**Code smell 1**

El código usa “class.equals” para saber la clase de una variable.

Para resolverlo, pusimos métodos nuevos en el modo de inscripción que nos permitan preguntarle si deja o no el lugar para anotarse.

**Code smell 2**

El método “toString” del jugador tiene un comentario que dice algo que no es lo que sucede en código.

Para resolverlo se agregaron las cosas que el comentario decía que debía devolver el método.

**Code smell 3**

La distribución de equipos esta dentro de partido, y le da muchas responsabilidades.

Para resolverlo creamos nuevas clases que tengan un método que nos permita distribuir.

**Code smell 4**

Se usan ints para distribuir errores en la clase Partido y se usan strings como variables de estado.

Para resolver esto decidimos que el método “validarInscripcion” tire la excepción directamente y reificamos los estados, haciendo que cada uno tenga un método “validar” que tire la excepción en caso de que se requiera.

**Code smell 5**

El código que queda del Code smell 3 todavía usa muchos ifs y números para cada distribución.

Para arreglar esto hicimos que la variable de instancia “distribución de equipos” sea de tipo “Distribuidor”, un tipo polimórfico que contiene a las clases nuevas creadas en el Code smell 3.

Luego sacamos los ifs y simplemente llamamos al método “distribuir” de la clase.