



# Instrucciones para hacer Push a GitHub



## Estado Actual

El repositorio local está **limpio y sincronizado**. Todos los cambios recientes han sido commiteados.

### Últimos Commits Locales:

```
881896f - Mejoras multi-instancia y Coolify
af54797 - Documentación deployment
e076894 - Guía verificación GitHub Actions
ab50611 - Unificación Dockerfile v11.0
f432fb2 - Mejoras implementadas octubre 2025
```



## Cómo Hacer Push

### Opción 1: Usar el Script Automatizado (Recomendado)

Ejecuta el script que verifica todo automáticamente:

```
cd /home/ubuntu/escalafin_mvp
./PUSH_AHORA.sh
```

El script hará:

- ✓ Verificar que no hay cambios sin commitear
- ✓ Mostrar commits pendientes de push
- ✓ Solicitar confirmación antes de pushear
- ✓ Ejecutar el push y mostrar resultado

### Opción 2: Push Manual

Si prefieres hacer el push manualmente:

```
cd /home/ubuntu/escalafin_mvp

# 1. Verificar estado
git status

# 2. Ver commits pendientes
git log origin/main..HEAD --oneline

# 3. Hacer push
git push origin main

# 4. Verificar resultado
git status
```

## Verificaciones Antes del Push

### 1. Verificar Autenticación SSH

```
ssh -T git@github.com
```

Deberías ver:

```
Hi qhosting! You've successfully authenticated, but GitHub does not provide shell access.
```

### 2. Verificar Remote Configurado

```
git remote -v
```

Debe mostrar:

```
origin  git@github.com:qhosting/escalafin-mvp.git (fetch)
origin  git@github.com:qhosting/escalafin-mvp.git (push)
```

### 3. Verificar Branch Actual

```
git branch
```

Debe mostrar `* main` (con asterisco)

## Resolución de Problemas

### Error: “Permission denied (publickey)”

**Causa:** La clave SSH no está configurada correctamente.

**Solución:**

```
# Verificar si existe la clave SSH
ls -la ~/.ssh/

# Si no existe, generar nueva clave
ssh-keygen -t ed25519 -C "tu-email@ejemplo.com"

# Copiar clave pública
cat ~/.ssh/id_ed25519.pub

# Agregar la clave en GitHub:
# https://github.com/settings/keys
```

### Error: “Updates were rejected”

**Causa:** El remote tiene commits que no tienes localmente.

**Solución:**

```
# Traer cambios del remote
git pull origin main --rebase

# Resolver conflictos si los hay
# git add <archivos-resueltos>
# git rebase --continue

# Intentar push nuevamente
git push origin main
```

## Error: “Cannot run ssh”

**Causa:** SSH no está disponible en el entorno actual.

### Solución:

Necesitas ejecutar el push desde tu máquina local o desde el servidor con acceso SSH adecuado.



## Después del Push

### 1. Verificar en GitHub

Visita: <https://github.com/qhosting/escalafin-mvp>

Verifica:

- ☒ Los commits aparecen en el historial
- ☒ Los archivos están actualizados
- ☒ No hay errores en GitHub Actions (si está configurado)

### 2. Verificar Build Automático

Si tienes GitHub Actions configurado:

1. Ve a: <https://github.com/qhosting/escalafin-mvp/actions>
2. Verifica que el workflow se ejecute correctamente
3. Revisa los logs si hay errores

### 3. Actualizar Deployment

Si tienes un deployment activo (Coolify, Vercel, etc.):

```
# En el servidor de deployment
cd /ruta/al/proyecto
git pull origin main
docker-compose down
docker-compose up -d --build
```



## Commits Recientes Incluidos

Este push incluirá todos los cambios relacionados con:







### Docker y Deployment

- ☒ Dockerfile v12.0 con mejores prácticas
- ☒ Scripts de backup y restore de base de datos
- ☒ Script start.sh mejorado con healthchecks

-  docker-compose.yml optimizado







## Documentación

-  Análisis CitaPlanner y recomendaciones
-  Guía de mejoras implementadas octubre 2025
-  Documentación de verificación GitHub Actions
-  Guías de deployment Coolify






## Scripts Multi-Instancia

-  coolify-multi-instance.sh
-  coolify-quick-setup.sh
-  deploy-coolify.sh
-  Plantillas y configuraciones







## Instancia Demo

-  Archivos preparados en `/instances/demo/`
-  Archivo comprimido `escalafin-demo-instance.tar.gz`
-  Guías de descarga e instalación



## Siguiente Paso Después del Push

Una vez completado el push exitosamente:

1.  Verificar que el build pase en GitHub Actions
2.  Probar el deployment en ambiente de prueba
3.  Actualizar la instancia demo si es necesario
4.  Documentar cualquier cambio adicional



## Comandos Útiles

```
# Ver estado detallado
git status -vv

# Ver diferencias con el remote
git diff origin/main

# Ver historial gráfico
git log --graph --oneline --all -10

# Ver archivos cambiados en último commit
git show --stat

# Deshacer último commit (mantener cambios)
git reset --soft HEAD~1

# Forzar push (usar con precaución)
git push origin main --force
```

## Soporte

---

Si encuentras problemas durante el push:

1. Revisa los logs de error cuidadosamente
  2. Verifica tu conexión SSH con GitHub
  3. Asegúrate de tener permisos de escritura en el repositorio
  4. Consulta la documentación oficial de Git: <https://git-scm.com/doc>
- 

**Fecha de creación:** 16 de Octubre de 2025

**Versión del proyecto:** EscalaFin MVP v12.0

**Repositorio:** <https://github.com/qhosting/escalafin-mvp>