**Resumen de artículos de las revistas**

**Mayra Alejandra Tamayo Perdomo**

**Jesús Ariel**

**Servicio Nacional de Aprendizaje SENA**

**Centro de la industria y los servicios**

**Análisis y Desarrollo de software 2694667**

**2024**

Angie casanova

La edad de oro en la arquitectura del software

La arquitectura de software ha venido evolucionando constantemente Ha pasado de ser una descripción a una a convertirse en una guía esencial para el desarrollo de software, el arquitecto debe proporcionar un sistemas solido escalable y mantenible

Publico: Que ha cambiado en la arquitectura desde los años 80

Primero se centraba en la funcionalidad y la eficiencia un enfoque limitado era lo que hacia y ya l adopción de la programación orientada a objetos que permitió la reutilización de código que ayudo a mejorar la escalabilidad y mantenibilidad del software

-En cuanto a la evolución cual a sido el aspecto mas importante para segirle el paso a las nuevas tecnologías

Capacitarse y educarse a las nuevas tecnologías y mejores prácticas adaptándose a las nuevas tecnologías

Aura maria fierro

Implementación de una arquitectura guiada por el dominio

Crea sistemas mas eficiente diseño guiado por el dominio ddd es una metodología de desarrollo de software que pone un enfoque principal para el negocio solucionar y como resolver algunos problemas, arquitectura limpia y la hexagonales, limpia busca mantener el código organizada hexagonal busca desacoplar la lógica del negocio del sistema, son la base de los ,micro servicios, beneficio tiene mayor alineación con el negocio mantenimiento más sencillo y sostenibilidad a largo plazo, en si gestiona la complejidad del software.

-que significa un sistema diado por el dominio

Significa que el software se centre que el software sea mas fácil de entender y tenga una mejor mantenibilidad

Camilo andres bautista

Arquitectura de software fundamentos y teorías

Es fundamental en el desarrollo del sistema son especialmente valiosas para productos donde abordan patrones de diseño, si no hay una arquitectura de software bien planteada y bien analizadas no hay un bien software ya que ella es la base donde para desarrollar un software mas eficiente es un boceto para el software para no tener tanta confusión en el software y que sea sostenible

-que aspecto cubre la arquitectura del software

Es la base de lo que se va a construir y de esto depende si el proyecto se desarrolla con éxito

Carolina martines

Arquitectura para herramientas de costo y programación

Esta herramienta apoya el aprendizaje autónomo de los aprendices, utiliza un metodología secuencial que va desde la materia hasta el costo, puede simular proceso lo que facilita la construcción y evidenciar los impactos

Cristian narvaez

Documentación y análisis

La arquitectura de software es clave en el desarrollo de sistemas empresariales eficientes, especialmente

para herramientas como ERP y CRM, que optimizan procesos y gestión. Este análisis revisa frameworks

1. Arquitectura en capas: organiza responsabilidades jerárquicamente, facilita mantenimiento y

cambios aislados, pero puede ser redundante y menos eficiente en casos complejos.

2. Cliente-servidor: separa roles entre clientes que solicitan datos y servidores que responden, ideal

para sistemas distribuidos y multiusuario.

3. Arquitectura en tres capas: divide funciones en presentación, lógica y datos, logrando

modularidad y escalabilidad.

El uso adecuado de estos enfoques asegura aplicaciones alineadas con objetivos empresariales, mejorando

recursos y rendimiento.

Cristian jeanpool

Análisis comparativo de diseño MVC y MVP para el rendimiento de aplicaciones web

Se centra en al eficacia de los patrones de diseño modelo vista controlador y modelo vista presentado, se hizo un estudió evaluando vario aspecto de cada uno de ellos, los aspectos que se evaluaron fueron el número de línea el tiempo de respuesta, el uso de memoria ram y el uso de cpu,se evidencio que la mejor opción era el modelo vista controlador ya que era mas eficiente de instalar aplicaciones complejas además el tiempo de desarrollo el modelo vista controlador era el más eficaz ya que era más rápido, se destacó más en su productividad y rendimiento en base a el otro modelo.

