

Evaluación Final

Objetivo

El objetivo de esta actividad es que los estudiantes demuestren su comprensión de los conceptos de orientación a objetos, patrones de diseño y su capacidad para aplicarlos en la práctica. Deberán elegir un patrón de diseño de cada familia (creacionales, estructurales y de comportamiento), explicarlo en detalle, proporcionar un ejemplo implementado en Python y presentar un ejemplo compuesto que utilice al menos dos de los tres patrones elegidos.

Consigna

1. Elija un patrón de diseño de cada una de las siguientes familias:
 - Patrones creacionales: *Factory Method*, *Abstract Factory*, *Builder*, *Singleton*
 - Patrones estructurales: *Adapter*, *Bridge*, *Decorator*, *Facade*, *Composite*
 - Patrones de comportamiento: *Observer*, *Strategy*, *Template Method*, *Chain of responsibility*, *Iterator*, *Command*, *State*
2. Para cada patrón elegido, realice lo siguiente:
 - a. Proporcione un ejemplo implementado en Python utilizando el patrón de diseño. Incluya el código en un repositorio de GitHub.
 - b. Diseñe y presente un ejemplo compuesto que utilice al menos dos de los tres patrones elegidos. Incluya el código en el mismo repositorio de GitHub y proporcione el enlace correspondiente.
3. Prepare un documento que contenga lo siguiente:
 - a. Una explicación detallada de cada patrón de diseño elegido, incluyendo su propósito, estructura y casos de uso comunes.
 - b. Diagramas de clase UML para cada ejemplo implementado, utilizando los patrones de diseño.
 - c. Enlaces a los repositorios de GitHub que contengan el código correspondiente.