Investigacion#2 Patrones de Arquitectura de software

**Arquitectura por capas**

La Programación por capas es una arquitectura cliente servidor, su meta principal es separar la lógica de los negocios de la lógica de diseño. Una de sus características principales es que el estilo de desarrollo se puede llevar a cabo en varios niveles, y en caso de que suceda algún cambio, solo se hace el cambio en el nivel que se requiere y no todo el código mezclado.

Además permite el trabajo de creación de una aplicación por niveles, es una buena forma de distribuir todo, y cada grupo de trabajo está separado del resto de niveles

**Arquitectura Orientada a eventos**

Este patrón de arquitectura se basa en la producción, detección, consumo de, y reacción de eventos. Es importante saber que un evento, es un “es un cambio de algo”. Este patrón se aplica a sistemas que produzcan eventos entre componentes de software que estén emparejados libremente y servicios. Este sistema basado en eventos está formado por emisores de eventos(o agentes) y consumidores de eventos.

Construir aplicaciones y sistemas alrededor de una arquitectura dirigida por eventos permite a estas aplicaciones y sistemas ser construidos de una manera que facilita un mayor grado de reacción, debido a que los sistemas dirigidos por eventos están, por el diseño, más normalizados para entornos no predecibles y asíncronos.

**Arquitectura de micro-servicio**

Se basa en procesos de desarrollo de software de entrega continua. Aplicaciones complejas que poseen pequeños e independientes procesos que se comunican entre si usando lenguajes API. Estos pequeños bloques de procesos, son altamente desacoplados y se enfocan en hacer tareas pequeñas, facilitando un enfoque modular para la construcción de sistemas.

Propiedades de arquitectura micro-servicio

Los servicios son fáciles de remplazar

Los servicios se organizan en capacidades, recomendaciones, logística,

Los servicios pueden implementarse en diferentes lenguajes de programación, base de datos, hardware y software, dependiente en que encaja mejor.

La arquitectura es mas simétrica que jerárquica (productor - consumidor)

**Arquitectura basada en espacio**

es un patrón de arquitectura de software por lograr una gran escalabilidad lineal de aplicaciones y de alto rendimiento utilizando el paradigma del espacio de tuplas . Sigue muchos principios de el estado de transferencia representacionalREST), arq. Orientada a servicios, y arq orientada a eventos.Las aplicaciones se construyen a partir de un conjuntos de unidades autosuficientes , conocidas como unidades de procesamiento. Estas unidades son independientes entre sí , de modo que la aplicación se puede ampliar mediante la adición de más unidades.

**Arquitectura basada en microkernel**

Esta arquitectura hace juego con el sistema, porque debe estar disponible a cada cambio del sistema y sus requerimientos que pida.

Esto quiere decir que separa la mínima funcionalidad principal de la funcionalidad extendida y las partes especificas del cliente

El modelo de arquitectura microkernel consiste en dos tipos de componentes de arquitectura: un sistema principal y módulos de enchufe de unión. La lógica de aplicación es dividida entre módulos de enchufe de unión independientes y el sistema básico principal, proporcionando la extensibilidad, la flexibilidad, y el aislamiento de rasgos de aplicación y costumbre que procesa la lógica.

Muchos sistemas operativos ponen en práctica el modelo de arquitectura microkernel, de ahí el origen del nombre de este modelo. De una perspectiva de aplicación de gestión, el sistema principal a menudo es definido como la lógica general de negocio sans el código de encargo para casos especiales, reglas especiales, o el tratamiento complejo condicional.