

ACTIVIDAD 7: Análisis Comparativo con Arquitecturas Industriales

En la industria del desarrollo de software, existen varios enfoques arquitectónicos consolidados que buscan resolver necesidades comunes de escalabilidad, mantenimiento y robustez. Entre las soluciones más habituales se encuentran la arquitectura por capas (Layered Architecture), la arquitectura de microservicios y la arquitectura hexagonal o limpia. A continuación, se ofrece una comparación de la propuesta de la plataforma inteligente de gestión de proyectos con estas metodologías.

La arquitectura diseñada para la plataforma mantiene una división clara en capas: presentación, aplicación, dominio (negocio) y datos. Esto se alinea en gran medida con la arquitectura tradicional por capas, que promueve la separación de responsabilidades y facilita la escalabilidad horizontal o la reutilización de componentes. Al igual que en la arquitectura por capas, se emplea un punto de acceso central (Facade) que simplifica la interacción con el resto de servicios, lo cual contribuye a un bajo acoplamiento entre módulos.

Por otro lado, la propuesta no se construye bajo un esquema de microservicios. Mientras que la arquitectura de microservicios fragmenta la aplicación en múltiples servicios independientes que se comunican a través de interfaces ligeras, la plataforma aquí planteada funciona como un sistema cohesionado, con diferentes servicios compartiendo un mismo proceso o despliegue. Aun así, el uso de interfaces y patrones como Strategy o Factory Method facilita la evolución a un esquema de microservicios en un futuro, si la demanda o el nivel de tráfico lo justificaran.

En comparación con enfoques como la arquitectura hexagonal o limpia, la propuesta conserva principios similares al mantener la lógica de negocio aislada y encapsulada en el dominio, usando abstracciones para evitar dependencias directas sobre frameworks o implementaciones específicas. Esta semejanza resulta beneficiosa para la mantenibilidad, ya que los patrones de inyección de dependencias y la modularidad allanan el camino hacia cambios o adiciones en la lógica de negocio.

Entre las fortalezas de la solución se encuentra su alineación con principios modernos (S.O.L.I.D., DRY, KISS, YAGNI) y el uso de patrones de diseño reconocidos, que mejoran la claridad del código y la facilidad de modificación. La principal oportunidad de mejora podría residir en descomponer aún más los componentes, evolucionando hacia una arquitectura más descentralizada (microservicios), si el crecimiento de la plataforma demanda alta escalabilidad independiente por servicio.

En conclusión, la arquitectura propuesta se aproxima en gran medida a las soluciones usadas en la industria, al aprovechar las ventajas de la separación en capas y el uso de

patrones de diseño. Su principal diferencia radica en no implementar desde el inicio un enfoque de microservicios, aunque sí sienta las bases necesarias para lograr una transición hacia modelos más distribuidos, en caso de que las necesidades del sistema lo requieran.