

## 20 Resúmenes

Nombre:  
Johan Calderón Perdomo

Instructor:  
Jesús Ariel González

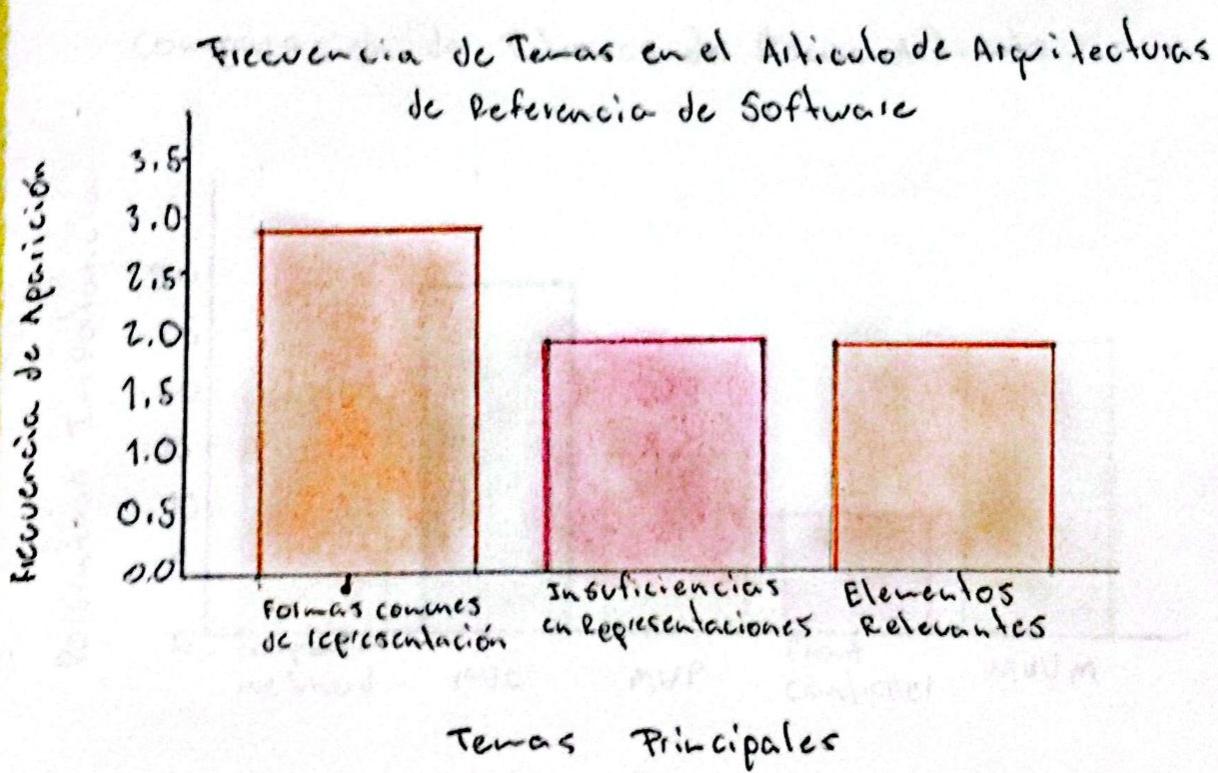
Servicio Nacional de Aprendizaje SENA  
Sede Industria

Neiva - Huila

14-09-2024

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

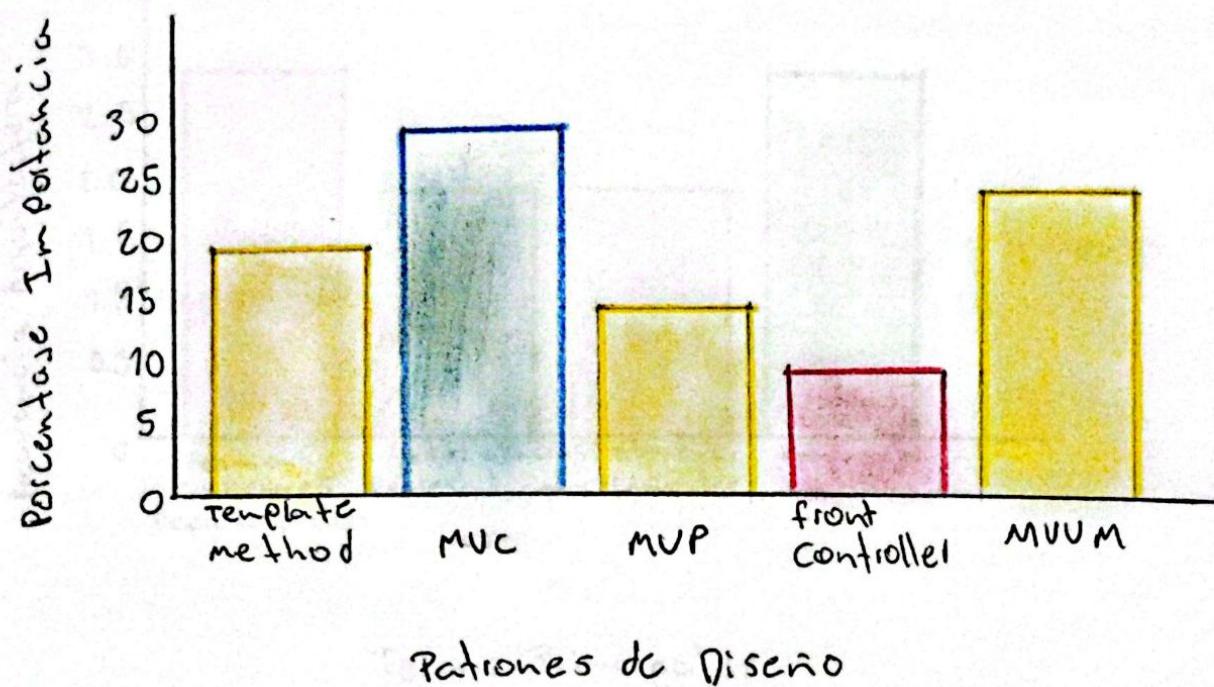
La arquitectura de software organiza un sistema a nivel alto, definiendo elementos y sus relaciones. Facilita el desarrollo al reutilizar componentes mediante arquitectura de referencia. Ha evolucionado a través de enfoques estructurales, patrones y escenarios, guiando el diseño y la compresión de sistemas complejos.



## Análisis comparativo de Patrones de Diseño de software

los patrones de diseños evitan duplicación de código y mejoran la calidad del software. El artículo compara varios patrones, concluyendo que no hay uno superior, ya que cada uno tiene un propósito específico, facilitando el mantenimiento y la modularidad del sistema.

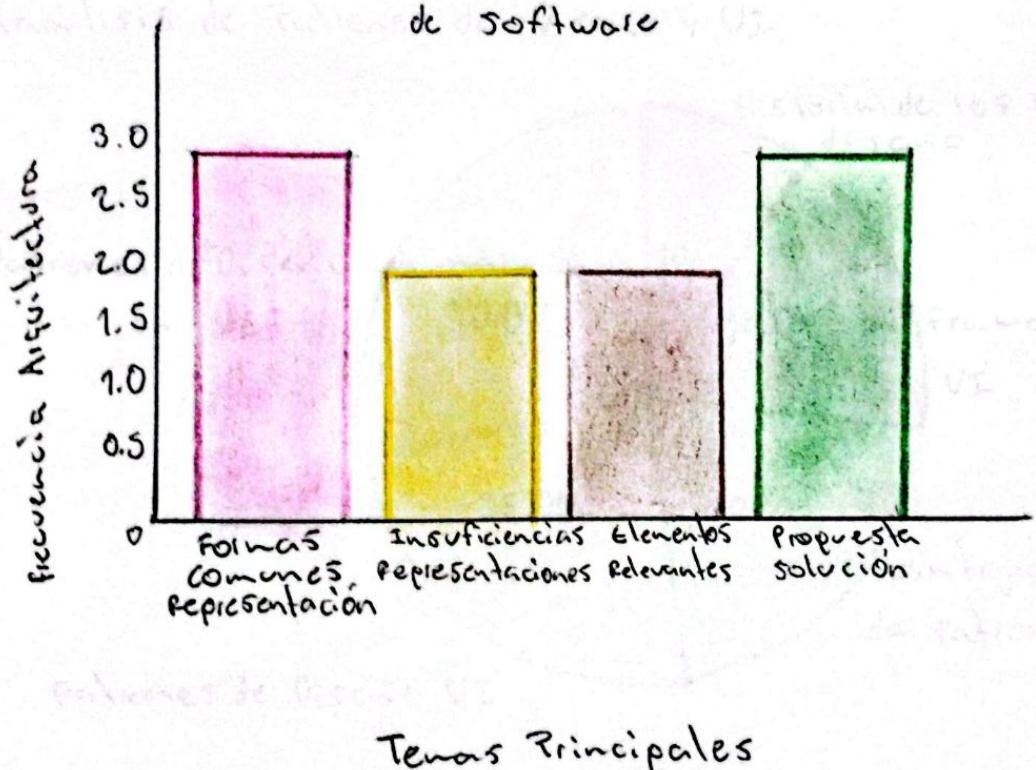
Comparación de Patrones de Diseño Artículo



## Análisis de secuencia discreta para la detección de patrones de diseño de software

Los Agentes de Interfaz ofrecen asistencia personalizada en software. Este trabajo propone usar Modelos de Maikou para detectar patrones de diseño en herramientas CASE, mejorando la precisión y tiempos de respuesta, y generando recomendaciones basadas en las acciones del usuario.

