

# Docker Desktop no Windows - Guia Completo

## Compatibilidade

**Boa notícia:** Todos os arquivos Docker (Dockerfile, docker-compose.yml, nginx.conf) são 100% compatíveis com Windows!

### O que muda:

- Scripts `.sh` não funcionam nativamente no Windows
- Use scripts `.bat` ou PowerShell
- Caminhos de arquivo funcionam igual
- Line endings (CRLF vs LF) podem causar problemas

## Configuração Inicial no Windows

### 1. Pré-requisitos

#### Docker Desktop instalado e rodando

- Download: <https://www.docker.com/products/docker-desktop/>
- Configuração mínima: 4GB RAM, 20GB disco

#### WSL2 habilitado (Docker Desktop usa WSL2)

```
powershell  
# Executar como Administrador no PowerShell  
wsl --install
```

### 2. Estrutura de Pastas (Igual para Windows)

```
C:\iptv-plataforma\  
|   └── backend\  
|       ├── Dockerfile  
|       ├── package.json  
|       ├── server.js  
|       └── ...  
|   └── frontend\  
|       ├── Dockerfile  
|       ├── nginx.conf  
|       ├── package.json  
|       └── ...  
└── nginx\
```

```
|   └── nginx.conf
|   ├── docker-compose.yml
|   ├── .env
|   ├── .gitignore
|   └── scripts\      # ← Scripts Windows
|       ├── start.bat
|       ├── stop.bat
|       └── logs.bat
```

### 3. Criar Arquivo .env (Windows)

#### Opção 1: Via PowerShell

```
powershell

# Navegar para a pasta do projeto
cd C:\iptv-plataforma

# Criar .env
New-Item -ItemType File -Path .env

# Editar com Notepad
notepad .env
```

#### Opção 2: Via CMD

```
cmd

cd C:\iptv-plataforma
type nul > .env
notepad .env
```

#### Conteúdo do .env:

```
env
```

```
# Supabase
SUPABASE_URL=https://seu-projeto.supabase.co
SUPABASE_SERVICE_KEY=sua_service_key
SUPABASE_ANON_KEY=sua_anon_key
```

```
# Mercado Pago
MP_ACCESS_TOKEN=seu_access_token
MP_PUBLIC_KEY=sua_public_key
```

```
# Email
EMAIL_HOST=smtp.gmail.com
EMAIL_PORT=587
EMAIL_USER=seu-email@gmail.com
EMAIL_PASS=sua-senha-de-app
```

```
# URLs (localhost funciona no Windows)
FRONTEND_URL=http://localhost:3000
BACKEND_URL=http://localhost:3001
```

## Scripts Windows (.bat)

### **start.bat**

```
batch
```

```
@echo off  
echo =====  
echo Iniciando Plataforma IPTV  
echo =====  
echo.
```

```
REM Verificar se Docker está rodando  
docker info >nul 2>&1  
if errorlevel 1 (  
    echo [ERRO] Docker Desktop nao esta rodando!  
    echo Por favor, inicie o Docker Desktop e tente novamente.  
    pause  
    exit /b 1  
)
```

```
REM Verificar se .env existe  
if not exist .env (  
    echo [ERRO] Arquivo .env nao encontrado!  
    echo Copie o .env.example e configure suas variaveis.  
    pause  
    exit /b 1  
)
```

```
echo [1/3] Construindo imagens Docker...  
docker-compose build
```

```
echo.  
echo [2/3] Iniciando containers...  
docker-compose up -d
```

```
echo.  
echo [3/3] Aguardando servicos iniciarem...  
timeout /t 10 /nobreak >nul
```

```
echo.  
echo =====  
echo Status dos Servicos  
echo =====  
docker-compose ps
```

```
echo.  
echo =====  
echo Aplicacao Iniciada!  
echo =====  
echo.  
echo Frontend: http://localhost:3000
```

```
echo Backend: http://localhost:3001
echo.
echo Pressione qualquer tecla para sair...
pause >nul
```

## stop.bat

```
batch

@echo off
echo =====
echo Parando Plataforma IPTV
echo =====
echo.

docker-compose down

echo.
echo Aplicacao parada com sucesso!
echo.
pause
```

