

## Propósito de las pruebas

Todas las pruebas en la clase `ControlGatoTest` se crearon para verificar que los métodos de la clase `ControlGato` identifiquen correctamente, según el estándar EMS (Extension of the Minimal Standard), las características de un gato (raza, color de cuerpo, patrón de pelaje, color de ojos y tipo de cola). Al usar un mock de `ControlPrincipal`, se aísla `ControlGato` de cualquier dependencia externa, garantizando que estemos evaluando únicamente su lógica interna.

Cada método `identificarXSegunEMS` recibe un arreglo de strings con un único elemento (código EMS o nombre legible) y debe devolver la cadena combinada “código–nombre” o `null` si la entrada no existe. Con estas pruebas se cubren:

Entrada por código válido → devuelve código + descripción.

Entrada por nombre válido → devuelve código asociado + nombre ingresado.

Entrada inválida → devuelve `null`.

## Resultados obtenidos

Raza (`identificarRazaSegunEMS`)

Código "BRI" → "BRI-British Shorthair" ✓

Nombre "Persian" → "PER-Persian" ✓

"RazaInexistente" → `null` ✓

Color de cuerpo (`identificarColorCuerpoSegunEMS`)

Código "a" → "a-blue" ✓

Nombre "red" → "d-red" ✓

Patrón de pelaje (`identificarPatronSegunEMS`)

Código "21" → "21-tabby / lynx" ✓

Nombre "spotted" → "24-spotted" ✓

Color de ojos (`identificarColorOjosSegunEMS`)

Código "61" → "61-blue" ✓

Nombre "green" → "64-green" ✓

Tipo de cola (`identificarColaSegunEMS`)

Código "51" → "51-rumpy" ✓

Nombre "stumpy" → "53-stumpy" ✓

Todas las aserciones pasaron, confirmando que `ControlGato` traduce correctamente códigos y nombres EMS a las descripciones esperadas y maneja entradas desconocidas devolviendo `null`.