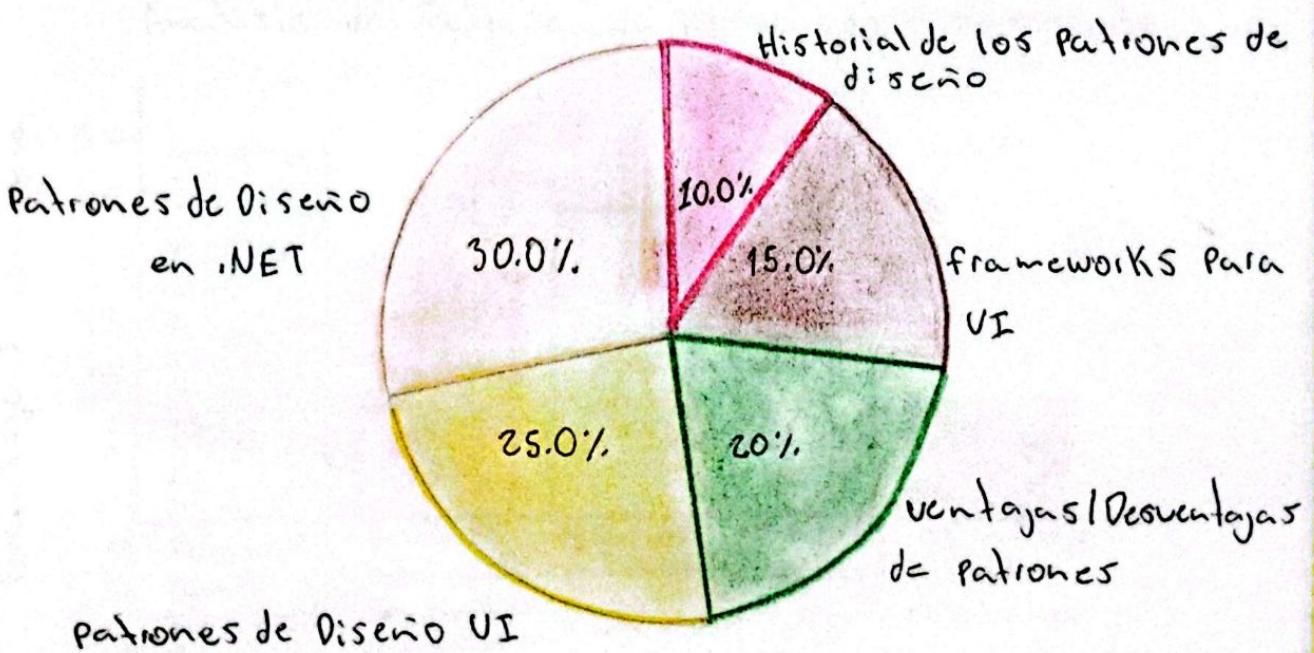
Frecuencia de Temas en el Artículo de Arquitecturas de Referencia de software



## Patrones de Diseño

Los patrones de diseño, popularizados por el libro *Design Patterns* en 1994, proporcionan soluciones reutilizables para crear software flexible. Este documento aborda dos tipos de patrones: los de diseño para la implementación en código, explicando seis patrones usados en .NET, y los de diseño de interfaz de usuario, enfocados en mejorar la experiencia del usuario en sitios web, con ejemplos y frameworks recomendados para facilitar el desarrollo.

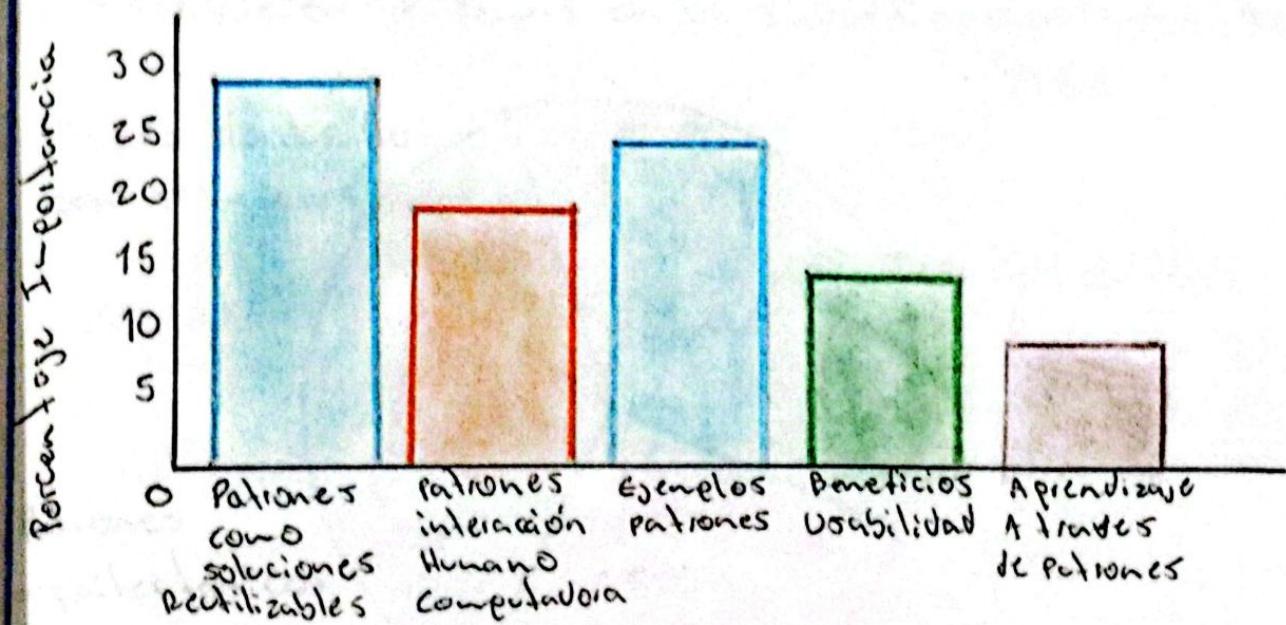
## Análisis de Patrones de Diseño y UI



## Patrones de Diseño: Mejores Experiencias y Productos Fácilmente Escalables

Los patrones de diseño en experiencias digitales facilitan el uso para los usuarios, quienes ya están familiarizados con ciertos estándares. Siguiendo estos patrones, se mejora la usabilidad y se evita una curva de aprendizaje empinada, lo que ayuda a que los usuarios se concentre en el contenido del sitio en lugar de cómo usarlo. Esto también aumenta la eficiencia del equipo de desarrollo.

