

✓ SOLUCIÓN: App Corriendo pero No Visible

Diagnóstico: El contenedor está corriendo correctamente (✓ Ready in 162ms) pero no es visible desde el navegador.

🔍 ANÁLISIS DE TUS LOGS

✓ Todo Está Funcionando Internamente

```
✓ Database schema is up to date!
✓ Base de datos ya tiene usuarios
✓ server.js encontrado en /app/server.js
✓ Next.js iniciado correctamente:
  - Local:      http://localhost:3000
  - Network:    http://0.0.0.0:3000
✓ Ready in 162ms
```

Conclusión: La aplicación está **100% funcional** dentro del contenedor.

✗ El Problema

No puedes acceder desde:

- `https://tu-dominio.com`
- `http://tu-ip:puerto`

Razón: Falta configuración de red/puertos en EasyPanel.

🎯 SOLUCIÓN: Configuración de Red en EasyPanel

Paso 1: Verificar Configuración de Puertos

1. Ve a tu app en EasyPanel
2. Click en **“Settings”** o **“Configuración”**
3. Busca sección **“Network”**, **“Ports”**, o **“Red”**

Debe verse así:

PUERTOS / PORTS
Container Port: 3000
Protocol: HTTP
Public: ✓ (checkbox marcado)
Domain: tu-dominio.com (opcional)

Paso 2: Agregar/Verificar Puerto

Si no ves ninguna configuración de puerto, agrégala:

Opción A: Con Dominio

```
Container Port: 3000
Protocol: HTTP
Enable Public Access: ✓
Domain: tu-dominio.com
SSL/TLS: ✓ (si tienes certificado)
```

Opción B: Sin Dominio (acceso por IP)

```
Container Port: 3000
Protocol: HTTP
Public Port: 80 (o cualquier puerto disponible)
Enable Public Access: ✓
```

Paso 3: Guardar y Aplicar

1. Click en **“Save”** o **“Guardar”**
2. Puede que necesite **“Restart”** o **“Reiniciar”** el servicio
3. Esperar 10-30 segundos



CONFIGURACIÓN DE DOMINIO

Si usas un dominio personalizado:

1. En EasyPanel

```
Domain Settings:
├ Domain: escalafin.tu-dominio.com
├ Enable SSL: ✓
└ Auto-renew certificate: ✓
```

2. En tu Proveedor de DNS

Agregar registro A o CNAME:

Registro A:

```
Type: A
Name: escalafin (o @ para root domain)
Value: [IP del servidor de EasyPanel]
TTL: 3600 (o Auto)
```

Registro CNAME:

```
Type: CNAME
Name: escalafin
Value: [hostname proporcionado por EasyPanel]
TTL: 3600 (o Auto)
```



Tiempo de propagación: 5 minutos a 24 horas (usualmente 15-30 min)

VERIFICACIÓN INMEDIATA

Test 1: Health Check Interno

En EasyPanel, ir a “**Console**” o “**Terminal**” del contenedor y ejecutar:

```
curl http://localhost:3000/api/health
```

Resultado esperado:

```
{"status": "ok"}
```

Si funciona →  La app está bien, el problema es de red.

Test 2: Verificar Puertos Abiertos

En el terminal del servidor (no del contenedor):

```
# Listar puertos en uso
netstat -tulpn | grep 3000

# 0 con ss
ss -tulpn | grep 3000
```

Debe mostrar: Algo escuchando en puerto 3000.

Test 3: Desde tu Computadora

```
# Probar acceso directo al puerto (reemplaza IP_DEL_SERVIDOR)
curl http://IP_DEL_SERVIDOR:3000/api/health

# 0 probar el dominio
curl https://tu-dominio.com/api/health
```

SOLUCIONES COMUNES

Problema 1: Puerto No Expuesto

Síntoma: Contenedor corriendo pero no hay forma de acceder.

Solución:

1. EasyPanel → Settings → Network/Ports
2. Agregar puerto 3000
3. Marcar “Public” o “Enable Public Access”
4. Guardar y reiniciar

Problema 2: Firewall Bloqueando

Síntoma: El puerto está expuesto pero no responde.

Solución: Verificar firewall del servidor:

```
# En Ubuntu/Debian
sudo ufw status
sudo ufw allow 3000/tcp

# En CentOS/RHEL
sudo firewall-cmd --list-all
sudo firewall-cmd --add-port=3000/tcp --permanent
sudo firewall-cmd --reload
```

Problema 3: Proxy Reverso No Configurado

Síntoma: Funciona por IP:puerto pero no por dominio.

Solución: En EasyPanel:

1. Settings → Domain/Proxy
2. Agregar configuración de proxy reverso:

Source: tu-dominio.com

Target: localhost:3000

SSL: Enable

Problema 4: DNS No Propagado

Síntoma: El dominio no resuelve o da timeout.

Verificar:

```
# Verificar que el dominio resuelve
nslookup tu-dominio.com
dig tu-dominio.com

# Verificar hacia dónde apunta
ping tu-dominio.com
```

Solución: Esperar propagación DNS (15-30 min usualmente).