## restart.bat

```
batch

@echo off
echo =====
echo Reiniciando Plataforma IPTV
echo =====
echo.

docker-compose down
timeout /t 3 /nobreak >nul
docker-compose up -d

echo.
echo Aplicacao reiniciada!
echo Use "logs.bat" para ver os logs
echo.
pause
```

## logs.bat

```
batch
```

```
@echo off
if "%1"==""
    echo Mostrando logs de todos os servicos...
    echo Pressione Ctrl+C para sair
    echo.
    docker-compose logs -f --tail=100
) else (
    echo Mostrando logs do %1...
    echo Pressione Ctrl+C para sair
    echo.
    docker-compose logs -f %1 --tail=100
)
```

## status.bat

```
batch
```

```
@echo off
echo =====
echo Status da Plataforma IPTV
echo =====
echo.

echo [1] Docker Status:
docker info >nul 2>&1
if errorlevel 1 (
    echo [X] Docker: Nao esta rodando
    pause
    exit /b 1
) else (
    echo [OK] Docker: Rodando
)

echo.

echo [2] Containers:
docker-compose ps

echo.

echo [3] Uso de Recursos:
docker stats --no-stream --format "table {{.Name}}\t{{.CPUPerc}}\t{{.MemUsage}}"

echo.

echo [4] Health Checks:
curl -s http://localhost:3001/health >nul 2>&1
if errorlevel 1 (
    echo [X] Backend: Offline
) else (
    echo [OK] Backend: Online
)

curl -s http://localhost:3000 >nul 2>&1
if errorlevel 1 (
    echo [X] Frontend: Offline
) else (
    echo [OK] Frontend: Online
)

echo.
pause
```

## clean.bat

batch

```
@echo off
echo =====
echo Limpeza Docker
echo =====
echo.
echo ATENCAO: Isso ira remover:
echo - Containers parados
echo - Imagens nao utilizadas
echo - Volumes orfaos
echo - Cache de build
echo.

set /p confirm="Deseja continuar? (S/N): "
if /i "%confirm%" neq "S" (
    echo Operacao cancelada.
    pause
    exit /b 0
)

echo.
echo Parando containers...
docker-compose down -v

echo.
echo Limpando sistema Docker...
docker system prune -af --volumes

echo.
echo Limpeza concluida!
echo Execute "start.bat" para reconstruir e iniciar
echo.
pause
```

## 🚀 Como Usar (Windows)

### Primeira Vez

cmd

```
# 1. Abrir PowerShell ou CMD na pasta do projeto  
cd C:\iptv-plataforma  
  
# 2. Configurar .env  
notepad .env  
  
# 3. Iniciar aplicação  
start.bat
```

## Uso Diário

### Via duplo-clique:

- Duplo clique em `start.bat` para iniciar
- Duplo clique em `stop.bat` para parar
- Duplo clique em `status.bat` para verificar

### Via linha de comando:

```
cmd  
  
start.bat      # Iniciar  
stop.bat       # Parar  
restart.bat    # Reiniciar  
logs.bat       # Ver logs de tudo  
logs.bat backend # Ver só backend  
logs.bat frontend # Ver só frontend  
status.bat     # Status detalhado  
clean.bat      # Limpar tudo
```

## ⚠ Problemas Comuns no Windows

### 1. Line Endings (CRLF vs LF)

**Problema:** Scripts falham com erro `\r': command not found`

### Solução:

```
powershell
```

```
# Configurar Git para não converter line endings  
git config --global core.autocrlf false
```

```
# Se já clonou, reconverter:  
git rm --cached -r .  
git reset --hard
```

## Ou usar .gitattributes:

```
gitattributes  
  
* text=auto  
*.sh text eol=lf  
*.bat text eol=crlf
```

## 2. Docker Desktop não inicia

### Soluções:

1. Habilitar virtualização na BIOS
2. Instalar WSL2: `wsl --install`
3. Atualizar Windows 10/11
4. Reiniciar PC após instalar Docker

## 3. Porta já em uso

### Verificar portas:

```
cmd  
  
netstat -ano | findstr :3000  
netstat -ano | findstr :3001
```

### Matar processo:

```
cmd  
  
taskkill /PID <numero_do_pid> /F
```

## 4. Permissões de arquivo

### Executar CMD/PowerShell como Administrador:

- Clique direito → "Executar como administrador"