### Ánalisis de Patrones de Diseño Segun el Artículo

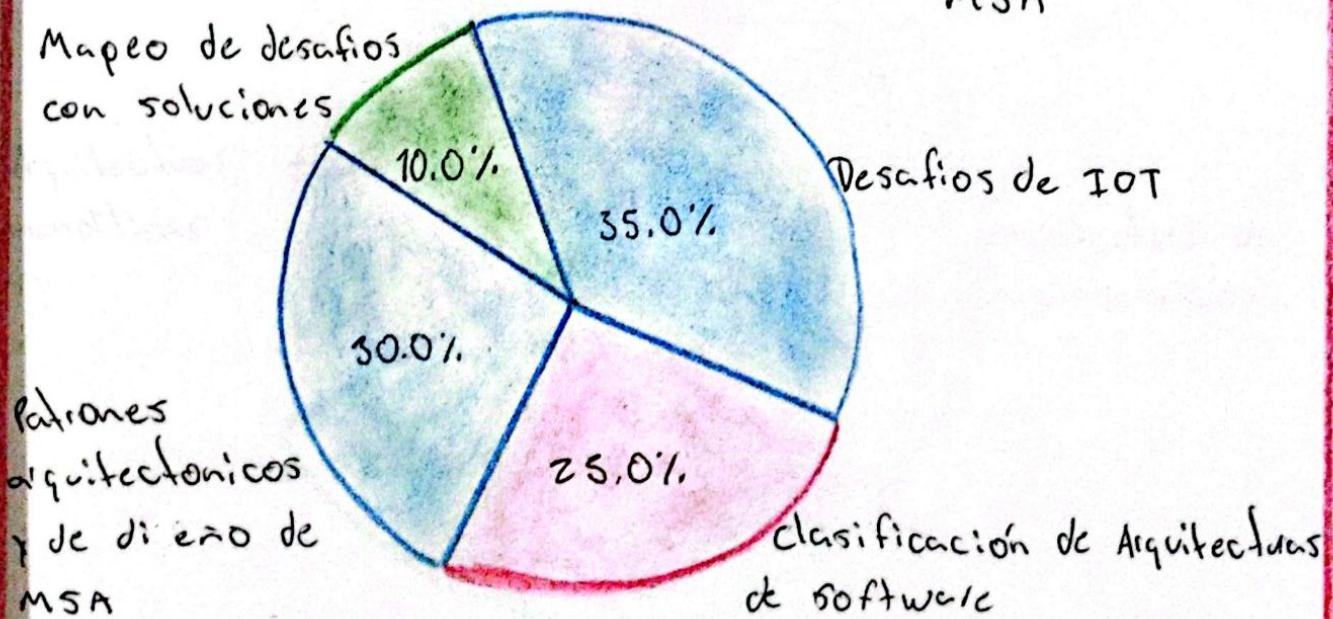


Aspectos Relevantes

# A Systematic Review on Software Architectures for IOT Systems and future Direction to the Adoption of Microservices Architecture

Este estudio revisa la investigación sobre la adopción de la Arquitectura de Microservicios (MSA) en sistemas de Internet de las cosas (IOT). Analiza 140 artículos para identificar patrones arquitectónicos, desafíos y soluciones en IOT. Destaca trece patrones de MSA, clasifica arquitecturas en nueve categorías, y aborda veintidós desafíos principales. Los hallazgos ofrecen recomendaciones para mejorar la arquitectura y el desarrollo de software de IOT, beneficiando a la industria y a la comunidad académica.

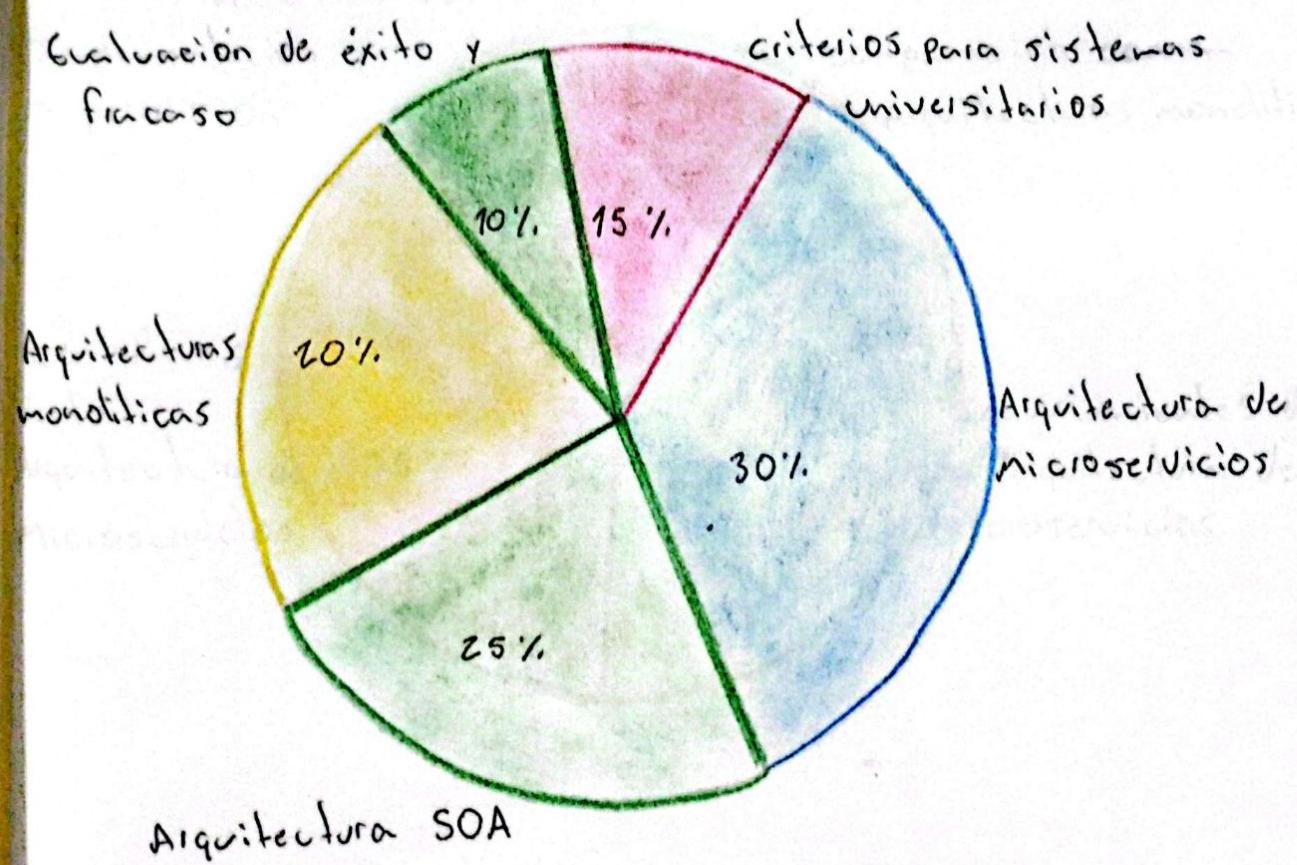
Distribución de Temas en la Investigación sobre IOT y MSA



## Software Architectures - Present and Visions

El artículo analiza diferentes tipos de arquitecturas de sistema de software, como monolíticas, SOA y microservicios, y evalúa cuál es más adecuada para sistemas universitarios. Examina desafíos específicos, factores de éxito y causas de fracaso, y ofrece una recomendación objetiva para la implementación en el ámbito académico.

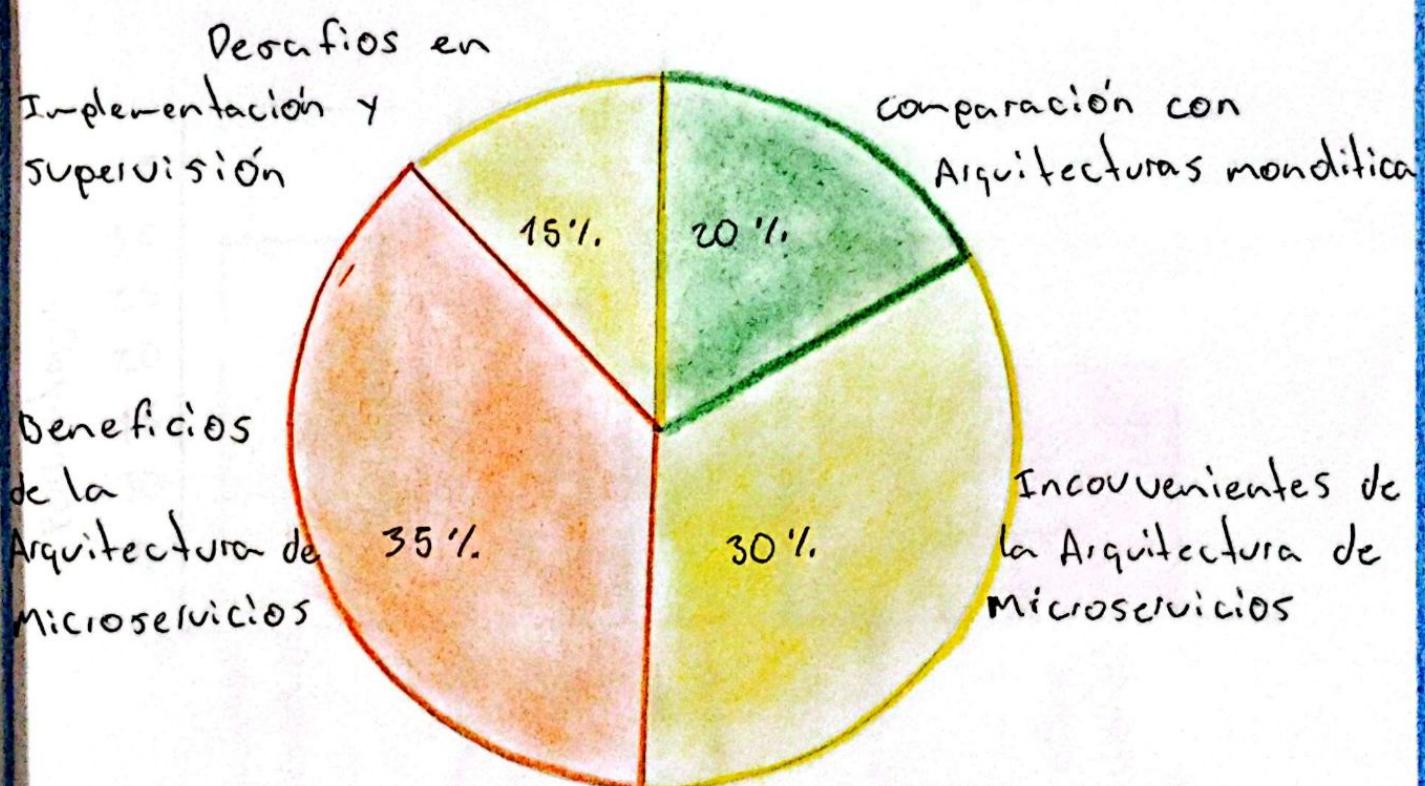
### Distribución de Temas en la investigación sobre Arquitecturas de Software para Sistemas universitarios



