









Guía de Despliegue EasyPanel - EscalaFin 2025 (ACTUALIZADA)

Información del Despliegue

Esta guía actualizada incluye el despliegue de EscalaFin con todas las características más recientes, incluyendo el **sidebar navegacional sticky** y las mejoras de UI implementadas.

Características Incluidas en el Despliegue

-  **Sidebar Navegacional Sticky** responsive
-  **Navegación Móvil** optimizada
-  **Sistema de Módulos PWA** dinámico
-  **Dark/Light Theme** completo
-  **Layout Provider** centralizado
-  **AWS S3 Integration** para archivos
-  **OpenPay Integration** para pagos
-  **WhatsApp API** (EvolutionAPI)

Objetivos del Despliegue

1. **Desplegar** EscalaFin en EasyPanel
2. **Configurar** base de datos PostgreSQL
3. **Integrar** servicios externos (S3, OpenPay, WhatsApp)
4. **Configurar** SSL y dominio personalizado
5. **Implementar** monitoreo y backup automático

Prerrequisitos

Cuenta EasyPanel

- ☐ Cuenta EasyPanel activa
- ☐ Servidor VPS configurado (min 2GB RAM, 20GB storage)
- ☐ Dominio DNS configurado
- ☐ Certificado SSL preparado

Servicios Externos

- ☐ AWS Account con S3 bucket
 - ☐ OpenPay merchant account
 - ☐ EvolutionAPI instance
 - ☐ SMTP server (para emails)
-



Paso 1: Configuración Inicial EasyPanel

1.1 Crear Nueva Aplicación

```
# En EasyPanel Dashboard
1. Click "Create App"
2. Seleccionar "Node.js"
3. Nombre: "escalafin-mvp"
4. Port: 3000
5. Environment: Production
```

1.2 Configuración de Recursos

```
# Configuración de recursos mínimos
CPU: 1 vCPU
RAM: 2GB
Storage: 20GB
Node Version: 18.17+
Package Manager: Yarn
```



Paso 2: Configuración de Base de Datos

2.1 PostgreSQL Setup

```
# En EasyPanel Services
1. Click "Add Service"
2. Seleccionar "PostgreSQL 15"
3. Database name: escalafin_db
4. Username: escalafin_user
5. Password: [generar password seguro]
6. Port: 5432
```

2.2 Variables de Conexión

```
# Variables que EasyPanel genera automáticamente
DB_HOST=postgres-service-name
DB_PORT=5432
DB_NAME=escalafin_db
DB_USER=escalafin_user
DB_PASSWORD=generated-password

# Construir DATABASE_URL
DATABASE_URL=postgresql://escalafin_user:generated-password@postgres-service-name:5432/escalafin_db
```

Paso 3: Variables de Entorno

3.1 Configuración Completa (.env)

```
# Base de datos (Auto-generada por EasyPanel)
DATABASE_URL=postgresql://escalafin_user:password@postgres-host:5432/escalafin_db

# Aplicación
NODE_ENV=production
NEXT_PUBLIC_APP_NAME=EscalaFin
NEXT_PUBLIC_APP_VERSION=2.1.0
NEXT_PUBLIC_ENVIRONMENT=production

# Autenticación
NEXTAUTH_URL=https://escalafin.tu-dominio.com
NEXTAUTH_SECRET=super-secret-key-para-production-2025

# AWS S3 (Configurar en EasyPanel Secrets)
AWS_ACCESS_KEY_ID=AKIA...
AWS_SECRET_ACCESS_KEY=secret-key
AWS_BUCKET_NAME=escalafin-production
AWS_REGION=us-east-1
AWS_FOLDER_PREFIX=uploads/

# OpenPay
OPENPAY_ID=merchant-id
OPENPAY_PRIVATE_KEY=sk_private_key
OPENPAY_PUBLIC_KEY=pk_public_key
OPENPAY_PRODUCTION=true

# EvolutionAPI
EVOLUTION_API_URL=https://tu-instancia.evolutionapi.com
EVOLUTION_API_TOKEN=tu-api-token
EVOLUTION_INSTANCE=escalafin-prod

# SMTP (opcional)
SMTP_HOST=smtp.gmail.com
SMTP_PORT=587
SMTP_USER=noreply@tu-dominio.com
SMTP_PASSWORD=app-password

# Configuración adicional
NEXT_PUBLIC_DOMAIN=escalafin.tu-dominio.com
NEXT_PUBLIC_API_URL=https://escalafin.tu-dominio.com/api
```

