Análisis de restricciones de entorno de arquitecturas de software

# Resumen

Este artículo se centra en el análisis de cómo diferentes restricciones de entorno al momento de implementar el software impactan el rendimiento de la arquitectura de software. Se busca examinar distintas arquitecturas como monolítica, SOA, microservicios entre otras. El objetivo principal es identificar como los distintos factores como la conectividad, la capacidad de procesamiento afecta el rendimiento de la arquitectura. Objetivo es proporcionar una guía que permita seleccionar la arquitectura más adecuada en función de las limitaciones impuestas por el entorno en el que operará el sistema.

Abstrac.

This article focuses on the analysis of how different environmental constraints at the time of software implementation impact the performance of the software architecture. It seeks to examine different architectures such as monolithic, SOA, microservices among others. The main objective is to identify how different factors such as connectivity, processing capacity affect the performance of the architecture. The objective is to provide a guide to select the most appropriate architecture according to the constraints imposed by the environment in which the system will operate.

Translated with DeepL.com (free version)

# Introducción

# Marco Teórico

@

Arquitectura de software: La arquitectura se tiene en cuenta como el proceso donde se analiza las características técnicas, sociales y del negocio más convenientes para el sistema. De acuerdo con Quilindo (2022), el diseño arquitectónico debe basarse en los requisitos y restricciones tecnológicas para asegurar la eficiencia del sistema en el entorno en que operará.

Tómese como arquitectura de software el conjunto de decisiones de diseño que definen la estructura fundamental de un sistema de cómputo. Contemplando las partes componentes del sistema: sus interfaces, comportamiento, las relaciones entre ellas, las relaciones de ellas con el medio y la evolución de cada componente junto al sistema. @

Arquitectura de software

La arquitectura se entiende como el conjunto de decisiones que define la estructura del sistema, se enfatiza en análisis de características de los componentes del sistema como sus comportamientos, organización, comunicación y relaciones, y también analiza el entorno social y del negocio con el fin de tomar las decisiones que enfaticen la eficiencia, efectividad, adaptabilidad y escalabilidad.

Microservicios

Se entiende como suites de servicios desplegados independientemente, servicios que son específicos. Descomponiendo funciones independientemente.

1. Ventajas:
   1. Modularidad,
   2. Escalabilidad,
   3. Despliegue independiente,
2. Desventajas:
   1. Alto consumo de memoria,
   2. Tiempo de desarrollo,
   3. Complejidad de gestión de un gran número de servicios,
   4. Pruebas o testeos complicados al despliegue distribuido.

SOA

Se entiendo como el sistema que descompone módulos desacoplados que se comunican entres si, facilita la escalabilidad y adaptación a diferentes entornos.

1. Ventajas
2. Desventajas

@

La respuesta del servicio es afectada directamente por aspectos externos como problemas en la red y configuración por citar algunos. Estos deben ser tenidos en cuenta en el diseño, desarrollándose los mecanismos de contingencia que eviten la parálisis de las aplicaciones y servicios que dependen de él. Debe manejar comunicaciones no confiables, mensajes impredecibles, reintentos, mensajes fuera de secuencia.2 Según lo anterior, se puede ver que la construcción de un servicio es una tarea mucho más complicada que la de un simple componente distribuido.

Cuando se usan múltiples servicios para implementar un sistema, es muy fácil que la comunicación entre estos se salga de control.

 El desarrollo bajo el paradigma de arquitectura orientada a servicios disminuye el acoplamiento entre los componentes de una aplicación, facilitando la adaptación de las aplicaciones desarrolladas e independizando la colaboración entre componentes de su distribución geográfica o plataforma de base. @

Monolítico

Se entiende como el sistema en donde toda la aplicación se encuentra apilada.

1. Ventajas:
   1. Simplicidad en el despliegue.
   2. Menor dependencia de la red.
2. Desventajas:
   1. Baja escalabilidad.
   2. Dificultad para realizar cambios o actualizaciones.

Restricciones de entorno

Las restricciones de entorno hacen referencia a aspectos que afectan al sistema negativamente generando baja eficiencia, baja eficacia o que directamente inutilice el sistema en la industria.

# Metodología

Análisis: Analizar los conceptos teóricos clave para determinar los limites de cada arquitectura de software

Criterios de evaluación: Se evaluará:

Dependencia de red: Se establece si la red es indispensable para el funcionamiento de la arquitectura analizada

Requerimiento de recursos: Se establece el uso de recursos y la eficiencia de estos

Escalabilidad: Se establece la capacidad del software que tiene para crecer en funcionalidad y no en complejidad

# Desarrollo

# Conclusión

Se determina que las arquitecturas de software se enfatizan en resolver problemas que se pueden presentar naturalmente esto hace que cada arquitectura tenga sus puntos fuertes y débiles lo cual cada arquitectura tiene sus limitaciones.

Microservicios: (Escalabilidad, disponibilidad de red)

SOA: (Escalabilidad, disponibilidad de red)

Monolítico: (baja escalabilidad, sin dependencia de red)

# Bibliografía

Velasco, J. I. P., Ruiz, A. I. R., & Alvira, H. A. P. (2019). Arquitectura basada en microservicios para aplicaciones web. *Tecnología Investigación y Academia*, *7*(2), 12–20. <https://revistas.udistrital.edu.co/index.php/tia/article/view/13364>

Quilindo, Luz.Amanda, Vega, Juan-Sebastián. (2021), Especificando una Arquitectura de Software.Revista Tecnol.Investig.Academia TIA, ISSN: 2344-8288, 10 (2), pp. 136-148. Bogotá-Colombia.

*DeepL Translate - El mejor traductor del mundo*. (s/f). Deepl.com. Recuperado el 12 de noviembre de 2024, de <https://www.deepl.com/es/translator>

(S/f). Unirioja.es. Recuperado el 27 de noviembre de 2024, de <https://dialnet.unirioja.es/servlet/articulo?codigo=4786655>

Contreras, D. A. B. (2018). Arquitectura de Microservicios. *Tecnología Investigación y Academia*, *6*(1), 36–46. <https://revistas.udistrital.edu.co/index.php/tia/article/view/9687>