A scalable architecture for automated monitoring of microservices

El artículo propone una arquitectura para monitorear microservicios, registrando métricas clave de rendimiento y utilizando algoritmos avanzados para clasificar eventos y automatizar decisiones. Utiliza Prometheus para métricas a corto plazo, OpenTSDB para almacenamiento a largo plazo y RabbitMQ para la transmisión de mensajes.

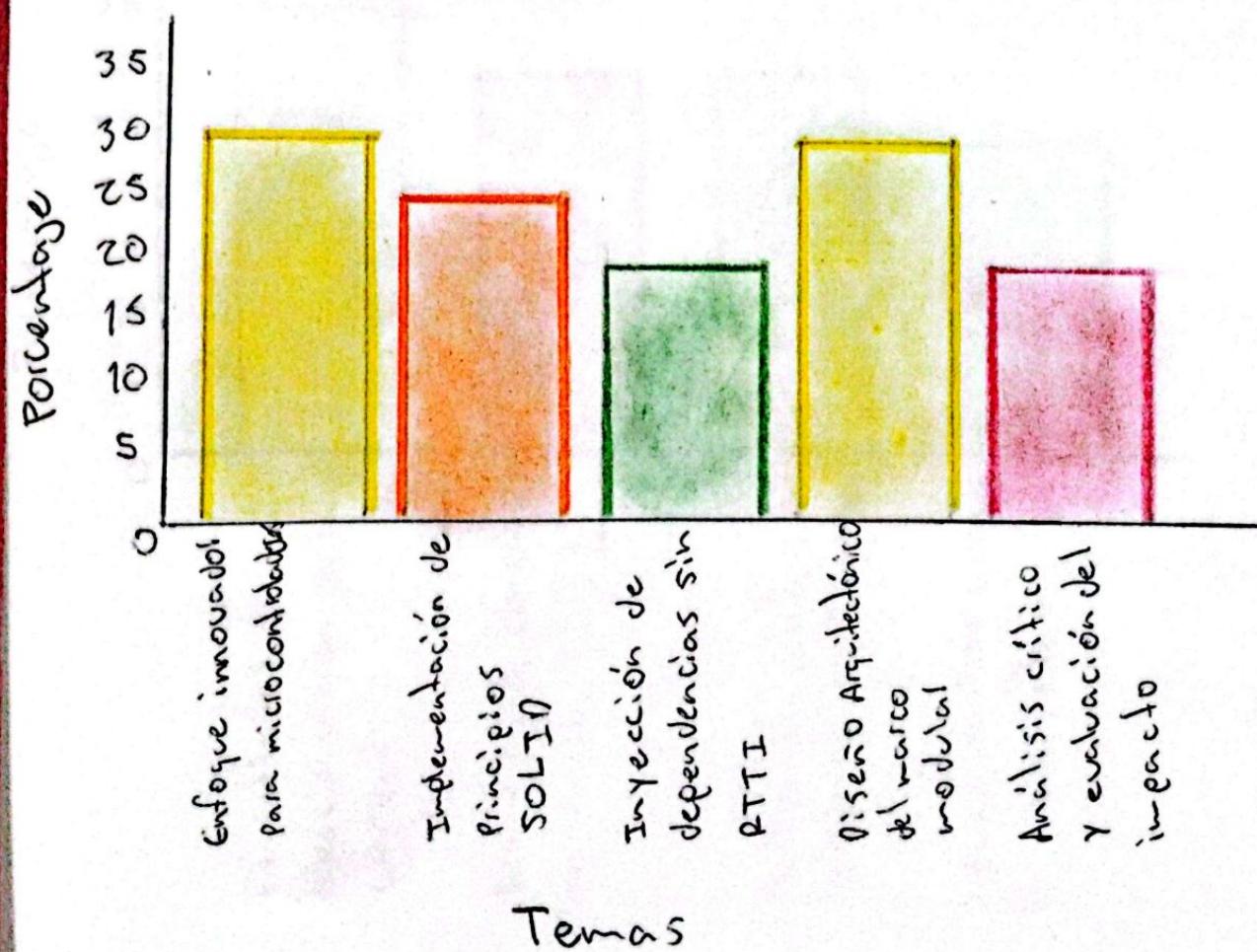
Distribución de Temas en la Arquitectura de microservicios vs. monolítica



## Implementation of a Universal framework Using Design Patterns for Application Development on Microcontrollers

El artículo presenta un marco modular para microcontroladores que aplica principios modernos y patrones de diseño para mejorar la calidad del software en sistemas integrados e IOT. Este enfoque permite la comunicación independiente entre módulos, integración flexible de hardware y adaptabilidad a diferentes entornos de desarrollo, mejorando la sostenibilidad y escalabilidad a largo plazo.

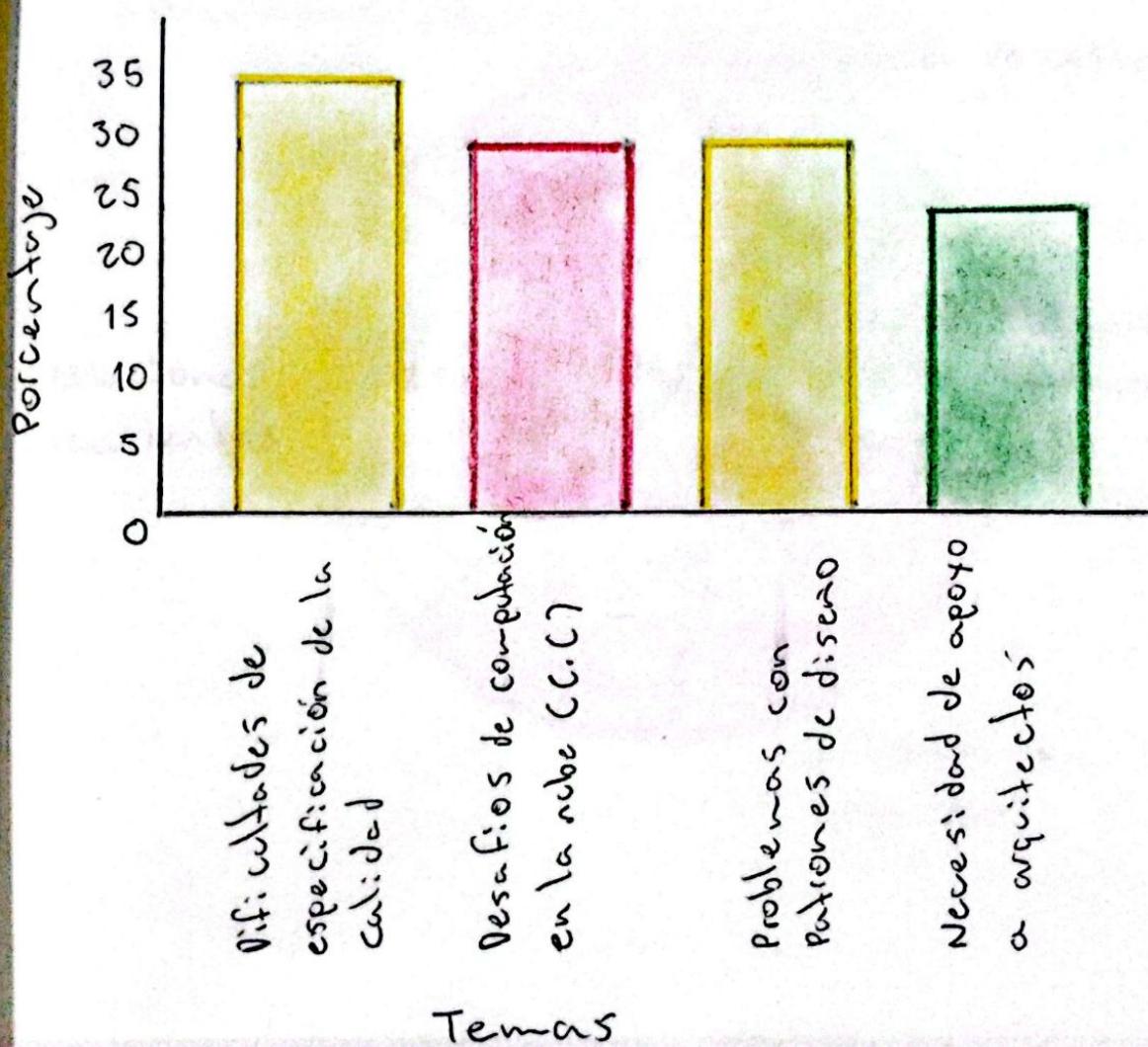
### Distribución de Temas en el Desarrollo de Microcontroladores



## Modelado y verificación de Patrones de Diseño de Arquitectura de software para entornos de Computación en la Nube

El trabajo presenta una herramienta gráfica para diseñar arquitecturas de software en aplicaciones web, basada en patrones de diseño. Se enfoca en cómo las funcionalidades del software se relacionan los atributos de calidad y utiliza el paradigma de computación en la nube para ofrecer acceso a recursos configurables de manera eficiente.