3.2 Configuración de Secrets en EasyPanel

```
# En EasyPanel Secrets (Variables seguras)
1. AWS_ACCESS_KEY_ID: [tu-access-key]
2. AWS_SECRET_ACCESS_KEY: [tu-secret-key]
3. OPENPAY_PRIVATE_KEY: [private-key]
4. NEXTAUTH_SECRET: [super-secret-key]
5. DATABASE_URL: [connection-string]
```



Paso 4: Configuración de Despliegue

4.1 package.json Scripts (EasyPanel optimizado)

```
{
  "name": "escalafin-mvp",
  "private": true,
  "scripts": {
    "dev": "next dev",
    "build": "next build",
    "start": "next start -p $PORT",
    "lint": "next lint",
    "postinstall": "prisma generate",
    "deploy": "yarn build && yarn start"
  },
  "engines": {
    "node": ">=18.17.0",
    "yarn": ">=4.0.0"
  }
}
```

4.2 Dockerfile (Si se usa contenedor)

```
FROM node:18-alpine

WORKDIR /app

# Instalar dependencias del sistema
RUN apk add --no-cache postgresql-client

# Copiar archivos de dependencias
COPY app/package.json app/yarn.lock ./
RUN yarn install --frozen-lockfile

# Copiar código fuente
COPY app/ .

# Generar Prisma client
RUN yarn prisma generate

# Build de la aplicación
RUN yarn build

EXPOSE 3000

CMD ["yarn", "start"]
```

4.3 Scripts de Inicialización

```
#!/bin/bash
# deploy.sh - Script de despliegue

echo "🚀 Iniciando despliegue EscalaFin..."

# Instalar dependencias
echo "📦 Instalando dependencias..."
yarn install --frozen-lockfile

# Generar Prisma client
echo "📄 Generando Prisma client..."
yarn prisma generate

# Ejecutar migraciones
echo "📊 Aplicando migraciones de base de datos..."
yarn prisma db push

# Sembrar datos iniciales (solo primera vez)
if [ "$SEED_DATABASE" = "true" ]; then
  echo "🌱 Sembrando datos iniciales..."
  yarn prisma db seed
fi

# Build de la aplicación
echo "🔨 Construyendo aplicación..."
yarn build

echo "✅ Despliegue completado!"
```

Paso 5: Configuración de Dominio y SSL

5.1 Configuración DNS

```
# Configurar registros DNS
A      escalafin.tu-dominio.com    → IP-DEL-SERVIDOR
CNAME  www.escalafin.tu-dominio.com → escalafin.tu-dominio.com
```

5.2 SSL Certificate

```
# En EasyPanel SSL
1. Ir a App Settings → SSL
2. Seleccionar "Let's Encrypt"
3. Domain: escalafin.tu-dominio.com
4. Enable "Force HTTPS"
5. Enable "Auto-renewal"
```

5.3 Verificación de SSL

```
# Test SSL certificate
curl -I https://escalafin.tu-dominio.com
openssl s_client -connect escalafin.tu-dominio.com:443 -servername escalafin.tu-dominio.com
```



Paso 6: Inicialización de Base de Datos

6.1 Conexión a PostgreSQL

```
# Conectar a PostgreSQL desde EasyPanel terminal
psql postgresql://escalafin_user:password@postgres-host:5432/escalafin_db

# O usando las variables de entorno
psql $DATABASE_URL
```

