

Resúmenes de Artículos

Angie lizeth Trujillo Gonzáles

No aplica

anyie zuje y gomez

la edad de oro de la arquitectura del software

la exposición estuvo un poco enredada por el tema que fue la primera exposición mientras se adaptaba al tema La arquitectura de software ha evolucionado desde los años 80, pasando de ser una simple descripción de sistemas a convertirse en una disciplina clave para diseñar y construir software complejo. Hoy en día, gracias a métodos y herramientas, permite crear sistemas más robustos, escalables y mantenibles, sirviendo como una guía esencial en el desarrollo de software.

Aura Maria Fierro Fierro 11

implementación de arquitectura de software por el dominio

El enfoque de Domain-Driven Design (DDD) en la ingeniería de software se centra en crear sistemas orientados al negocio, definiendo cómo se percibe y desarrolla el software. Las arquitecturas limpia y hexagonal desempeñan un papel clave en este contexto.

La arquitectura limpia facilita la comprensión y legibilidad del sistema, promoviendo un diseño claro y bien estructurado. Por su parte, la arquitectura hexagonal añade flexibilidad al permitir que el sistema sea organizado tanto vertical como horizontalmente, lo que mejora su capacidad de adaptación y mantenimiento.

Camilo Andrés Bautista Cuellar 1

Arquitectura de software fundamentos teoría y práctica

Se afirma que, sin una buena arquitectura de software, no es posible desarrollar una aplicación de calidad. También se menciona la importancia de las arquitecturas Web y Red. Antes de comenzar a programar, es fundamental definir una arquitectura que sirva como base, lo que permite desarrollar la página de manera más rápida y flexible. porque es importante la arquitectura de software es la base del proyecto la arquitectura de software tiene los componentes, aparte de eso se habla de la pirámide de costo tiempo y alcance

Carolina Martinez Cortes 4

arquitectura para herramientas de costos y programación

El artículo aborda una arquitectura educativa aplicada a la materia de Costo, Presupuesto y Programación de Obra. También se menciona la arquitectura secuencial, la cual sigue una secuencia lógica y ordenada, con pasos definidos. Este enfoque es especialmente útil para sistemas complejos que requieren un desarrollo estructurado y progresivo.

Cristian Fernando Narváez sanchez 24

Documentación y análisis de los principales frameworks de arquitectura de software en aplicaciones empresariales

Se habla sobre los ERP (Sistemas de Planificación de Recursos Empresariales), que se encargan de organizar diferentes áreas de una empresa, incluyendo la parte financiera. También se mencionan los CRM (Gestión de Relaciones con Clientes), que facilitan la gestión de contactos, el marketing, el análisis de datos y otros procesos relacionados.

En estos sistemas, se utilizan diversas arquitecturas, como la arquitectura en capas (de dos capas, tres capas) y la arquitectura cliente-servidor, dependiendo de las necesidades y la complejidad del sistema.

Cirtian Jeanpool Bahamon Granada

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

Se describe una prueba realizada a dos modelos: el modelo Vista-Controlador y el modelo Vista-Presentador. Los aspectos evaluados en la prueba incluyeron el uso de memoria RAM, las líneas de código y el tiempo de respuesta. Estos fueron los principales puntos considerados para analizar y comparar estos patrones de diseño.

Dylan Santiago Narvaez Pinto

No aplica

Isabella Cordoba Gutiérrez

No aplica

Ivan Andrés Murcia Epia

no aplica

John Sebastián Penna Arias

Arquitectura Hexagonal

Se explica qué es y cómo funciona la arquitectura hexagonal, un estilo arquitectónico que busca aislar la lógica de negocio del resto del sistema. Esto permite que la aplicación interactúe con diferentes tipos de interfaces externas, como bases de datos, interfaces de usuario o servicios externos, de manera independiente y flexible.

José Manuel Gasca Bonilla 3

Modelo herramienta software para la gestión de riesgo en el desarrollo de aplicaciones web soportado en el estándar ISO/IEC 27005

El modelo propone una arquitectura de software basada en tres perspectivas: conceptual, lógica y física. También aborda los riesgos asociados al inicio de una aplicación web.

Además, se menciona el estándar ISO/IEC 27005, un modelo utilizado en proyectos internacionales que ofrece beneficios como la protección de datos y la mejora de la seguridad en los negocios nacionales.

