**Universidad de San Carlos de Guatemala**

**Facultad de Ingeniería**

**Escuela de Ciencias y Sistemas**

**Curso: Análisis y Deseño de Software II**

**Ejemplo de patrones de diseño**

**Integrantes:**

**2000-18283 Henry Haroldo Fajardo Velásquez**

**2011-13930 Eddie Orlando Xuyá Monroy**

**2008-16543 Jose Eliu Lacam Perez**

**2013-14035 Estuardo Rafael López Ramírez**

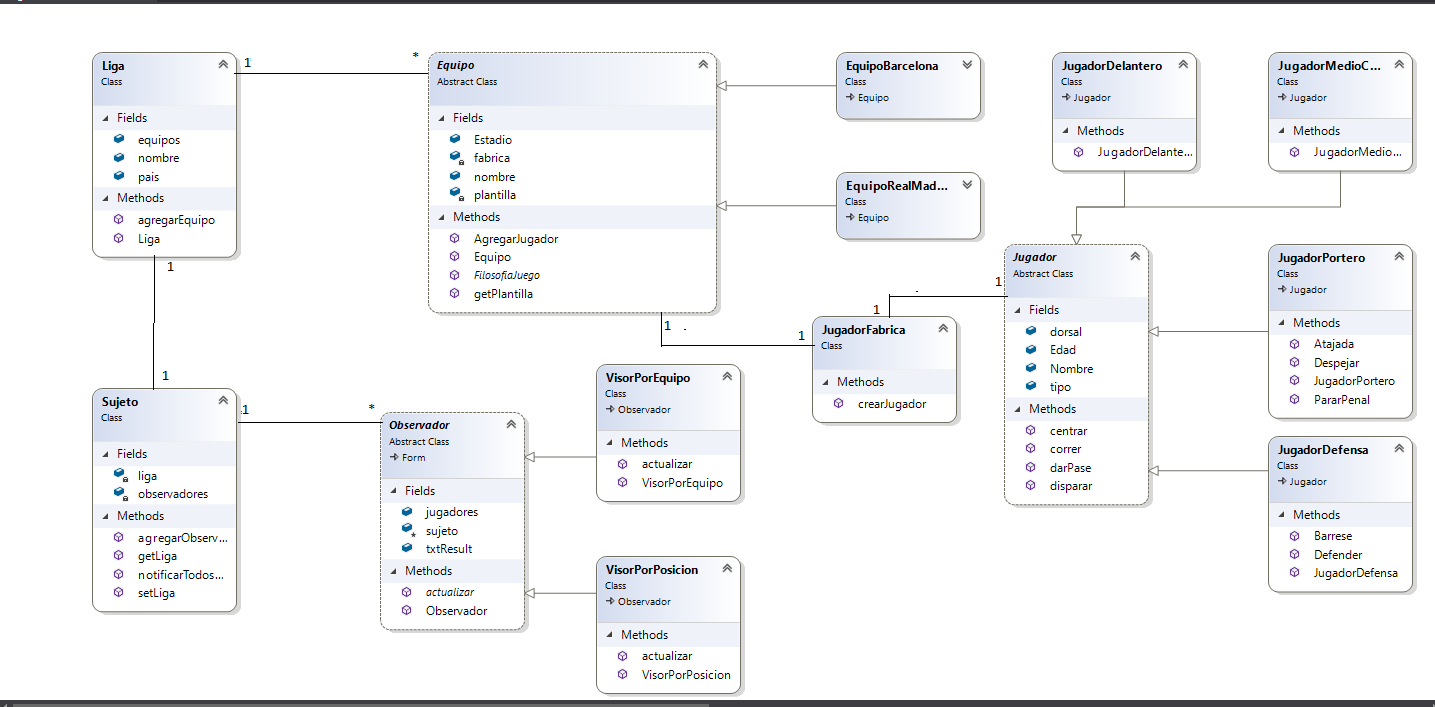
# Patrones de diseño

## Descripción del problema a resolver

**Liga de Futbol:** Se desea crear una solución de software para la creación de una aplicación que sea capaz mostrar reportes de los equipos que componen una liga, así como también de los jugadores que componen cada equipo, la aplicación debe permitir en todo momento poder agregar nuevos jugadores.