6.2 Ejecutar Migraciones

```
# En el terminal de la app EasyPanel
cd /app
yarn prisma db push
yarn prisma db seed
```

6.3 Verificación de Tablas

```
-- Verificar estructura de la base de datos
\d
\d "User"
\d "Client"
\d "Loan"
\d "PWAModule"

-- Verificar datos iniciales
SELECT COUNT(*) FROM "User";
SELECT COUNT(*) FROM "PWAModule";
```

Paso 7: Configuración de Servicios Externos

7.1 AWS S3 Setup

```
# Crear bucket para producción
aws s3 mb s3://escalafin-production --region us-east-1

# Configurar CORS policy
aws s3api put-bucket-cors --bucket escalafin-production --cors-configuration '{
  "CORSRules": [
    {
      "AllowedOrigins": ["https://escalafin.tu-dominio.com"],
      "AllowedMethods": ["GET", "PUT", "POST", "DELETE"],
      "AllowedHeaders": ["*"],
      "MaxAgeSeconds": 3000
    }
  ]
}'
```

7.2 OpenPay Configuration

```
# Test OpenPay connection
curl -X GET "https://api.openpay.mx/v1/$OPENPAY_ID" \
  -u "$OPENPAY_PRIVATE_KEY:" \
  -H "Content-type: application/json"
```

7.3 EvolutionAPI Setup

```
# Test EvolutionAPI connection
curl -X GET "$EVOLUTION_API_URL/instance/$EVOLUTION_INSTANCE" \
  -H "Authorization: Bearer $EVOLUTION_API_TOKEN"
```

Paso 8: Optimizaciones EasyPanel

8.1 Performance Settings

```
# EasyPanel App Configuration
Memory Limit: 2GB
CPU Limit: 1000m
Restart Policy: Always
Health Check: /api/health
Health Check Interval: 30s
```

8.2 Build Optimizations

```
// next.config.js
module.exports = {
  output: 'standalone',
  experimental: {
    outputFileTracingRoot: '.',
  },
  compress: true,
  poweredByHeader: false,
  generateEtags: true,
  images: {
    domains: ['escala-fn-production.s3.amazonaws.com'],
    formats: ['image/webp', 'image/avif']
  }
}
```

8.3 Caching Strategy

```
# Configurar Redis cache (opcional)
# En EasyPanel Services → Add Redis
REDIS_URL=redis://redis-service:6379

# En la aplicación
NEXT_CACHE_HANDLER=redis
REDIS_CACHE_TTL=3600
```



Paso 9: Monitoreo y Logging

9.1 Health Checks

```
// app/api/health/route.ts
export async function GET() {
  try {
    // Test database connection
    await prisma.$queryRaw`SELECT 1`;

    // Test S3 connection
    await s3Client.send(new HeadBucketCommand({ Bucket: process.env.AWS_BUCKET_NAME }));

    return Response.json({
      status: 'healthy',
      timestamp: new Date().toISOString(),
      services: {
        database: 'up',
        s3: 'up',
        sidebar: 'active',
        modules: 'loaded'
      }
    });
  } catch (error) {
    return Response.json({ status: 'unhealthy' }, { status: 500 });
  }
}
```


9.2 Logging Configuration

```
# Logs en EasyPanel
1. Application Logs: Auto-collected
2. Error Logs: Streamed to dashboard
3. Custom Metrics: Via /api/metrics endpoint
4. Performance: Built-in monitoring
```

9.3 Alerts Setup

```
# EasyPanel Alerts
Resource Usage Alert: >80% CPU/Memory
Health Check Failure: 3 consecutive failures
SSL Expiry: 30 days before expiration
Database Connection: On failure
```



Paso 10: Despliegue y Validación

10.1 Despliegue Final

```
# En EasyPanel
1. Git Push → Auto-deploy activado
2. Manual Deploy → Build & Deploy
3. Zero-downtime deployment habilitado
4. Rollback automático en caso de error
```

