Normalización Robusta de Cadenas: El Poder de Casefold

Cuadernillo | Vol. 1

mercedev.es — 2026-05-05 | Fase 11 (CI/CD y Lighthouse)

1. El Desafío (Síntoma)

Al compilar el índice curado de la Biblioteca mediante el orquestador SSG (Static Site Generation - Generación de Sitios Estáticos), se detectó una fragmentación en la Arquitectura de la Información. Una misma estantería temática (por ejemplo, "Arquitectura de Software") aparecía listada dos veces de forma independiente.

El origen del error radicaba en variaciones minúsculas de tipografía introducidas por el factor humano en el YAML Frontmatter de distintos artículos (ej. `tema: "Arquitectura de Software"` frente a `tema: "arquitectura de software"`). Dado que los diccionarios en Python son sensibles a mayúsculas y minúsculas (*case-sensitive*), el script interpretaba estas variaciones como categorías completamente distintas.

2. La Maniobra (Lógica)

Se implementó una normalización robusta de cadenas de texto directamente en el bucle de agrupación del script `scripts/merci/merci-publish.py`.

En lugar de forzar a los autores a escribir siempre de la misma forma, se capturó el valor original (`tema_original = pub["tema"]`) para su uso visual en la interfaz, pero se utilizó el método `.casefold()` de Python para generar la clave interna de agrupación en el diccionario: `tema = tema_original.casefold()`

3. El Aprendizaje / Deuda Técnica

En el procesamiento de lenguajes y agrupación de datos, confiar en la disciplina tipográfica del usuario es un antipatrón de diseño. El sistema debe ser tolerante a variaciones inofensivas.

Se eligió explícitamente `.casefold()` en lugar del tradicional `.lower()` porque el primero es un método agresivo diseñado específicamente para comparaciones de cadenas independientes del formato (*case-matching*). Mientras que `.lower()` se limita a convertir caracteres ASCII a minúsculas, `.casefold()` maneja correctamente caracteres especiales y ligaduras de otros idiomas (como el alemán o el griego), garantizando que la agrupación lógica nunca falle por cuestiones de codificación internacional.