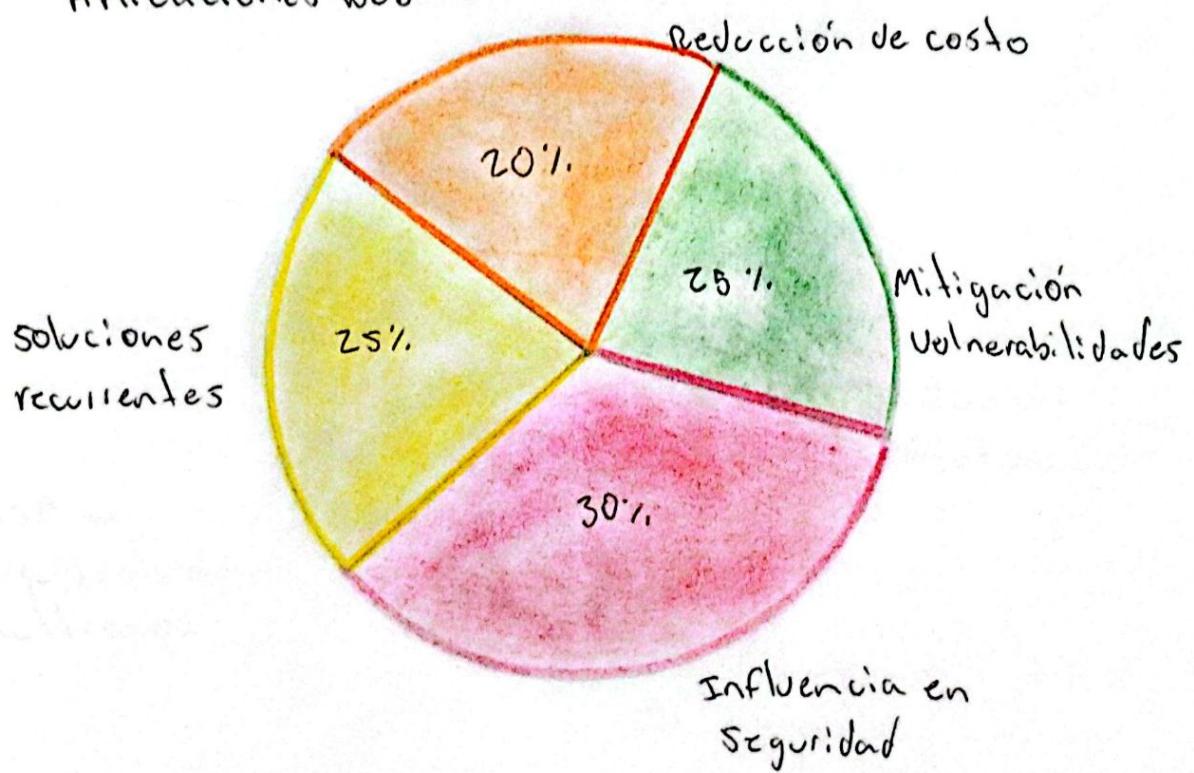
### Distribución de Temas en el futuro de la calidad del software en CC



## Incidencia de los Patrones de Diseño de software en la Seguridad de Aplicaciones web

los patrones de diseño ayudan a estructurar y organizar el código de manera eficiente, y pueden mejorar la seguridad de aplicaciones web al abordar problemas como validación de datos y autenticación. La investigación bus... determinó como estos patrones ayudan a mitigar vulnerabilidades, reduciendo costos y mejorando la mantenibilidad del software

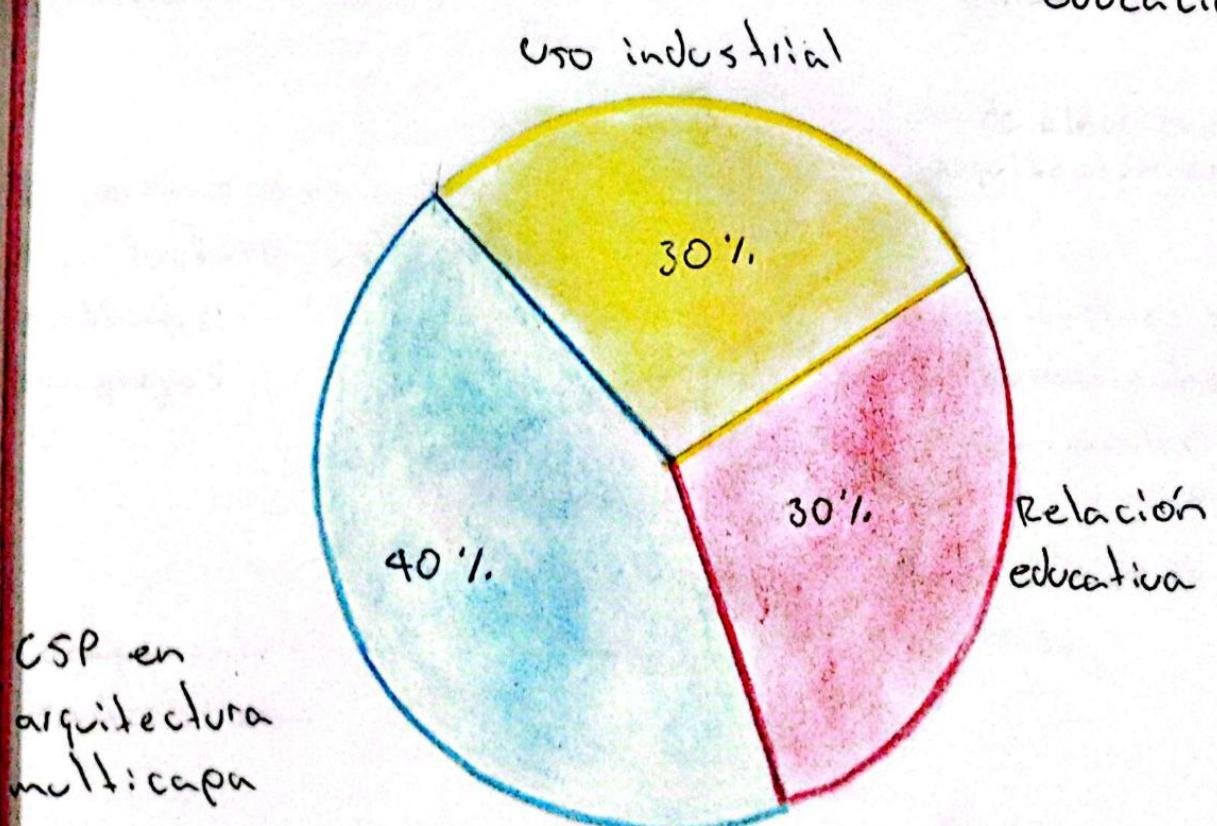
## Influencia de los Patrones de Diseño en la seguridad de Aplicaciones web



## Integración de patrones de seguridad y patrones de diseño J2EE

El trabajo analiza dos catálogos populares de patrones de diseño de software de Sun Microsystems (oracle), define sus relaciones para un uso coherente y presenta casos prácticos de aplicaciones que integran patrones de ambos catálogos.

