



# Guía de Despliegue en Coolify - EscalaFin

---



## Índice

---

1. [Pre-requisitos](#)
  2. [Configuración Inicial](#)
  3. [Despliegue Automático](#)
  4. [Configuración Manual](#)
  5. [Variables de Entorno](#)
  6. [Monitoreo y Logs](#)
  7. [Solución de Problemas](#)
- 



## Pre-requisitos

---

### Servidor Coolify

- ☒ Coolify instalado en `adm.escalafin.com`
- ☒ Acceso SSH al servidor
- ☒ Docker y Docker Compose funcionales
- ☒ Certificados SSL configurados

### Servicios Externos

- ☒ Base de datos PostgreSQL
- ☒ Cuenta AWS S3 para archivos
- ☒ Credenciales Openpay para pagos
- ☒ WhatsApp API (EvolutionAPI)

### Herramientas Locales

```
# Instalar dependencias necesarias
curl -fsSL https://get.docker.com -o get-docker.sh
sudo sh get-docker.sh

# Verificar instalación
docker --version
docker-compose --version
```

---

## Configuración Inicial

### Paso 1: Preparar el Proyecto

```
# Clonar el repositorio
git clone https://github.com/tu-usuario/escalafin-mvp.git
cd escalafin-mvp

# Dar permisos al script de despliegue
chmod +x deploy-coolify.sh

# Ejecutar configuración inicial
./deploy-coolify.sh setup
```

### Paso 2: Configurar Variables de Entorno

Edita el archivo `.env.coolify` generado:

```
# Variables críticas que DEBES configurar manualmente:

# Openpay (Pagos)
OPENPAY_MERCHANT_ID="tu_merchant_id_real"
OPENPAY_PRIVATE_KEY="tu_private_key_real"
OPENPAY_PUBLIC_KEY="tu_public_key_real"

# AWS S3 (Archivos)
AWS_ACCESS_KEY_ID="tu_access_key_real"
AWS_SECRET_ACCESS_KEY="tu_secret_access_key_real"
AWS_BUCKET_NAME="escalafin-production"

# WhatsApp API
EVOLUTION_API_URL="https://tu-evolution-api.com"
EVOLUTION_API_TOKEN="tu_token_real"
```

## Despliegue Automático

### Opción A: Script Automatizado

```
# Despliegue completo
./deploy-coolify.sh deploy

# Ver estado
./deploy-coolify.sh status

# Ver logs
./deploy-coolify.sh logs
```

### Opción B: Interfaz Web de Coolify

#### 1. Acceder a Coolify:

`https://adm.escalafin.com`

#### 2. Crear Nuevo Proyecto:

- Nombre: `escalafin-mvp`

- Tipo: Docker Compose
- Repositorio: Tu repo de GitHub

### 3. Configurar Servicios:

- App (Next.js)
- Database (PostgreSQL)
- Redis (Cache)

### 4. Configurar Dominio:

- Dominio principal: escalafin.com
- Alias: www.escalafin.com

---

## Configuración Manual

---

### 1. Configurar el Proyecto en Coolify

#### 1.1 Conectar Repositorio

```
# En la interfaz de Coolify:  
Source: GitHub  
Repository: tu-usuario/escalafin-mvp  
Branch: main  
Auto Deploy: Enabled
```

#### 1.2 Configurar Build

```
Build Method: Docker Compose  
Compose File: docker-compose.coolify.yml
```

#### 1.3 Configurar Servicios

##### Aplicación Principal:

```
Service: app  
Port: 3000  
Health Check: /api/health  
Memory: 1Gi  
CPU: 1000m
```

##### Base de Datos:

```
Service: db  
Image: postgres:14-alpine  
Persistent Storage: postgres_data  
Memory: 512Mi
```

##### Redis:

```

Service: redis
Image: redis:7-alpine
Persistent Storage: redis_data
Memory: 256Mi

```

## 2. Configurar Dominio y SSL

```

# En la configuración de Coolify:
Domain: escalafin.com
WWW Redirect: Enabled
SSL: Let's Encrypt (Auto)

```

## 3. Configurar Variables de Entorno

### 3.1 Variables de Aplicación

```

NODE_ENV=production
NEXTAUTH_URL=https://escalafin.com
NEXTAUTH_SECRET=[GENERAR_SECRETO_32_CHARS]

```

### 3.2 Variables de Base de Datos

```

DATABASE_URL=postgresql://escalafin:$@postgres_password@db:5432/escalafin
POSTGRES_USER=escalafin
POSTGRES_PASSWORD=[GENERAR_PASSWORD_SEGURA]
POSTGRES_DB=escalafin

```

### 3.3 Variables de Servicios Externos

```

# Openpay
OPENPAY_MERCHANT_ID=tu_merchant_id
OPENPAY_PRIVATE_KEY=tu_private_key
OPENPAY_PUBLIC_KEY=tu_public_key
OPENPAY_BASE_URL=https://api.openpay.mx/v1

# AWS S3
AWS_ACCESS_KEY_ID=tu_access_key
AWS_SECRET_ACCESS_KEY=tu_secret_key
AWS_BUCKET_NAME=escalafin-production
AWS_REGION=us-east-1
AWS_FOLDER_PREFIX=escalafin/

# WhatsApp
EVOLUTION_API_URL=https://tu-evolution-api.com
EVOLUTION_API_TOKEN=tu_token
EVOLUTION_INSTANCE_NAME=escalafin

```

---



## Variables de Entorno Completas

### Generar Secretos

```
# Generar NEXTAUTH_SECRET
openssl rand -base64 32

# Generar password de PostgreSQL
openssl rand -base64 32

# Generar password de Redis
openssl rand -base64 32
```