Pena

Arquitectura hexagonal

Llamada también Arquitectura de puertos y adaptadores, promueven la separación de respondalibilidades que facilita el desarrollo y mantenimiento; esta arquitectura divide el sistema un capa una capa central que contiene la idea de negocio y varias interfaces que se comunican con ella por medio de puertos y adaptadores, separa la lógica de negocio de la dependencias externas para facilitar el desarrollo mantenimiento y pruebas de software

Gasca

Modelo y herramienta software para la gestión de riesgos en el desarrollo de aplicaciones web

Tres perspectivas conceptuales lógicas y físicas conceptuales, creación de proyecto, parametrización, identificación de riesgo y gestión de riesgo. la perspectiva físicas conceptuales Utiliza diagrama uml física conceptual estructura el diseño de la base de datos, el iso es un estándar que trata de gestionar los riesgo en el desarrollo de aplicaciones web se pueden utilizar tanto para proyecto medianos o grandes, proporciona un enfoque sistemático contiene un buen estándar de seguridad y se pueden identificar con tiempo los riesgos para así tener un desarrollo más robusto

Julián

Arquitectura para una herramienta de patrones de diseño

Esta diseñada para integrar patrones de diseño permiten manipular y diseñar patrones de modelado practico facilita la eficiencia y reutilización del software ayudando a los trabajadores a desarrollar, compositin estructurara un elemento en una arquitectura mas flexible comman facilita la ejecución de operaciones de un objeto observe sincroniza diferentes componentes, en si la arquitectura facilita la interacción entre la interfaces graficas del usuario, es un gran ejemplo como los patrones pueden mejorar la eficiencia y reutilización de un software.

Laura

Mapeo de arquitectura de software

Se refiere un proceso que analizar y identificar como está estructurado internamente un proyecto en modulo servicio base de datos o documentos, uno de los problemas mas comunes es cuando esta mal gestionado se proponen soluciones para gestionar el diseño del sistemas, si la información recuperada no es clara es difícil de entender y de utilizarla en otro proyectos afectando la comunicación.

Maria del mar

Arquitectura de microservicios para el desarrollo wed

La asamblea generar de ecuador con su sistema dti hace mucho tiempo habían desarrollado una arquitectura monolítica que no cambia, necesitaban cambiar este desarrollo y estaban en el proceso de evaluar que arquitectura le funciona a su proyecto, ante esto escogieron fue la de micro servicios que está dividida en capas, esta fue la mas adecuada para su proyecto.

Maria jose Murcia

Patrones de usabilidad en las arquitecturas de software

Integrar la usabilidad desde el principio es importante porque permite detectar problemas temprano, evitando costos y tiempo extra. También asegura que el software sea fácil de usar, accesible y funcional, mejorando la experiencia del usuario desde el inicio. Esto ayuda a los desarrolladores a trabajar mejor y a crear un producto más intuitivo.

Mariana charry

Arquitectura de software para entornos remotos móviles

La evolución de los sistemas operativos móviles ha sido muy importante porque ahora son más estables y confiables. Esto permite a los desarrolladores crear aplicaciones más grandes y complejas. Además, se ha definido una arquitectura móvil que sigue principios reconocidos de la arquitectura de software, ayudando a estandarizar y mejorar los métodos de desarrollo. Gracias a estos avances, las aplicaciones son más rápidas, fiables y ofrecen una mejor base para crear software innovador y de calidad.

Mariana Gonzales

Marco de trabajo para seleccionar

Nos habla de la importancia de la arquitectura de software, nos presenta un marco de trabajo para seleccionar patrones lo cual mejora la calidad y mantenimiento y adaptabilidad de un software este marco guía a los arquitectos, se describen patrones arquitectónicos como: mvc, mvp. Microservicios y arquitectura de la nube

Maydy

Desarrollo de aplicaciones web utilizandondo el patrón me dodelo vista controlador

El modelo de diseño mvc es útil para diseño de software interactivos se encarga de manejar datos y la lógica de la aplicación, El patrón MVC es muy útil para organizar sistemas interactivos y aplicaciones web. Sin embargo, dividir las partes de una aplicación demasiado pronto puede complicar las cosas. La propuesta de "Partición Flexible" permite aplicar MVC sin cambiar mucho el código cuando hay modificaciones en la arquitectura, haciendo el desarrollo más fácil y adaptable.

Mayra

Análisis comparativo de patrones de diseño de software

La información destaca la importancia de los **patrones de diseño** como soluciones estándar para problemas comunes en el desarrollo de software. Analiza cinco patrones clave (Template Method, MVC, MVP, Front Controller y MVVM), evaluando sus componentes, ventajas, desventajas y usos en diferentes contextos. Concluye que no hay un patrón superior, sino que cada uno tiene un propósito específico y que los patrones mejoran la organización, mantenimiento y calidad del código. También resalta que elegir el patrón adecuado según las necesidades del proyecto fomenta prácticas de programación más efectivas y sostenibles.

Patricias

Marco de trabajo para la seleccionar un patrón arquitectónico en el desarrollo de software

el artículo propone un marco para elegir patrones arquitectónicos como MVC o Microservicios en el desarrollo de software. Explica cómo seleccionar el mejor patrón según el proyecto y sus características. Esto mejora la calidad, escalabilidad y mantenimiento del software, ayudando a planear estructuras sólidas desde el inicio para ahorrar tiempo y costos a largo plazo.