## Influencia de los Patrones de Diseño en Seguridad y Educación



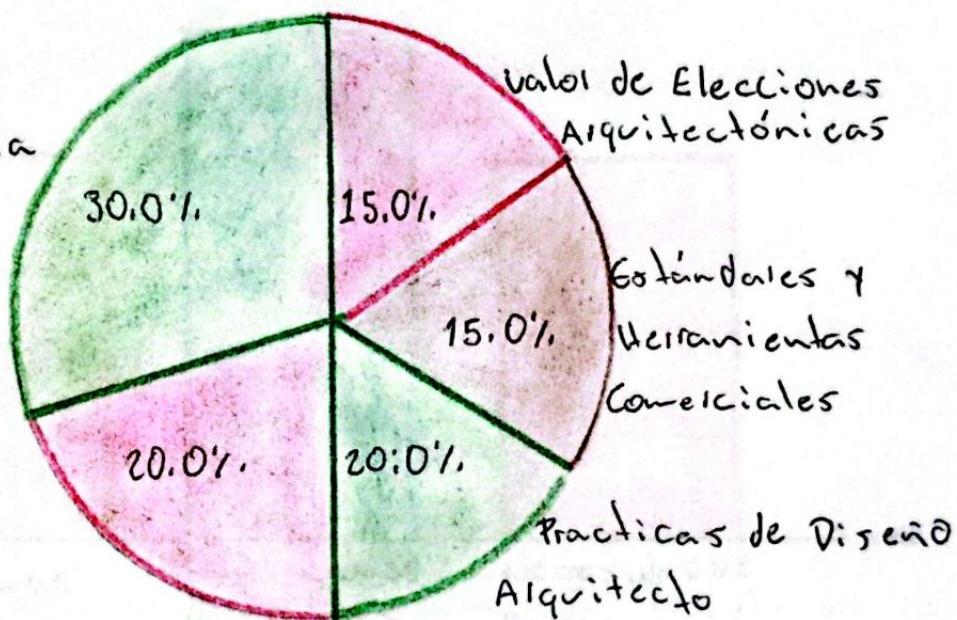
## Software Architecture

El diseño arquitectónico en software es crucial para manejar la creciente complejidad de los sistemas. Una buena arquitectura garantiza requisitos clave como rendimiento, escalabilidad y confiabilidad. En la última década, ha ganado importancia con la creación de libros, herramientas, cursos y estándares formales que ayudan a reutilizar y mejorar los diseños arquitectónicos.

## Analisis de la Arquitectura de Software

Importancia de la  
Arquitectura en  
sistemas  
complejos

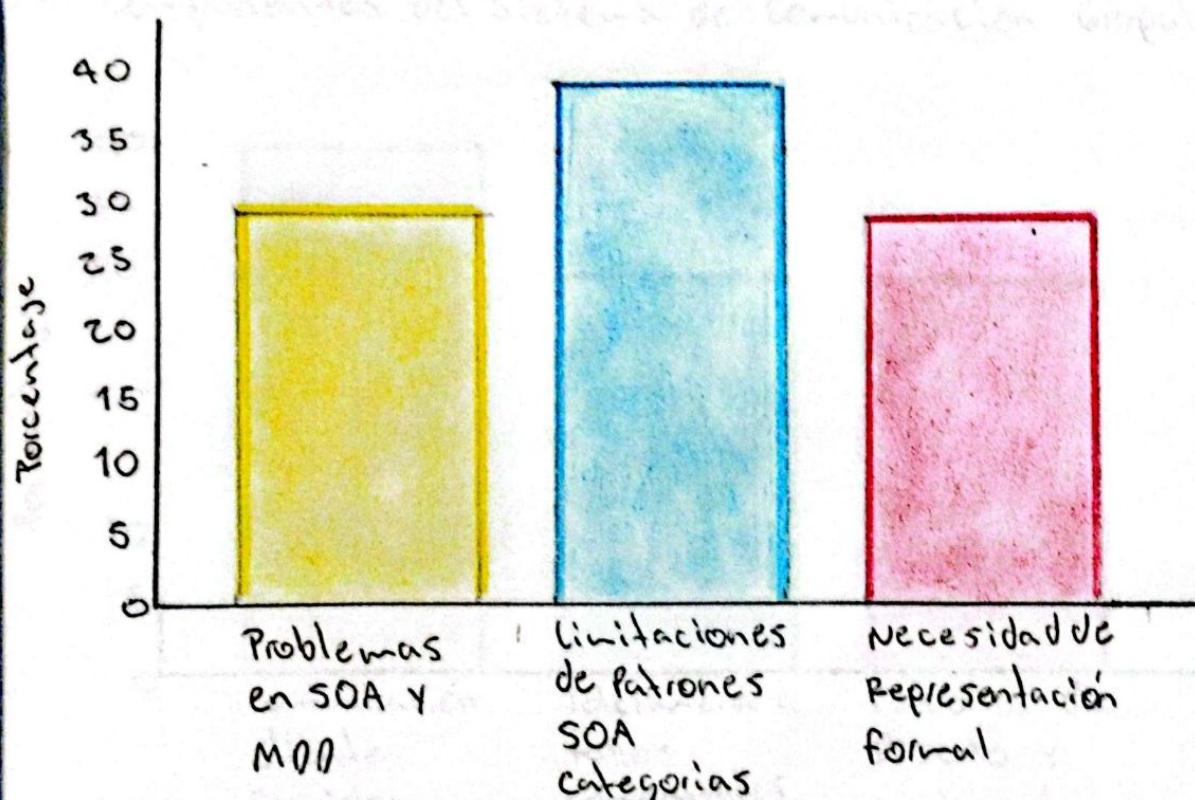
componentes y  
control global



## Application of Design Patterns in Service Oriented Architecture

El artículo investiga cómo los patrones de diseño informales de Arquitectura Orientada a Servicios (SOA) quedan resueltos problemas en el desarrollo de software basado en modelos. Utiliza análisis orientado a objetos y teorías de categorías para formalizar estos patrones, mejorando su aplicación en el modelado de sistemas.

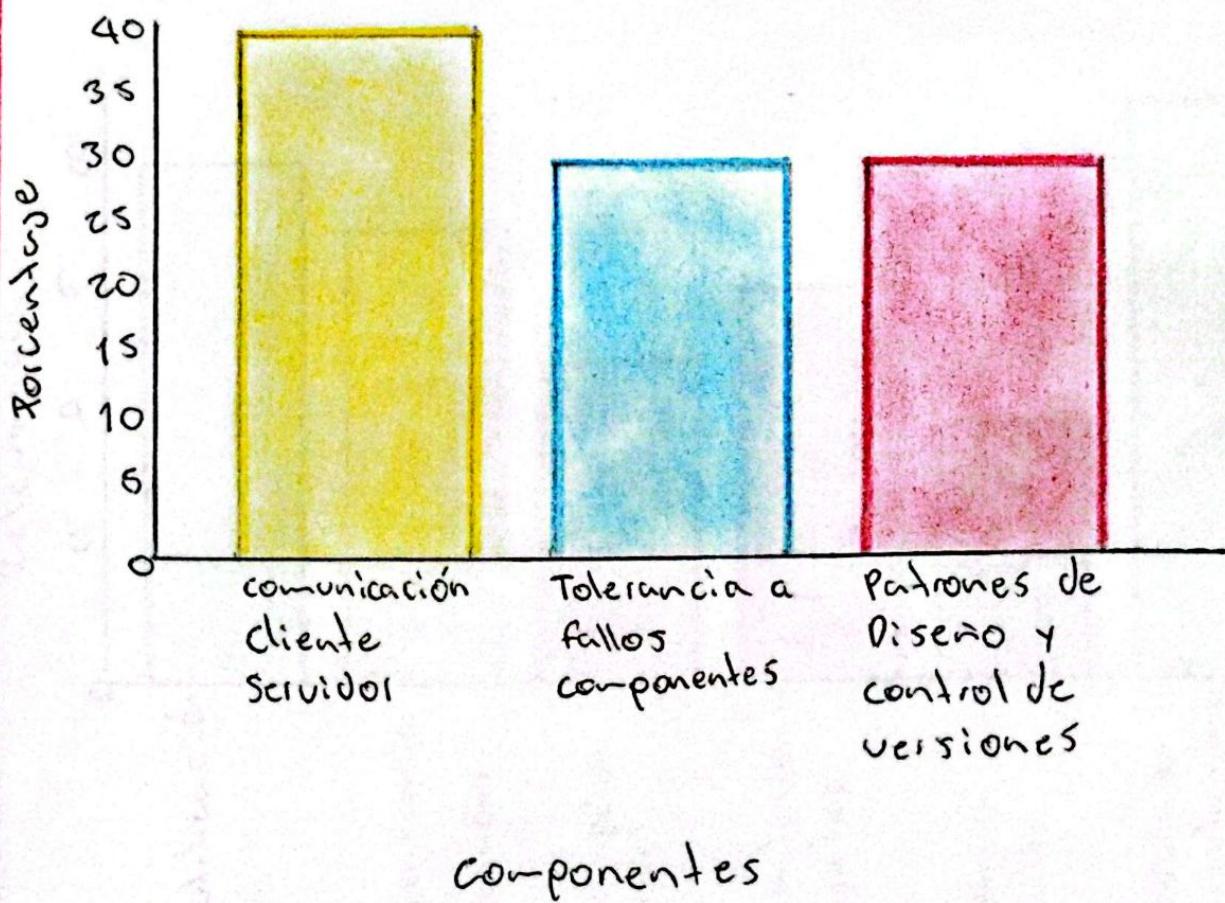
### Desafíos en la Aplicación de Patrones de Diseño en SOA



## Design and Implementation of a Java-Based Client Server Application

El informe describe el desarrollo del sistema de comunicación grupal (GCS) en java que utilizan sockets y protocolos de comunicación para gestionar grupos a través de una interfaz de línea de comandos. Destaca la tolerancia a errores, patrones de diseño y uso de control de versiones, proporcionando una visión completa de su arquitectura e implementación.

