

PATRONES DE SOFTWARE

línea horizontal

**22 DE SEPTIEMBRE DEL 2019 REVISIÓN: 1.0**

En el presente proyecto hemos contemplado usar los siguientes patrones de Software:

* **MVC o Modelo-Vista-Controlador**

***Fundamento el uso del patrón MVC:***

Este patrón se ha implementado por la facilidad de organización en el ámbito web, ya que permite separar los datos de una aplicación, la interfaz de usuario, y la lógica de control en componentes distintos. Este patrón basa su heurística en la reutilizacion de codigo y el desglosado de conceptos.

* **Patrón Dao**

El patrón del Objeto de acceso a datos (DAO) es un patrón estructural que nos permite aislar la capa de aplicación / negocio de la capa de persistencia (generalmente una base de datos relacional, pero podría ser cualquier otro mecanismo de persistencia) utilizando una API abstracta.

La funcionalidad de esta API es ocultar de la aplicación todas las complejidades involucradas en la realización de operaciones CRUD en el mecanismo de almacenamiento subyacente. Esto permite que ambas capas evolucionan por separado sin saber nada el uno del otro.

* **Patrón Facade:**

El patrón Facade (fachada) tiene la característica de ocultar la complejidad de interactuar con un conjunto de subsistemas proporcionando una interface de alto nivel, la cual se encarga de realizar la comunicación con todos los subsistemas necesarios. La fachada es una buena estrategia cuando requerimos interactuar con varios subsistemas para realizar una operación concreta ya que se necesita tener el conocimiento técnico y funcional para saber qué operaciones de cada subsistema tenemos que ejecutar y en qué orden, lo que puede resultar muy complicado cuando los sistemas empiezan a crecer demasiado.

* **Inyección de dependencias**

La inyección de dependencias es un patrón de diseño de software usado en la Programación Orientada a Objetos, que trata de solucionar las necesidades de creación de los objetos de una manera práctica, útil, escalable y con una alta versatilidad del código.

En la mayoría de los frameworks actuales se aplica la Inyección de dependencias como parte de las herramientas y modelos que facilitan al programador. Como cualquier patrón de diseño de software trata de solucionar de una manera elegante un problema habitual en el desarrollo de software, por lo que también es idóneo utilizar este patrón en el desarrollo de proyectos a pequeña escala.