

## **Documentación Entrega 4**

Decidí implementar el patrón Builder en ambos requerimientos. Mi explicación es la siguiente: al tener diferentes criterios de ordenamiento de la lista, puedo elegir crear un Ordenador de Lista distinto para cada Partido, por ejemplo, ahora quiero ordenar mi lista por la calificación del último partido y luego cambiarla, entonces podría hacerlo tranquilamente. Me permite también ser versátil a la hora de agregar nuevos criterios, ya que implementaría la lógica de ese sin tocar el resto por ejemplo. Puedo variar la representación interna de mi objeto, es decir, cambiarle el Ordenador. También me permite aislar el código de los Ordenadores como mencione anteriormente. (Libro Gamma página 89 – patrón Builder). A la hora de crear los equipos tomé la misma decisión que para los ordenadores, creí que merecían un tratamiento similar.

Utilizo polimorfismo, ya que todas las subclases del builder, entienden el mensaje de ordenar, pero dentro tienen un comportamiento distinto, propio de cada tipo de Ordenador.

Podría haber utilizado un Factory en lugar de un Builder.

Decidí no crear mensajes para ordenar la lista dentro del partido, ya que, en el caso de querer agregar varios criterios de ordenamiento de la lista, de a poco se iría convirtiendo en un God Object, lo cual no está bueno.