Valentian silva

Análisis comparativo de patrones de diseño de software

Los patrones de diseño ayudan a crear software fuerte y escalable, ofreciendo soluciones según el contexto y las necesidades del proyecto. Se comparan patrones como Template Method, MVC, MVP, Front Controller y MVVM, considerando su lenguaje, complejidad y seguridad. No hay un patrón perfecto; la elección depende del proyecto, permitiendo decisiones más eficientes y sostenibles.

Willy

Buenas practicas en la construcción del software

El artículo trata sobre buenas prácticas en la construcción de software. Menciona diferentes arquitecturas y metodologías utilizadas para diseñar soluciones eficientes. Habla de la importancia de las arquitecturas como capas, monolíticos, microservicios, y más, destacando cómo organizan la comunicación y estructura de aplicaciones

Yordy

Introducción a los patrones de doseño

La reflexión del artículo destaca la importancia de usar patrones de diseño para crear software flexible y seguro. Mencione que estos patrones ayudan a mejorar la calidad del código y permiten soluciones prácticas para resolver problemas comunes. También se habla de cómo los patrones organizan las estructuras de pr

Además, el artículo explica cómo estos patrones contribuyen a la creación de software eficiente y reutilizable, ayudando a los desarrolladores a tomar mejores decisiones en sus pro.

Otra ficha

Calderón

Revisión de elementos conceptuales para la representación de las arquitecturas de referencias de software

La arquitectura de software ayuda a organizar un sistema de manera clara y eficiente, definiendo los elementos y sus relaciones. Facilita el desarrollo de los componentes mediante patrones y estructuras, haciendo que sistemas complejos sean más fáciles de entender y manejar. A medida que evoluciona, permite crear software más eficiente y comprensible.

Maryury

Monolitos en arquitectura de software

La de micro servicio permite adaptarse a proyectos robusto para mayor flexibilidad y mantenibilidad en el software, compara las arquitecturas monolíticas de micro servicios, se refería a como la empresas cambian los proyectos monolíticamente cambiarlos a micro servicios para ofrecer mas servicios una ventaja es la alta escalabilidad y flexibilidad, los aislamientos de fallo ya que un error de micro servicio no afecta todo el sistema permitiendo crecimiento continuo

Carlos Pantoja

Impacto de implementaciónes web del patrón mvc en los requisitos de calidad percibidos

El artículo estudió dos formas de implementar el patrón MVC en aplicaciones web y cómo afectan la calidad, como el tiempo de respuesta y la escalabilidad. Compara la implementación clásica de MVC con una alternativa que usa tuberías y filtros en Python. Concluye que, aunque esta alternativa tiene potencial, aún se necesitan más estudios para determinar su efectividad en la práctica.

Nikol

Desarrollo de un arquitectura de software para el robot móvil lazaro

La arquitectura de software para robots tiene tres niveles: gestiona los componentes básicos, permite crear aplicaciones de control y ofrece una interfaz con un simulador 3D para monitorear y programar el robot. Este enfoque mejora la organización, eficiencia y reutilización en la programación de robots.

Juan David

Marco de trabajo para seleccionar un patrón arquitectónico en el desarrollo de software

El artículo habla sobre un estudio que investigó cuáles son las arquitecturas de software más usadas por desarrolladores. Destacan varias: Arquitectura en la nube: Ofrece seguridad y flexibilidad.MVC (Modelo-Vista-Controlador): Es popular por su mantenibilidad y rendimiento.Microservicios: Ayudan al rendimiento y flexibilidad del software.MVP (Modelo-Vista-Presentador): Ideal para aplicaciones móviles y web.

Cada una de estas arquitecturas es útil según las necesidades, como aplicaciones web, móviles o sistemas en la nube. También se nombra que se creo un modelo para escoger una arquitectura de software la cual sugiere debe seguir unos pasos para escoger adecuadamente la arquitectura: la Selección del medio que esta destinado para el software, el usuario debe escoger las tres principales características que necesita en su software

Erich

Arquitectura de software basada en microservicios para desarrollo de aplicaciones web

Explora el desarrollo de una arquitectura flexible y moderna busca superar las limitaciones la investigación se enfoca en como los servicios ofrecen una solución mas eficiente y adaptable para el desarrollo y mantenimiento de aplicaciones, arquitectura monolítica es limitada para las necesidades actuales de aplicaciones y que las arquitectura de servicios es apta para los que nos piden los servicios

Marlon

Patrones de diseño(XII): patrones estructurales-Flyweight

Este articulo describe el patrón flyweigh permite reducir el uso de memoria compartiendo tantos datos como sea posible con otros objetos similares. Se presentan ejemplos de su aplicación en sistemas donde la eficiencia de memoria crucial, como en aplicaciones graficas y juegos