Patrón Singlenton

|  |  |
| --- | --- |
| Ventajas | Desventajas |
| Una clase Singleton puede ser modificada fácilmente si los requerimientos cambian. | Llegará a un punto donde se requieran nuevas instancias. |
| La clase Singlenton es la única que puede crear objetos de la clase, eso hace que su singularidad sea segura. | Puede causar un problema de acceso “multithread”, esto depende de la manera en la que se haya implementado este patrón. |
| Simplifica el desarrollo y permite que sea mantenible la programación. | Posee restricción de ejecuciones paralelas. |
| Reduce el uso de memoria al evitar instancias de objetos que siempre devolverán la misma información. | El Singleton es instanciado directamente desde su propia clase promocionando el uso de métodos privados y estáticos. Esto acopla la clase que los use además de impedir el uso adecuado de inyección de dependencias. |
| Permite el refinamiento de las operaciones y la representación. | Se pueden producir valores que se mantienen siempre dificultando el uso de objeto de reemplazo(mock) en test. |

(Villadiego, 2013) (Alfredo, 2016)

¿Cree que su uso es adecuado en este programa?

# References

Alfredo. (2016, Diciembre 14). *Singlenton*. Retrieved from TimeOfSoftware: http://timeofsoftware.com/patron-singleton/

Villadiego, F. (2013, marzo 31). *El objeto único [Patrón Singleton]*. Retrieved from http://eljaviador.com/el-objeto-unico-patron-singleton.html