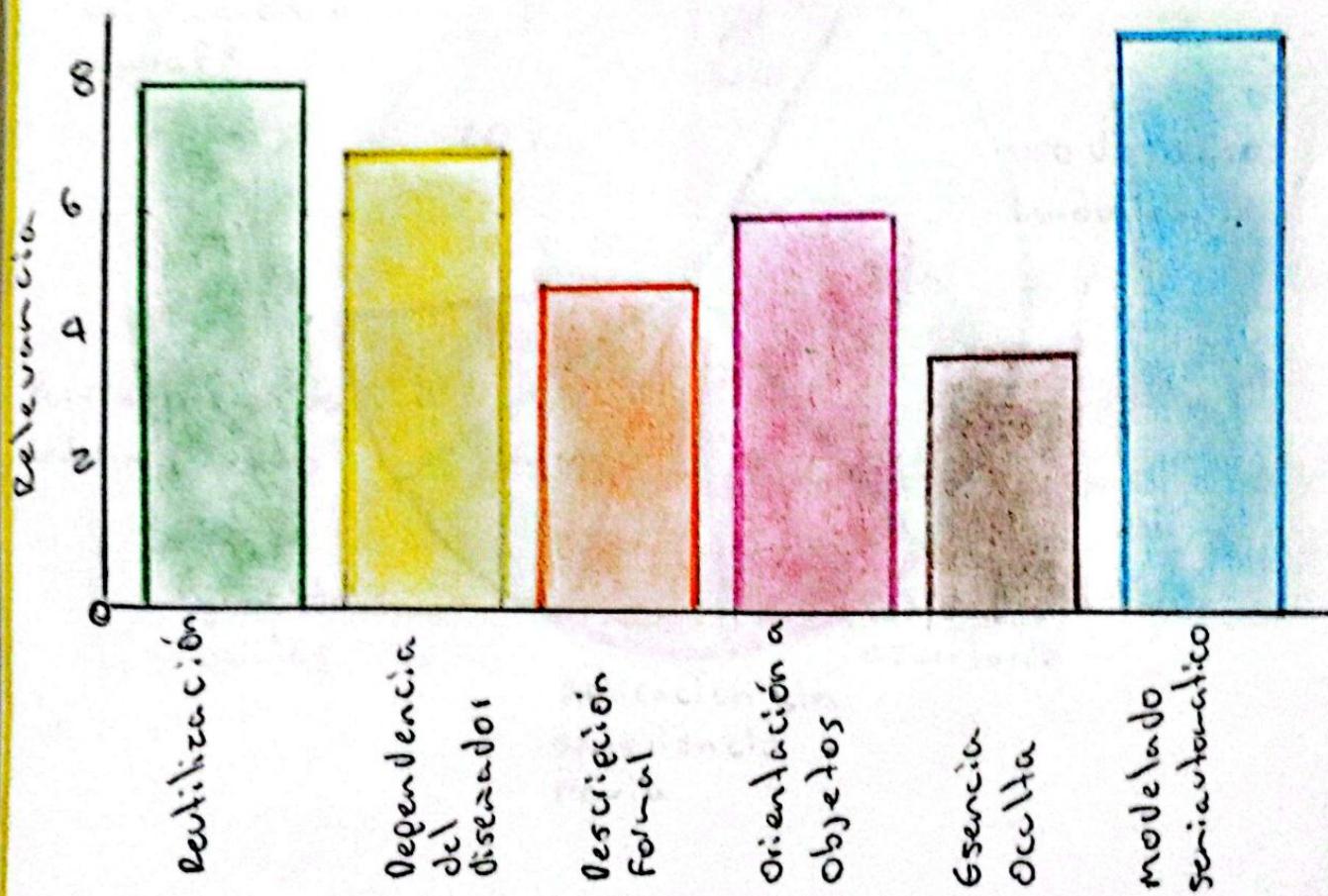
### Componentes del Sistema de Comunicación Grupal (GCS)



## Modeling Design Patterns for Semi-Automatic reuse in System Design

Los patrones de diseño ofrecen soluciones reutilizables para problemas comunes en el desarrollo de software, pero su diseño depende del diseñador. A menudo, se describen de manera real, informal o demasiado orientada a objetos. El artículo propone un enfoque para modelar estos patrones y utilizarlos semiautomáticamente para mejorar el diseño del software, mostrando que este método ayuda a identificar problemas clave en aplicaciones prácticas.

### ANÁLISIS DE LOS PATRONES DE DISEÑO SEGÚN EL TEXTO

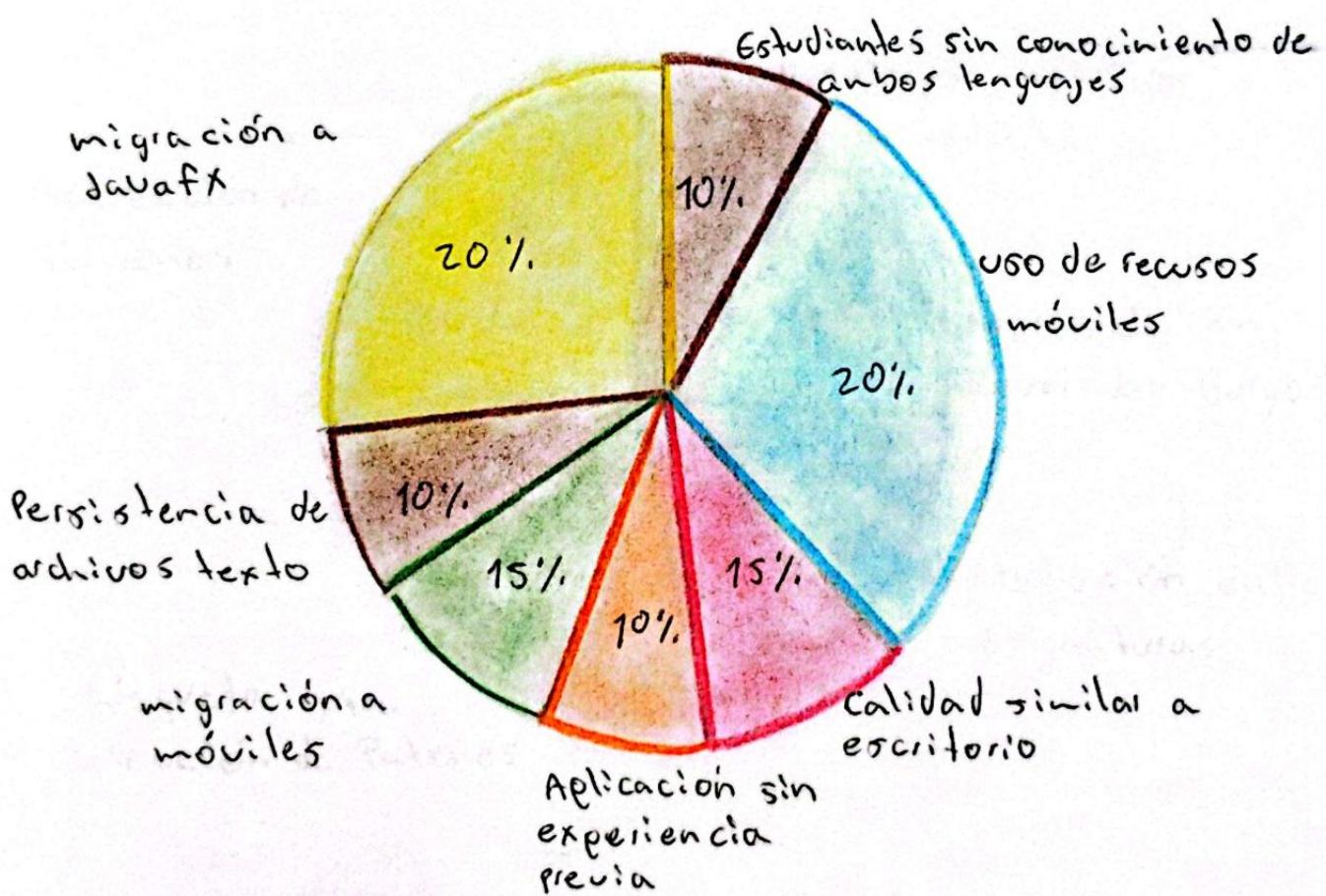


Aspectos de los Patrones de Diseño

## Convertir aplicaciones de escritorio en aplicaciones móviles con Java

El artículo presenta un proceso para convertir aplicaciones J2SE a JavaFX, enfocándose en aplicaciones locales que gestionan archivos de texto y se desean migrar a móviles. Proporcionan pautas para crear aplicaciones JavaFX desde cero, manteniendo la calidad y funcionalidad de las versiones de escritorio, se aplicó en un curso avanzado en UNININTO, logrando una conversión exitosa sin necesidad de experiencia previa en JavaFX.

