

Discrepancia de Entornos: VS Code vs Terminal Externa (Zsh)

Cuadernillo | Vol. 1

mercedev.es — 2026-04-24 | Fase 7 (Automatización y
Clasificación)

Contexto: Al trabajar en el proyecto, se detectó que la terminal integrada del IDE (Integrated Development Environment - Entorno de Desarrollo Integrado) VS Code activaba automáticamente el entorno virtual de Python (mostrando `(.venv)` en el *prompt*). Sin embargo, al abrir una terminal externa (Zsh) en la misma carpeta, el entorno no existía, lo que provocaba errores de dependencias no encontradas (como `weasyprint` o `Pillow`) al intentar ejecutar los scripts del orquestador.

Hecho: - Se diagnosticó que VS Code inyecta comandos de activación en su propia sub-sesión de terminal. - Se aplicó la activación manual del entorno virtual en la sesión de la terminal externa del sistema operativo.

Detalle técnico: Para sincronizar el estado de una terminal externa POSIX (Portable Operating System Interface - Interfaz de Sistema Operativo Portátil) con el entorno del proyecto, se debe invocar el script de activación directamente. Situado en la raíz del repositorio, se ejecuta:

```
# Activar el entorno virtual en Zsh o Bash
source .venv/bin/activate

# Para verificar que Python ahora apunta al entorno aislado:
which python

# Para salir del entorno cuando se termine de trabajar:
deactivate
```

Motivo / criterio: Aislamiento de sesiones y Cero Magia Negra. Las terminales externas del sistema operativo están diseñadas para ser agnósticas y seguras por defecto. No activan entornos ni modifican el `$PATH` sin una orden explícita del usuario (`source`). Entender que la "magia" de VS Code es solo automatización local evita horas de depuración al desplegar código en servidores o al cambiar de terminal.

Fuentes / Bibliografía: - Documentación oficial de Python sobre creación y activación de entornos virtuales (`venv`). - Documentación de VS Code sobre la integración de entornos Python en la terminal.