Imagen que contiene Icono

Descripción generada automáticamente

**DESARROLLO DE SOFTWARE**

**GUÍA 2 – ACTIVIDAD 1**

**PATRONES DE DISEÑO DE SOFTWARE**

**AUTORES:**

MAYCKOLL ANDRÉS TORRES DIAZ.

**TUTOR**

DILSA ENITH TRIANA MARTÍNEZ

**3 DE MARZO DE 2024**

**INTRODUCCIÓN**En el presente informe se especifica y representa los patrones de diseño de software los cuales son soluciones reutilizables a problemas comunes que surgen durante el desarrollo de sistemas informáticos. Estos pueden facilitar y estructurar de manera más organizada la disponibilidad, adaptabilidad y fiabilidad en cuanto la funcionalidad estes directamente proporcionales al objetivo principal del aplicativo, además de proporcionar la reutilización y actualización del código facilitando su puesta a marcha en el ámbito productivo.

**OBJETIVOS**

**Objetivo General:**

* Explicar la funcionalidad principal de los patrones de diseño de software con el fin de implementarlo en el desarrollo y actualización de este.

**Objetivos Específicos:**

* Planificar la estructura del software para adaptabilidad y reutilización de código sin perder las funcionalidades guiados en la arquitectura de los patrones de software.
* Demostrar la disponibilidad y adaptabilidad del software.

**Patrón MVC (Modelo-Vista-Controlador):** La estructura por patrón MVC se realiza segmentando una aplicación en tres elementos claramente definidos.

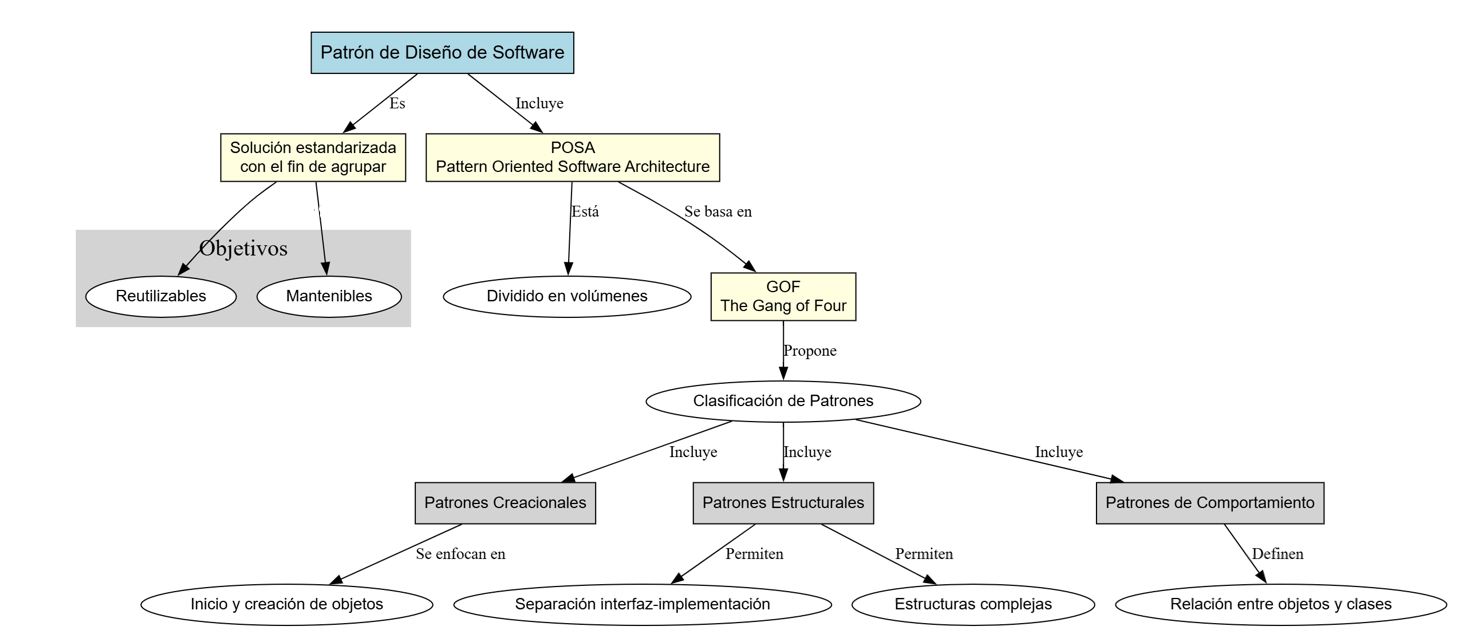
**Formato:** Esta unidad se encarga de gestionar la evidencia de la aplicación y la lógica comercial. Es en este lugar donde se guarda y maneja la información.

**Vista:** Es lo que percibe el usuario, la interfaz con la que se relaciona. La vista muestra los datos que se extraen del modelo.

**Sistema de Control**: Este elemento tiene la responsabilidad de gestionar la relación entre el usuario y la aplicación. Recibe las acciones del usuario y determina cómo actuar, ya sea renovando el modelo o alterando lo que el usuario percibe en la pantalla.

Este patrón ayuda a mantener el código más ordenado, dado que distingue de manera clara las funciones de las demás.

**Patrones de Diseño de Software**

****

LINK GITHUB: https://github.com/mehirakiva/EAN\_Patrones\_de\_diseno\_software/blob/d86a85a72c1ef460a06a5c64ceffd37f2c4f9910/Actividad1/Mapa\_Conceptual.jpg

**CONCLUSIONES**

Esta arquitectura permite las adaptabilidad y actualización del software con el objetivo principal de crear una aplicación escalable, adaptable y confiable guiados por patrones de diseño de software.