

ANÁLISIS: ¿Qué pasa si se elimina Dockerfile Demo?

Fecha: 29 de Octubre 2025

Archivo a evaluar: `/instances/demo/Dockerfile`

UBICACIÓN Y CONTEXTO

```
escalafin_mvp/
├── Dockerfile                ← Dockerfile RAÍZ (producción)
├── instances/
│   └── demo/
│       ├── Dockerfile        ← Dockerfile DEMO (a evaluar)
│       ├── docker-compose.yml ← USA este Dockerfile
│       ├── deploy-demo.sh
│       └── ...otros archivos demo...
```

PROPÓSITO DE `/instances/demo/`

La carpeta `/instances/demo/` es parte de un **sistema multi-instancia** que permite:

1. **Deploys independientes** de la misma aplicación
2. **Configuraciones específicas** por instancia (demo, staging, producción)
3. **Aislamiento** entre ambientes
4. **Deploy con Coolify** (según archivos encontrados)




Evidencia:

```
./instances/demo/deploy-demo.sh
./instances/demo/deploy-manual.sh
./instances/demo/INSTRUCCIONES_DESPLIEGUE_MANUAL.md
./instances/demo/docker-compose.yml
```



IMPACTO DE ELIMINAR `instances/demo/Dockerfile`

LO QUE NO SE ROMPERÁ



1. Producción Principal

-  El Dockerfile raíz (`/Dockerfile`) es **independiente**
-  EasyPanel puede seguir usando el Dockerfile raíz
-  Deploy principal NO afectado

2. GitHub/Repositorio

-  El repositorio principal sigue funcional
-  Solo se elimina un archivo de una subcarpeta

3. Desarrollo Local (app/)

-  La carpeta `app/` no depende del Dockerfile Demo
-  Desarrollo local sigue igual

LO QUE SÍ SE ROMPERÁ

1. Docker Compose en `/instances/demo/`

Archivo afectado: `instances/demo/docker-compose.yml`

```
services:
  app:
    build:
      context: .
      dockerfile: Dockerfile # ← BUSCA ESTE ARCHIVO
```

Error esperado:

```
ERROR: Cannot locate specified Dockerfile: Dockerfile
```

Impacto: No se podrá hacer `docker-compose up` en esa carpeta.

2. Scripts de Deploy de Instancia Demo

Archivos afectados:

- `instances/demo/deploy-demo.sh`
- `instances/demo/deploy-manual.sh`

Impacto: Los scripts que despliegan la instancia demo fallarán al intentar construir la imagen.

3. Documentación Referenciada

Archivos con referencias:

- `COMPARACION_DOCKERFILES_29_OCT_2025.md` (recién creado)
- `COMO_DESCARGAR_INSTANCIA_DEMO.md`
- `FIX_SYMLINKS_29_OCT_2025.md`

Impacto: Las referencias quedarán desactualizadas.

🤔 ESCENARIOS DE USO

Escenario 1: Solo usas EasyPanel con Dockerfile raíz

```
└─ Dockerfile (raíz)
   └─ EasyPanel → ✅ USA ESTE
```

Impacto de eliminar Demo: ✅ **CERO** - No lo usas

Escenario 2: Usas sistema multi-instancia Coolify

```
├─ instances/
│  └─ demo/
│     └─ Dockerfile → Coolify Demo
│  └─ staging/
│     └─ Dockerfile → Coolify Staging
│  └─ prod/
│     └─ Dockerfile → Coolify Prod
```

Impacto de eliminar Demo: ❌ **ALTO** - Rompe deploy de instancia demo

Escenario 3: Desarrollo local con docker-compose

```
cd instances/demo/
docker-compose up
```

Impacto de eliminar Demo: ❌ **MEDIO** - No podrás levantar ambiente demo local



ALTERNATIVAS SI SE ELIMINA

Opción A: Usar Dockerfile Raíz en docker-compose

Modificar `instances/demo/docker-compose.yml` :

```
services:
  app:
    build:
      context: ../../          # ← Cambiar a raíz del proyecto
      dockerfile: Dockerfile   # ← Usa el Dockerfile raíz
```

Ventaja: Reutiliza el Dockerfile optimizado

Desventaja: El context cambia (requiere ajustes)

Opción B: Crear Symlink

```
cd /home/ubuntu/escalafin_mvp/instances/demo/
ln -s ../../Dockerfile Dockerfile
```