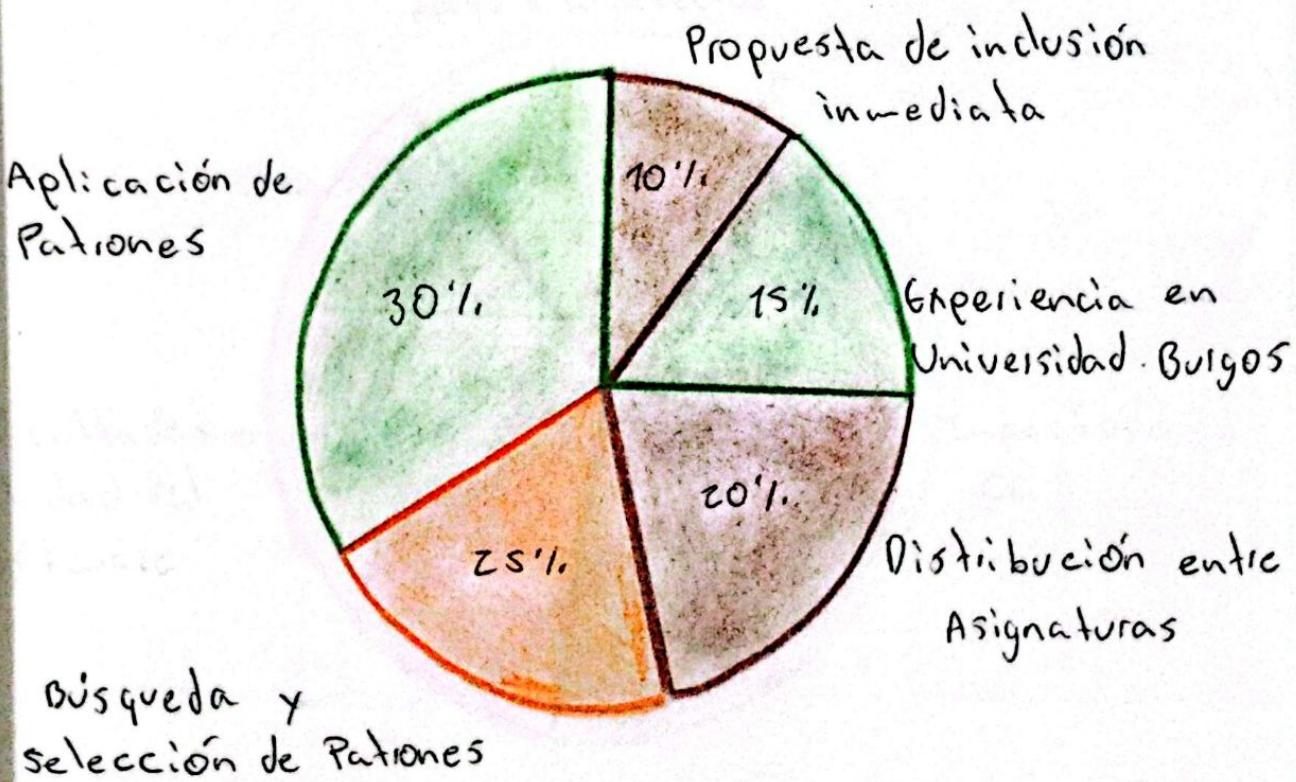
### Analisis de la migración de J2SE a Java



## Inclusión de patrones de diseño en un plan de estudios de Ingeniería Técnica en Informática de Gestión

El trabajo propone incluir patrones de diseño en el currículo de Ingeniería Técnica en informática de gestión (ITIG), diferenciando entre la aplicación y la selección de patrones. Se basa en la experiencia previa de la Universidad de Burgos y sugiere una integración adecuada de estos conceptos en las asignaturas del plan de estudios.

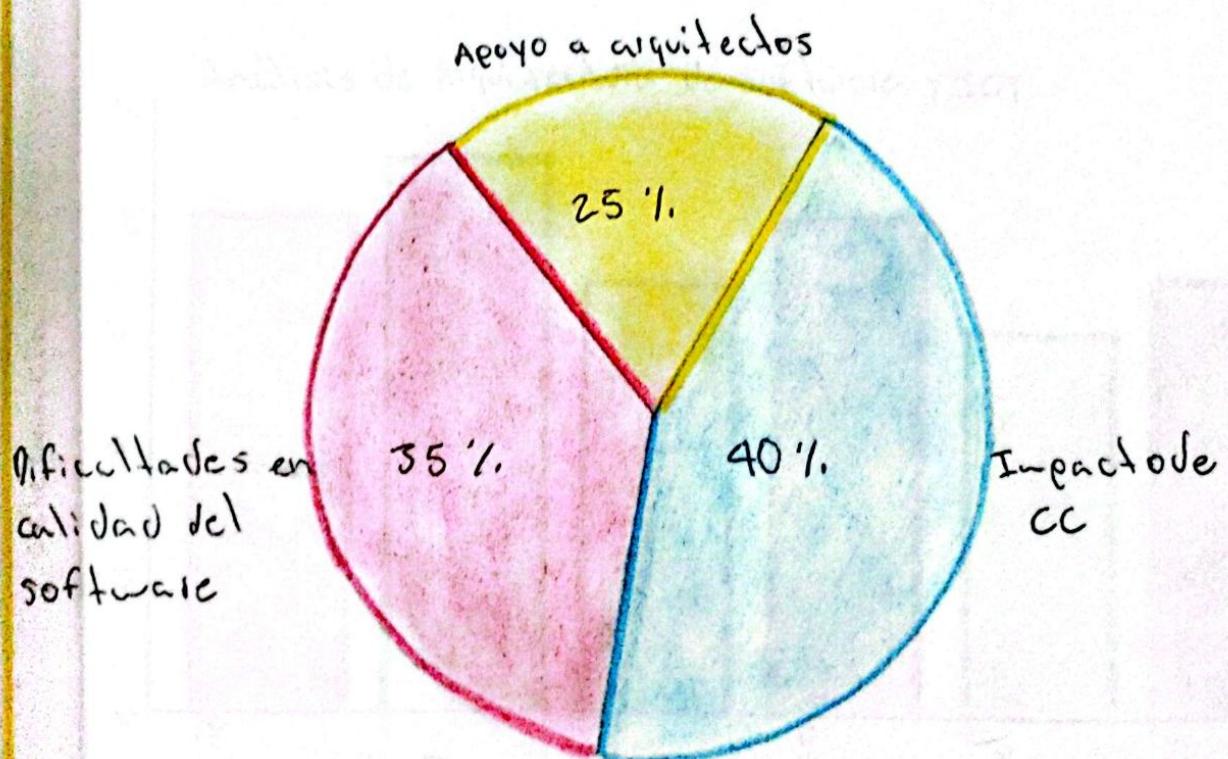
### Ánalisis sobre la inclusión de Patrones de Diseño en ITIG



## Modelado y Verificación de Patrones de Diseño de Arquitectura de software para Entornos de computación en la Nube

El trabajo presenta una herramienta gráfica para diseñar arquitecturas de software en aplicaciones web, utilizando un modelo de soporte. Destaca que las funcionalidades del software se separan de los atributos de calidad y que la computación en la nube facilita el acceso a recursos configurables de manera eficiente.

## Dificultos en la Evaluación de la calidad del software en la computación en la Nube



# A Systematic Review on Software Architectures for IOT Systems and future Direction to the Adoption of Microservices Architecture

Este estudio revisa la investigación sobre la adopción de la Arquitectura Micoservicios (MSA) en sistema de Internet de las cosas (IOT). Analiza 140 artículos para identificar patrones arquitectónicos, desafíos y soluciones en IOT. Destaca trece patrones de MSA, clasifica arquitecturas en nueve categorías, y aborda veintidós desafíos principales. Los hallazgos ofrecen recomendaciones para mejorar la arquitectura y el desarrollo de software de IOT, beneficiando a la industria y a la comunidad académica.