10.2 Post-Deploy Checklist

- [] ☒ Aplicación responde en HTTPS
- [] ☒ Base de datos conectada
- [] ☒ SSL certificate válido
- [] ☒ Health checks passing
- [] ☒ Sidebar navegación funciona
- [] ☒ Módulos PWA cargan correctamente
- [] ☒ Autenticación operativa
- [] ☒ S3 uploads funcionan
- [] ☒ OpenPay integración activa
- [] ☒ WhatsApp notifications operativas

10.3 Testing URLs

```
# URLs críticas para probar
https://escalafin.tu-dominio.com/
https://escalafin.tu-dominio.com/api/health
https://escalafin.tu-dominio.com/auth/login
https://escalafin.tu-dominio.com/admin/dashboard
https://escalafin.tu-dominio.com/clients
https://escalafin.tu-dominio.com/admin/modules
```

Paso 11: Backup y Maintenance

11.1 Backup Automatizado

```
# Configurar backup diario en EasyPanel
1. Database Backup: Automático cada 24h
2. File System Backup: Weekly snapshots
3. S3 Sync: Diario para archivos críticos
4. Configuration Backup: Mensual
```

11.2 Scripts de Maintenance

```
#!/bin/bash
# maintenance.sh

echo "🔧 Ejecutando mantenimiento..."

# Limpiar logs antiguos
find /app/logs -name "*.log" -mtime +7 -delete

# Optimizar base de datos
psql $DATABASE_URL -c "VACUUM ANALYZE;"

# Verificar integridad S3
aws s3 ls s3://escalafin-production --recursive --summarize

echo "✅ Mantenimiento completado!"
```

11.3 Monitoring Dashboard

```
// Métricas a monitorear
const metricas = {
  response_time: '<200ms',
  uptime: '>99.9%',
  memory_usage: '<80%',
  cpu_usage: '<70%',
  database_connections: '<80% pool',
  s3_upload_success: '>99%',
  sidebar_load_time: '<100ms'
};
```

Troubleshooting

Problemas Comunes

1. Build Failures

```
# Error: Module not found
Solution: yarn install --frozen-lockfile

# Error: Prisma client not generated
Solution: yarn prisma generate

# Error: Next.js build timeout
Solution: Aumentar memoria en EasyPanel settings
```

2. Database Issues

```
# Error: Connection refused
Solution: Verificar DATABASE_URL y firewall rules

# Error: Migration failed
Solution: yarn prisma db push --force-reset
```

3. SSL Certificate Issues

```
# Error: Certificate not valid
Solution: Renovar en EasyPanel SSL settings

# Error: Mixed content warnings
Solution: Verificar NEXTAUTH_URL usa HTTPS
```





4. Sidebar/Navigation Issues

```
# Error: Sidebar not showing
Solution: Verificar módulos PWA están habilitados en DB



# Error: Mobile navigation broken
Solution: Verificar responsive breakpoints en CSS
```

Soporte EasyPanel

Recursos de Ayuda

-  **Documentación:** <https://easypanel.io/docs>
-  **Community:** Discord/Slack channels
-  **Tutorials:** YouTube channel
-  **Support:** support@easypanel.io






Información de Contacto

-  **Emergency:** emergency@escalafin.com
-  **Technical:** devops@escalafin.com







-  **Business:** admin@escalafin.com

Checklist Final de Despliegue





Infraestructura

- []  EasyPanel app configurada
- []  PostgreSQL database operativa
- []  SSL certificate instalado
- []  Dominio DNS configurado
- []  Variables de entorno establecidas





Aplicación

- []  Build exitoso sin errores
- []  Sidebar navegación funciona
- []  Navegación móvil responsive
- []  Módulos PWA cargan dinámicamente
- []  Autenticación multi-rol operativa
- []  Dark/Light theme funciona





Servicios

- []  AWS S3 uploads activos
- []  OpenPay pagos funcionando
- []  WhatsApp API conectada
- []  Email notifications operativas

Monitoreo

- []  Health checks configurados
- []  Logging funcionando
- []  Alerts configuradas
- []  Backup automatizado activo

Performance

- []  Tiempo de carga < 3 segundos
- []  Sidebar responsive < 100ms
- []  Navigation fluida
- []  Sin errores en consola