://findriver-app.vercel.app` (Permite que la web se conecte al servidor sin bloqueo de seguridad CORS).

---

4. Paso 3: Frontend en Vercel

Publicamos la página web React/Vite:

1. **Proyecto:** Importado desde GitHub en Vercel.

2. **Framework:** Vite (Detección automática).

3. **Variables de Entorno (Client-Side):**

Se configuraron las claves públicas de Firebase para que el Login funcione:

* `VITE_API_URL`: Apunta al Backend (`https://findriver-app.onrender.com`).

* `VITE_FIREBASE_API_KEY`: Credencial de acceso.

* `VITE_FIREBASE_AUTH_DOMAIN`: Dominio de autenticación.

* `VITE_FIREBASE_PROJECT_ID`: Identificador del proyecto.

* *(Y resto de credenciales de Firebase Config)*.

---

5. Paso 4: Conexión y Seguridad

Para que el sistema funcione como uno solo:

1. **CORS (Cross-Origin Resource Sharing):**

   El Backend (Render) fue configurado explícitamente para aceptar peticiones SOLO desde el Frontend (Vercel). Esto evita que otros sitios usen tu API.

   * Config: `app.use(cors({ origin: process.env.FRONTEND_URL ... }))`

2. **Autenticación:**

   El Frontend envía un Token seguro a Render, y Render verifica ese token con Firebase antes de dar acceso a los datos.

---

6. Auditoría y Mantenimiento

Se realizó una auditoría final (`audit_report.md`) confirmando:

* ✅ **Sin Errores Críticos:** La aplicación carga y navega fluidamente.
* ✅ **Modo Producción:** El "Usuario Mock" de desarrollo está desactivado.
* ✅ **Logs Limpios:** Se eliminaron `console.log` innecesarios.

🔄 Para futuras actualizaciones:

1. Haz tus cambios en el código local.
2. Ejecuta: `git add .`, `git commit -m "Descripción"`, `git push`.
3. Vercel y Render detectarán el cambio y se actualizarán **automáticamente**.

Generado por Antigravity Agent para FinDriver Pro.