**Patrones de Arquitectura de Software**

**Arquitectura en capas:**

La arquitectura basada en capas se enfoca en la distribución de roles y responsabilidades de forma jerárquica proveyendo una forma muy efectiva de separación de responsabilidades. El rol indica el modo y tipo de interacción con otras capas, y la responsabilidad indica la funcionalidad que está siendo desarrollada.

Por ejemplo, una aplicación web típica está compuesta por una capa de presentación (funcionalidad relacionada con la interfaz de usuario), una capa de negocios (procesamiento de reglas de negocios) y una capa de datos (funcionalidad relacionada con el acceso a datos).

**Arquitectura orientada a eventos:**

es un patrón de arquitectura de software que promueve la producción, detección, consumición y reacción en base a eventos

**Aquitectura de microkernel:**

se aplica a sistemas de software que deben estar habilitados para adaptarse a requerimientos cambiantes del sistema. Separa un núcleo de funcionalidad mínima de la funcionalidad extendida y de partes específicas al cliente. También sirve como un socket para conectores en estas extensiones y coordinar su colaboración.

**Arquitectura de microservicios**:

es un enfoque para desarrollar una aplicación software como una serie de pequeños servicios, cada uno ejecutándose de forma autónoma y comunicándose entre sí, por ejemplo, a través de peticiones HTTP a sus API. Normalmente hay un número mínimo de servicios que gestionan cosas comunes para los demás (como el acceso a base de datos), pero cada micro servicio es pequeño y corresponde a un área de negocio de la aplicación.