Cinco vueltas a EcomTPV, y lo que salió

Del 9 al 10 de septiembre de 2026. Cinco pasadas completas, cada una con los mismos seis pasos: borrar todas las capturas, conectar con las cuatro tiendas PrestaShop de cliente que hay montadas en local, probar las capacidades del mapa, no guardar dos fotos iguales, revisar con un negocio y revisar con cinco.

Esto no es un informe de que todo salió bien. Es la lista de lo que se rompió.

Los tres fallos del programa

1. Devolver no reponía las existencias

Siete sitios de la interfaz cambiaban una línea del ticket construyéndola campo por campo, y **los siete perdían** `Articulo`. Lo contó un guion, no un vistazo:

DÓNDE	QUÉ PERDÍA
Cambiar la cantidad	Articulo, Modificadores, Nota, NumeroDeSerie
Retarifar por cliente	y además CodigoBarras y Referencia
Aplicar un cupón	y además CodigoBarras y Referencia
Poner una nota	Articulo, NumeroDeSerie
Cambiar el precio	Articulo, Modificadores, Nota, NumeroDeSerie
Descuento de línea	Articulo, Modificadores, Nota, NumeroDeSerie
Canjear puntos	Articulo, Modificadores, Nota, NumeroDeSerie

`Articulo` es el identificador de la combinación vendida: lo único que distingue una talla de otra, porque las ocho tallas de una camiseta comparten la referencia del producto padre. Sin él, **devolver no repone el almacén**. La venta sale perfecta, el ticket sale perfecto, y el descuadre aparece semanas después contando género.

Uno de los siete ya se había arreglado un mes antes. Los otros seis seguían ahí, porque **arreglar un sitio no arregla un patrón**. Ahora hay una sola forma de cambiar una línea —`linea.Con(...)`, que copia todo— y una prueba que **lee el código fuente** y falla si alguien vuelve a construir una a mano.

2. La nota de línea no se veía

El programa tiene su botón «Nota», su ventana y su texto de ayuda: «sin cebolla», «poco hecho», «sin gluten». Todo bien escrito. Y la pantalla no la pintaba: se escribía la nota, se aceptaba, y el ticket salía igual que antes.

Lo único que hace falta de una nota es que **alguien la lea**. La veía quien la escribía, en la ventana donde la escribía, y ahí se acababa.

3. El cajero no veía lo que estaba vendiendo

El panel del pedido decía «Pedido actual · 1 línea», el total ponía 5,00 €, y **el recuadro que enseña las líneas medía 0 píxeles**.

El bloque de totales y botones iba anclado abajo sin tope, y las líneas rellenaban lo que sobrara. En esa clase de reparto el anclado se lleva todo lo que pide. Con las dos franjas de aviso arriba —215 px, y salen en cuanto falta el certificado o hay envíos pendientes con la AEAT— no quedaba nada.

A 1280×800. El manual documenta como mínimo 1024×768, así que en el mínimo era peor.

Ninguna prueba lo cazaba, y ésa es la parte que importa. Todas preguntaban al árbol de la interfaz, no a la pantalla: buscar el texto de una línea lo encuentra igual de bien aunque esa línea mida cero de alto. La prueba de la nota decía «ok, la nota sale en el ticket» mientras en la pantalla no salía nada.

Se vio abriendo una captura y mirándola. No hay otra forma.

Y dos fallos de la propia prueba, que acusaban al programa

Se cuentan porque una prueba que culpa al programa de algo que hace bien cuesta más tiempo que un fallo de verdad.

- **Comparaba todas las facturas del libro contra los tickets de hoy.** Mientras la tanda cabía dentro del mismo día las dos cuentas salían iguales. La tanda del 9 de septiembre empezó a las 23:5x y cruzó la medianoche: cuatro facturas quedaron en el día 9 y tres en el 10. El informe contó 3 —bien— y la prueba gritó que el libro tenía 7.
- **Mandaba las teclas por el canal que no era.** Los atajos del programa escuchan por el camino de ida y la prueba mandaba por el de vuelta, así que concluía que «la tecla + no suma

una unidad» con las teclas funcionando perfectamente.

Lo que se puede decir de la cobertura, sin adornos

El mapa tiene **786 capacidades**. Decir «786 probadas» sería mentira, y decir «147 de 786» asusta sin querer decir lo que parece. Medido:

	CUÁNTAS	QUÉ SON
De pantalla	143	nombran una ventana o un botón: se pueden pulsar
Del núcleo	229	no hay nada que pulsar; las cubren las 1.560 pruebas unitarias
Sin decir dónde viven	414	hay que mirarlas una a una

El número que hay que subir es el primero. En estas cinco vueltas pasó de 58 a más de 100, y buena parte vino de abrir **las ventanas de segundo nivel** —Lotes, Proveedores, Traspasos, Reposición, Etiquetas, Cierres, Tiendas, Fichajes—, que cuelgan de otra pantalla y que **ninguna prueba había abierto nunca**. Que es exactamente donde se esconde el código que no invoca nadie.

Cómo se comprueba todo esto

```
cd "…/utiles/EcomTPV/pruebaui"
dotnet build && ./bin/Debug/net8.0-windows/pruebaui.exe <carpeta de capturas> 5 <vuelta>
```

Deja dos ficheros que **no se borran nunca**:

- `_CAPACIDADES.log` — una línea por capacidad y por vuelta, con su resultado.
- `_CAPACIDADES.md` — el resumen: «Capacidad 3.11 · vueltas 1, 2, 3 · confirmado sin errores».

Y se publica solo en **https://ecomtpv.gmartos.es/capacidades.php**.

Las capturas se borran al empezar cada vuelta, a propósito: así lo que queda al final es lo que ha salido de esa vuelta, no un montón acumulado. Y no hay dos iguales: cada imagen se compara por su huella y la repetida no se guarda —en una vuelta se descartaron 845 de 1.417—, pero **la capacidad sí queda apuntada a la imagen que ya existía**, que antes no pasaba y

hacía parecer que faltaban capturas donde no faltaban.