## 5. Docker sem acesso à rede

Configurar DNS no Docker Desktop:

1. Settings → Docker Engine
2. Adicionar:

```
json
{
  "dns": ["8.8.8.8", "8.8.4.4"]
}
```

## 6. Erro ao buildar imagens

Limpar cache e rebuild:

```
cmd
docker-compose build --no-cache
```

## 🔧 Configurações Recomendadas do Docker Desktop

### Resources (Recursos)

Settings → Resources → Advanced

CPU: 4 cores (ou metade dos disponíveis)

Memory: 4GB mínimo, 8GB recomendado

Swap: 1GB

Disk: 40GB

### File Sharing

Certifique-se que C:\ está compartilhado:

Settings → Resources → File Sharing

✓ C:\

### Docker Engine (Configuração Avançada)

```
json
```

```
{  
  "builder": {  
    "gc": {  
      "defaultKeepStorage": "20GB",  
      "enabled": true  
    }  
  },  
  "experimental": false,  
  "features": {  
    "buildkit": true  
  }  
}
```

## Dicas para Windows

### 1. Use Windows Terminal (Recomendado)

- Melhor que CMD/PowerShell padrão
- Download: Microsoft Store → "Windows Terminal"

### 2. VS Code com Docker Extension

Extensions:

- Docker (Microsoft)
- Remote - Containers

### 3. Atalhos de Teclado

- **[Win + R]** → **[cmd]** → Enter (abre CMD rápido)
- **[Win + X]** → **[A]** (PowerShell como Admin)

### 4. Criar atalhos na área de trabalho

```
cmd  
  
# Botão direito → Novo → Atalho  
# Destino:  
C:\iptv-plataforma\start.bat
```

## Comandos Docker (Windows)

```
cmd
```

```
# Ver containers rodando  
docker ps  
  
# Ver todos os containers  
docker ps -a  
  
# Parar container específico  
docker stop iptv-backend  
  
# Remover container  
docker rm iptv-backend  
  
# Ver imagens  
docker images  
  
# Remover imagem  
docker rmi node:18-alpine  
  
# Ver logs de um container  
docker logs iptv-backend -f  
  
# Entrar no container (shell)  
docker exec -it iptv-backend sh  
  
# Ver uso de recursos  
docker stats  
  
# Limpar tudo  
docker system prune -a
```

## 🎯 Checklist de Verificação

- Docker Desktop instalado e iniciado
- WSL2 habilitado
- Projeto em `C:\iptv-plataforma\`
- Arquivo `.env` configurado
- Scripts `.bat` criados
- Portas 3000 e 3001 livres
- Internet funcionando
- Memória Docker: 4GB+

## Troubleshooting Windows Específico

### PowerShell vs CMD

**PowerShell** (recomendado):

```
powershell  
# Navegar  
cd C:\iptv-plataforma  
  
# Executar  
.\\start.bat
```

**CMD** (tradicional):

```
cmd  
# Navegar  
cd C:\iptv-plataforma  
  
# Executar  
start.bat
```

### Docker não reconhece comandos

**Adicionar Docker ao PATH:**

1. **[Win + Pause]** → Configurações avançadas
2. Variáveis de ambiente
3. PATH → Adicionar: **(C:\Program Files\Docker\Docker\resources\bin)**

### Suporte

Se tiver problemas:

#### 1. Docker Desktop Logs:

- Clique direito no ícone do Docker
- Troubleshoot → View logs

#### 2. Event Viewer (Windows):

- **[Win + X]** → Event Viewer
- Windows Logs → Application

#### 3. Verificar versões:

```
cmd
```

```
docker --version
```

```
docker-compose --version
```

```
wsl --version
```

## ✓ Resumo: Os arquivos Docker funcionam no Windows?

**SIM!** Todos os arquivos são compatíveis:

- `docker-compose.yml` - Funciona igual
- `Dockerfile` - Funciona igual
- `nginx.conf` - Funciona igual
- `.env` - Funciona igual
- Scripts `.sh` - Use `.bat` no lugar
- Containers Linux - Rodam via WSL2

Use os scripts `.bat` que criei acima!