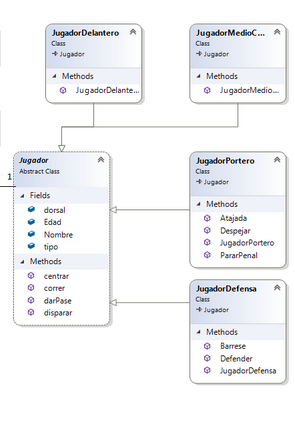
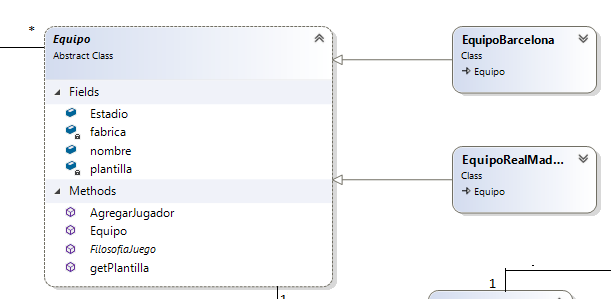
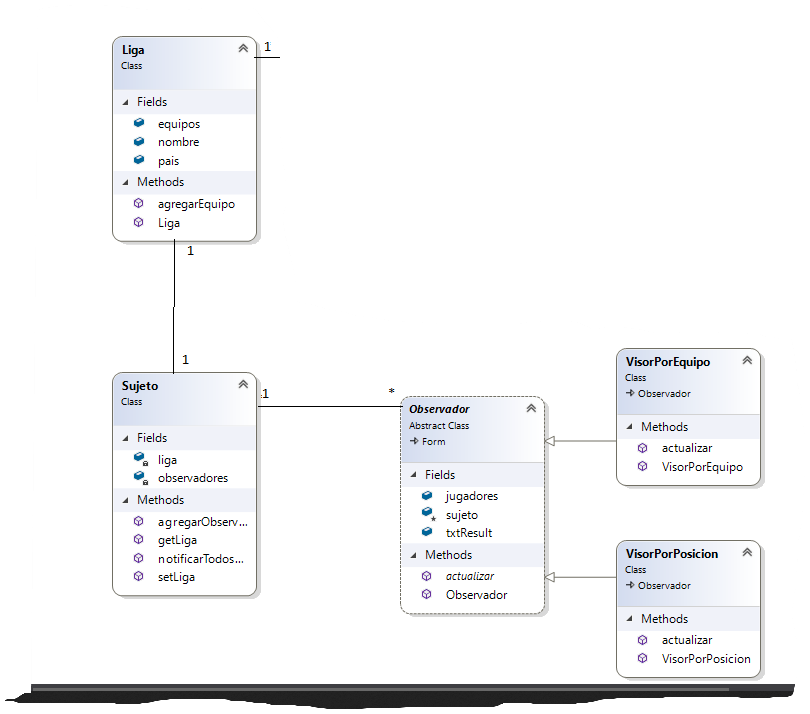
## Descripción y diagrama de clases de la solución

El diagrama de clases de la solución quedo de la siguiente manera.

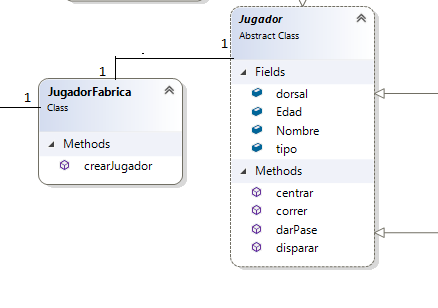


El patrón Estrategia implemento en la forma que se definieron las clases para los jugadores, aplicando esta lógica se pueden agregar cualquier tipo de jugador definiendo siempre una nueva clase que hereda del interfaz jugador.

También se aplicó el patrón Estrategia con la creación de los equipos que conforman la liga.

El patrón observador se implementó para la creación de los reportes, para que se actualicen de forma dinámica cada que se ingrese un nuevo jugador.

El patrón Factory se utilizo para poder crear objetos de tipo jugador sin tener que crear directamente la instancia del objeto.



## Código de la solución:

El código de la solución se encuentra disponible en este [repositorio](https://github.com/EstuardoLopez/G5_Patrones_Disenio) de GitHub. El

## Los patrones Aplicables son:

Los patrones que aplicamos fueron

* Factory Patrón de creación
* Observador Patrón de comportamiento
* Estrategia Patrón de creación