

detalles. Entenderá que la función se encarga de añadir páginas de configuración y detalles en una página de prueba, que después muestra en HTML. Si está familiarizado con JUnit^[11], verá que esta función pertenece a algún tipo de estructura de pruebas basada en la Web y, evidentemente, es correcto. Resulta sencillo adivinar esta información del Listado 3-2 pero no del Listado 3-1. ¿Qué tiene la función del Listado 3-2 para que resulte sencilla de leer y entender? ¿Qué hay que hacer para que una función transmita su intención? ¿Qué atributos podemos asignar a nuestras funciones para que el lector pueda intuir el tipo de programa al que pertenecen?

Tamaño reducido

La primera regla de las funciones es que deben ser de tamaño reducido. La segunda es que *deben ser todavía más reducidas*. No es una afirmación que pueda justificar. No puedo mostrar referencias a estudios que demuestren que las funciones muy reducidas sean mejores. Lo que sí puedo afirmar es que durante casi cuatro décadas he creado funciones de diferentes tamaños. He creado monstruos de casi 3000 líneas y otras muchas funciones de entre 100 y 300 líneas. También he creado funciones de 20 a 30 líneas de longitud. Esta experiencia me ha demostrado, mediante ensayo y error, que las funciones deben ser muy reducidas.

En la década de 1980 se decía que una función no debía superar el tamaño de una pantalla. Por aquel entonces, las pantallas VT100 tenían 24 líneas por 80 columnas, y nuestros editores usaban 4 líneas para tareas administrativas. En la actualidad, con una fuente mínima y un monitor de gran tamaño, se pueden encajar 150 caracteres por línea y 100 líneas o más en una pantalla. Las líneas no deben tener 150 caracteres. Las funciones no deben tener 100 líneas de longitud. Las funciones deben tener una longitud aproximada de 20 líneas.

¿Qué tamaño mínimo debe tener una función? En 1999 visité a Kent Beck en su casa de Oregon. Nos sentamos y comenzamos a programar. Me enseñó un atractivo programa de Java/Swing que había llamado *Sparkle*. Generaba un efecto visual en pantalla, similar a la varita mágica del hada de

Cenicienta. Al mover el ratón, salían estrellitas del cursor, y descendían a la parte inferior de la pantalla en un campo gravitatorio simulado. Cuando Kent me enseñó el código, me sorprendió la brevedad de las funciones. Estaba acostumbrado a ver programas de Swing con funciones que ocupaban kilómetros de espacio vertical. En este programa, las funciones tenían dos, tres o cuatro líneas de longitud. Todas eran obvias. Todas contaban una historia y cada una llevaba a la siguiente en un orden atractivo. ¡Así de breves deberían ser todas las funciones!^[12]

¿Qué tamaño mínimo deben tener sus funciones? Deberían ser más breves que las del Listado 3-2. De hecho, el Listado 3-2 debería reducirse como el Listado 3-3.

Listado 3-3

HtmlUtil.java (nueva refactorización).

```
public static String renderPageWithSetupsAndTeardowns(
    PageData pageData, boolean isSuite) throws Exception {
    if (isTestPage(pageData))
        includeSetupAndTeardownPages(pageData, isSuite);
    return pageData.getHtml();
}
```

Bloques y sangrado

Esto implica que los bloques en instrucciones `if`, `else`, `while` y similares deben tener una línea de longitud que, seguramente, sea la invocación de una función. De esta forma, no sólo se reduce el tamaño de la función, sino que también se añade valor documental ya que la función invocada desde el bloque puede tener un nombre descriptivo. También implica que las funciones no deben tener un tamaño excesivo que albergue estructuras anidadas. Por tanto, el nivel de sangrado de una función no debe ser mayor de uno o dos. Evidentemente, de esta forma las funciones son más fáciles de leer y entender.

Hacer una cosa