Julian David Fierro Casanova 1

Una arquitectura para una herramienta de patrones de diseño

Se habla sobre la arquitectura para integrar patrones de diseño orientados a objetos, lo cual nos ayuda a simplificar y reutilizar el código. Se mencionan tres patrones de diseño:

Composite, que permite estructurar jerarquías de manera más flexible. Command, que facilita la manipulación de los comandos de movimiento. Observer, que sincroniza diferentes componentes del sistema. Estos patrones contribuyen a una mejor organización y eficiencia en el desarrollo de software.

Kevin Camilo Muñoz Campos

No aplica

Laura valentina ariza alejo 11

Mapeo de Arquitecturas de software

Esto nos habla sobre la recuperación de vistas arquitectónicas de sistemas de software, y cómo se soluciona mediante ingeniería inversa, con documentación y reutilización de diagramas UML, para que sea más fácil de entender e interpretar.

Manuel Ricardo Diez Corredor

NO aplica

Maria del mar Artunduaga Artunduaga

arquitectura de microservicios para desarrollo web

Se menciona que se migrará un proyecto que actualmente se encuentra en una arquitectura monolítica a una arquitectura de microservicios, con el fin de que el proyecto no se quede estático, sino que sea más rápido y se pueda mantener por módulos.

Maria José Murcia Martínez 1

Patrones de Usabilidad en la Arquitectura de Software

Nos habla sobre el proyecto 'Status' y la experiencia de usuario. Nos propone las opciones de deshacer, cancelar y soporte para múltiples idiomas para mejorar la interacción del usuario. En lugar de esperar hasta el final para evaluar la usabilidad, 'Status' utiliza un método inductivo que permite ajustar el diseño de forma continua durante el proceso.

Mariana Charry Prada

arquitectura de software para entornos móviles

Mariana Gonzáles Calderón

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

Mejora la calidad, mantenimiento y adaptabilidad del software. Los patrones más utilizados son MVC, MVP, microservicios y arquitectura en la nube.

Maydy Viviana Conde Ladino

Desarrollo de aplicación web utilizando el patrón de diseño Modelo Vista Controlador.

El modelo de diseño Modelo Vista Controlador para sistemas interactivos es complejo. Nos habla sobre este patrón de diseño, que es popular por su aplicación en el manejo de la lógica de la aplicación, la vista del usuario y el controlador, que maneja la interacción con el usuario.

Mayra Alejandra Tamayo Perdomo

análisis comparativo de patrones de diseño de software

Se destaca la importancia de los patrones de diseño estándar para resolver problemas comunes en el desarrollo de software. El documento analiza cinco patrones clave: Template Method, MVC y MVP. No existe un modelo que sea superior a otro; cada uno tiene un propósito específico. Por lo tanto, es fundamental ser cuidadoso al seleccionar el patrón adecuado al iniciar un proyecto, ya que la elección impactará directamente en su desarrollo y mantenimiento.

Patricia Sarmiento 12

marco de trabajo para seleccionar un patron arquitectónico en el desarrollo de software

El artículo plantea que, para desarrollar un software, es fundamental seleccionar una arquitectura adecuada para el proyecto, lo que garantiza mayor eficiencia. Además, destaca la importancia de contar con un arquitecto en el equipo, ya que es la persona que posee el conocimiento necesario para tomar decisiones clave en el diseño y estructura del sistema.

Valentina Silva Garrido

ananlisis comaprativo de patrones de diseño de software

Cada desarrollador elige el patrón que mejor se ajuste a su diseño, evaluando aspectos como la escalabilidad, la mantenibilidad y la compatibilidad. En el artículo, también se analizan los lenguajes de programación utilizados en cada patrón de diseño y se considera la seguridad. Además, se realiza una evaluación de la complejidad y la seguridad, calificándolas en una escala del 1 al 3.

Willian Steban Gonzales Cortes

Buenas practicas en la construcción del software

Este artículo de reflexión explora diferentes arquitecturas de software y metodologías de desarrollo utilizadas para crear soluciones eficientes flexibles en sistemas de ti. Primero, detalla la arquitectura de solución, que guia el diseño de estructuras integradas Para responder a necesidades actuales y futuras, y la arquitectura de software, que de tiene patrones y tecno logias específicas

Yordy Erick Nuñez Pineda

Introducción a los patrones de diseño

Los patrones de diseño son soluciones reutilizables y probadas para problemas comunes que surgen durante el desarrollo de software. Estos patrones proporcionan un enfoque estandarizado para abordar problemas que han sido encontrados y resueltos en el pasado, ayudando a los desarrolladores a crear sistemas más eficientes, escalables y mantenibles. Existen varios tipos de patrones de diseño, que se agrupan generalmente en tres categorías: creacionales, estructurales y comportamentales.