### Archivo .env.coolify Completo

```
# === CONFIGURACIÓN DE APLICACIÓN ===
NODE_ENV=production
NEXTAUTH_URL=https://escalafin.com
NEXTAUTH_SECRET=tu_nextauth_secret_de_32_caracteres_o_mas
NEXTAUTH_DEBUG=false

# === BASE DE DATOS ===
DATABASE_URL=postgresql://escalafin:${POSTGRES_PASSWORD}@db:5432/escalafin
POSTGRES_USER=escalafin
POSTGRES_PASSWORD=tu_postgres_password_super_segura
POSTGRES_DB=escalafin

# === REDIS ===
REDIS_PASSWORD=tu_redis_password_segura

# === OPENPAY (PAGOS) ===
OPENPAY_MERCHANT_ID=tu_merchant_id_de_openpay
OPENPAY_PRIVATE_KEY=tu_private_key_de_openpay
OPENPAY_PUBLIC_KEY=tu_public_key_de_openpay
OPENPAY_BASE_URL=https://api.openpay.mx/v1

# === AWS S3 (ARCHIVOS) ===
AWS_ACCESS_KEY_ID=tu_aws_access_key_id
AWS_SECRET_ACCESS_KEY=tu_aws_secret_access_key
AWS_BUCKET_NAME=escalafin-production
AWS_REGION=us-east-1
AWS_FOLDER_PREFIX=escalafin/

# === WHATSAPP API ===
EVOLUTION_API_URL=https://tu-evolution-api.com
EVOLUTION_API_TOKEN=tu_evolution_api_token
EVOLUTION_INSTANCE_NAME=escalafin

# === CONFIGURACIONES ADICIONALES ===
TZ=America/Mexico_City
PORT=3000
```



## Monitoreo y Logs

### Ver Logs en Tiempo Real

```
# Desde el script
./deploy-coolify.sh logs

# Desde Coolify web
https://adm.escalafin.com -> Proyecto -> Logs
```

### Métricas de Rendimiento

- CPU y Memoria por servicio
- Tiempo de respuesta de la aplicación
- Estado de salud de servicios
- Tráfico de red

### Alertas Configuradas

- Error 500 > 5% en 5 minutos
- Memoria > 80% por 10 minutos
- Base de datos desconectada
- Certificado SSL próximo a vencer



## Solución de Problemas

### Problemas Comunes

#### 1. Error de Construcción Docker

```
# Limpiar cache de Docker
docker system prune -a

# Reconstruir imagen
./deploy-coolify.sh deploy
```

#### 2. Variables de Entorno No Cargadas

```
# Verificar archivo .env.coolify
cat .env.coolify

# Recargar variables en Coolify
# Interface Web -> Settings -> Environment Variables -> Reload
```

#### 3. Base de Datos No Conecta

```
# Verificar estado del servicio PostgreSQL
# En Coolify: Services -> db -> Logs

# Verificar variables de conexión
DATABASE_URL=postgresql://escalafin:password@db:5432/escalafin
```

## 4. SSL No Funciona

```
# Verificar dominio DNS
nslookup escalafin.com

# Renovar certificado SSL
# En Coolify: Domains -> SSL -> Renew
```

## Logs de Diagnóstico

```
# Ver logs de aplicación
docker-compose -f docker-compose.coolify.yml logs app

# Ver logs de base de datos
docker-compose -f docker-compose.coolify.yml logs db

# Ver logs de Redis
docker-compose -f docker-compose.coolify.yml logs redis
```

## Comandos de Emergencia

```
# Rollback rápido
./deploy-coolify.sh rollback

# Reiniciar todos los servicios
# En Coolify: Actions -> Restart All Services

# Backup de emergencia de BD
pg_dump $DATABASE_URL > backup-emergency-$(date +%Y%m%d-%H%M%S).sql
```

---

## Verificación Post-Despliegue

---

### Checklist de Verificación

- [ ] ☒ Aplicación accesible en <https://escalafin.com>
- [ ] ☒ Login admin funcional
- [ ] ☒ Base de datos conectada
- [ ] ☒ Archivos S3 funcionando
- [ ] ☒ Pagos Openpay configurados
- [ ] ☒ WhatsApp API activa
- [ ] ☒ SSL válido y activo
- [ ] ☒ Redirects HTTP -> HTTPS
- [ ] ☒ Logs sin errores críticos

## URLs de Prueba

```
# Salud de la aplicación
curl https://escalafin.com/api/health

# Login admin
https://escalafin.com/admin

# API de usuarios
https://escalafin.com/api/admin/users
```

## Credenciales de Prueba

```
Admin:
  Email: admin@escalafin.com
  Password: admin123

Asesor:
  Email: asesor1@escalafin.com
  Password: asesor123

Cliente:
  Email: cliente1@example.com
  Password: cliente123
```

---

## Soporte

### Contacto Técnico

- **GitHub Issues:** [Crear Issue](https://github.com/tu-usuario/escalafin-mvp/issues) (<https://github.com/tu-usuario/escalafin-mvp/issues>)
- **Documentación:** Ver archivos MD en el repositorio
- **Logs:** Usar `./deploy-coolify.sh logs`

### Recursos Adicionales

- [Documentación Coolify](https://coolify.io/docs) (<https://coolify.io/docs>)
- [Docker Compose Reference](https://docs.docker.com/compose/) (<https://docs.docker.com/compose/>)
- [Next.js Deployment](https://nextjs.org/docs/deployment) (<https://nextjs.org/docs/deployment>)

---

 ¡Listo! Tu aplicación EscalaFin ya está desplegada en Coolify.

Para comenzar, accede a: **<https://escalafin.com>**