Ventaja: Reutiliza automáticamente el Dockerfile raíz

Desventaja: Symlinks pueden causar problemas en Git/Docker

Opción C: Eliminar toda la carpeta /instances/demo/

```
rm -rf /home/ubuntu/escalafin_mvp/instances/demo/
```

Ventaja: Simplifica estructura

Desventaja: Pierdes configuración específica de demo



ANÁLISIS DE REFERENCIAS

Referencias encontradas: 18

```
# Archivos que mencionan instances/demo
./instances/demo/deploy-demo.sh
./instances/demo/deploy-manual.sh
./instances/demo/docker-compose.yml          # ← CRÍTICO
./COMPARACION_DOCKERFILES_29_OCT_2025.md
./COMO_DESCARGAR_INSTANCIA_DEMO.md
./FIX_SYMLINKS_29_OCT_2025.md
...y 12 más
```



RECOMENDACIONES



ELIMINAR si:

1. ☒ Solo usas EasyPanel con Dockerfile raíz
2. ☒ No tienes sistema multi-instancia activo
3. ☒ No necesitas deploy de instancia demo independiente
4. ☒ Quieres simplificar la estructura del proyecto

Pasos seguros:

```
# 1. Backup primero
cp -r instances/demo instances/demo.backup

# 2. Eliminar
rm -rf instances/demo/

# 3. Commit
git add .
git commit -m "chore: Remove unused demo instance"
git push origin main
```

⚠ NO ELIMINAR si:

1. ⚠ Usas Coolify para deploys multi-instancia
2. ⚠ Alguien más del equipo usa `instances/demo/`
3. ⚠ Tienes CI/CD configurado que lo use
4. ⚠ Necesitas ambientes aislados (demo, staging, etc.)

Alternativa: Actualizar el Dockerfile Demo para que use el mismo enfoque que el Raíz

MIGRACIÓN RECOMENDADA

En lugar de eliminar, **MIGRAR** el Dockerfile Demo para usar el Raíz:

Paso 1: Modificar docker-compose.yml

```
# instances/demo/docker-compose.yml
services:
  app:
    build:
      context: ../../      # Apuntar a raíz
      dockerfile: Dockerfile # Usar Dockerfile raíz
    # ... resto igual
```

Paso 2: Eliminar Dockerfile Demo

```
rm instances/demo/Dockerfile
```

Paso 3: Verificar

```
cd instances/demo
docker-compose build # Debería usar Dockerfile raíz
```



DECISIÓN SUGERIDA

Para tu caso específico:

Dado que:

- ☒ Estás usando EasyPanel
- ☒ Ya tienes Dockerfile raíz optimizado
- ☐ ? No está claro si usas sistema multi-instancia activamente

Recomendación:

Opción 1: Simplificar (RECOMENDADO)

```
# Eliminar toda la carpeta instances/demo/
rm -rf instances/demo/

# Usar solo el Dockerfile raíz para todo
# EasyPanel → Dockerfile raíz
# Desarrollo local → Dockerfile raíz con docker-compose en raíz
```

Ventajas:

- ☒ Estructura más simple
- ☒ Un solo Dockerfile que mantener
- ☒ Menos confusión

Opción 2: Mantener pero unificar

```
# Hacer que instances/demo/ use el Dockerfile raíz
# Modificar docker-compose.yml para apuntar a raíz
```

Ventajas:

- ☒ Mantiene separación de ambientes
- ☒ Usa Dockerfile optimizado



CONCLUSIÓN

¿Se puede eliminar seguramente?

Sí, si:

- ☒ Solo usas EasyPanel (ya configurado con Dockerfile raíz)
- ☒ No dependes de `instances/demo/` para desarrollo local
- ☒ No tienes sistema multi-instancia en producción

¿Qué se rompe?

- ☒ `docker-compose up` en `instances/demo/` fallará
- ☒ Scripts de deploy específicos de demo fallarán
- ☒ Referencias en documentación quedarán obsoletas

¿Vale la pena?

SÍ - Simplifica la estructura y evita confusión entre dos Dockerfiles diferentes.



ACCIÓN RECOMENDADA

```
# 1. Backup de seguridad
tar -czf instances-demo-backup-$(date +%Y%m%d).tar.gz instances/demo/

# 2. Eliminar carpeta demo
rm -rf instances/demo/

# 3. Commit
git add .
git commit -m "chore: Remove demo instance, use root Dockerfile only"
git push origin main

# 4. Actualizar EasyPanel
# Build Context: /
# Dockerfile Path: ./Dockerfile
```

Documentado por: DeepAgent

Fecha: 29 de Octubre 2025

Versión: 1.0