Problema 5: SSL/Certificado

Síntoma: Funciona con HTTP pero no HTTPS.

Solución:

1. EasyPanel → Settings → SSL
2. Enable “Auto SSL” o “Let’s Encrypt”
3. Generar certificado
4. Esperar 1-2 minutos



CHECKLIST DE VERIFICACIÓN

En EasyPanel

- ☐ ☒ Contenedor está corriendo (status: running)
- ☐ ☒ Logs muestran “✓ Ready in XXXms”
- ☐ ☒ Puerto 3000 está configurado en Settings
- ☐ ☒ “Public Access” está habilitado

- ☐ ☒ Dominio configurado (si aplica)
- ☐ ☒ SSL habilitado (si aplica)

En tu Proveedor de DNS (si usas dominio)

- ☐ ☒ Registro A o CNAME creado
- ☐ ☒ Apuntando a IP correcta
- ☐ ☒ DNS propagado (verificar con nslookup)

Pruebas Funcionales

- ☐ ☒ `curl http://localhost:3000/api/health` funciona (desde contenedor)
- ☐ ☒ `curl http://IP_SERVIDOR:3000/api/health` funciona (desde afuera)
- ☐ ☒ `curl https://tu-dominio.com/api/health` funciona
- ☐ ☒ Navegador muestra la página de login



CONFIGURACIÓN RECOMENDADA

Para Producción con Dominio

```
Settings → Network:
  Container Port: 3000
  Protocol: HTTP
  Public: Yes

Settings → Domain:
  Domain: escalafin.tu-dominio.com
  SSL: Auto (Let's Encrypt)
  Force HTTPS: Yes

Settings → Environment:
  NEXTAUTH_URL: https://escalafin.tu-dominio.com
  PORT: 3000
  HOSTNAME: 0.0.0.0
  NODE_ENV: production
```

Para Testing sin Dominio

```
Settings → Network:
  Container Port: 3000
  Public Port: 80
  Protocol: HTTP
  Public: Yes

Settings → Environment:
  NEXTAUTH_URL: http://IP_DEL_SERVIDOR
  PORT: 3000
  HOSTNAME: 0.0.0.0
  NODE_ENV: production
```

ACCIONES INMEDIATAS

1. Verificar Configuración Actual

Toma screenshot de:

- Settings → Network/Ports
- Settings → Domain (si tienes)
- Status del contenedor

2. Aplicar Configuración Mínima

Si no tienes puerto configurado:

1. Settings → Network → Add Port
2. **C**ontainer Port: 3000
3. Public: ☒
4. **S**ave
5. Restart **c**ontainer

3. Probar Acceso

```
# Opción 1: Por IP directa
http://[IP_DEL_SERVIDOR]:3000

# Opción 2: Por dominio (si configuraste)
https://tu-dominio.com

# Opción 3: Health check
curl http://[IP_O_DOMINIO]/api/health
```

4. Si Aún No Funciona

Comparte:

1. Screenshot de configuración de puertos
2. Resultado de: `curl http://localhost:3000/api/health` (ejecutado desde el contenedor)
3. Resultado de: `curl http://IP_SERVIDOR:3000/api/health` (desde tu computadora)
4. Screenshot del browser mostrando el error

COMANDOS DE DIAGNÓSTICO

Ejecutar desde el Terminal del Contenedor

```
# Test interno
curl http://localhost:3000/api/health

# Ver procesos
ps aux | grep node

# Ver puertos escuchando
netstat -tulpn | grep 3000

# Ver variables de entorno
env | grep -E "(PORT|HOSTNAME|NEXTAUTH)"
```

Ejecutar desde el Servidor (fuera del contenedor)

```
# Ver contenedores corriendo
docker ps | grep escalafin

# Ver logs en tiempo real
docker logs -f [CONTAINER_ID]

# Ver puertos mapeados
docker port [CONTAINER_ID]

# Probar conexión al contenedor
curl http://localhost:3000/api/health
```



LO MÁS PROBABLE

Basado en que tu app está corriendo perfectamente (✓ Ready in 162ms), el problema es **99% seguro** uno de estos:

1. **Puerto no expuesto públicamente** en EasyPanel (70% probabilidad)
2. **Dominio no configurado o DNS no propagado** (20% probabilidad)
3. **Firewall bloqueando el puerto** (10% probabilidad)

Solución más rápida:

1. EasyPanel → Settings → Network
2. Agregar puerto 3000 público
3. Save y Restart
4. Probar: `http://IP_SERVIDOR:3000`



RESULTADO ESPERADO

Una vez configurado correctamente, deberías poder:

```
# Health check exitoso
$ curl https://tu-dominio.com/api/health
{"status":"ok"}

# Login visible en navegador
$ curl -I https://tu-dominio.com/auth/login
HTTP/2 200
content-type: text/html
```

Y en el navegador ver:

- Página de login de EscalaFin
- Estilos cargados
- Sin errores en consola
- Formulario de login funcional

Estado: El contenedor está funcionando perfectamente. Solo necesita configuración de red en EasyPanel.

Próximo paso: Configurar puerto 3000 como público en Settings → Network