

# **Bases Conceptuales Acerca del Lenguaje Unificado de Modelado (UML) y Patrones de Diseño**

**GA4-220501095-AA2-EV03**



**Isidro J Gallardo Navarro**

**Ficha:3070299**

**2025**

**Tecnología en Análisis y Desarrollo de  
Software.**

**ADSO**

Lista de chequeo a cubrir:

- Elaborar de forma correcta el resumen de UML abarcando la mayoría de información del tema.
- Elaborar un Glosario de terminología UML.

## Introducción

### historia importancia

En el mundo actual del desarrollo de software, la complejidad de los sistemas informáticos ha crecido exponencialmente. Desde aplicaciones móviles simples hasta sistemas empresariales complejos que manejan millones de transacciones diarias, la necesidad de comunicar ideas, diseños y arquitecturas de manera clara y universal se ha vuelto fundamental. Es aquí donde el Lenguaje Unificado de Modelado (UML) emerge como una herramienta indispensable para profesionales de la tecnología.

UML no es simplemente un conjunto de diagramas o símbolos; es un lenguaje visual estandarizado que permite a arquitectos de software, desarrolladores, analistas de sistemas y stakeholders del negocio comunicarse efectivamente, independientemente de su trasfondo técnico o cultural. Al igual que los planos arquitectónicos permiten a ingenieros, constructores y propietarios entender cómo será un edificio antes de construirlo, UML proporciona los "planos" para el software antes de escribir la primera línea de código.

La importancia de UML radica en su capacidad para reducir la ambigüedad en la comunicación técnica, minimizar errores costosos durante el desarrollo y facilitar el mantenimiento a largo plazo de los sistemas. En un proyecto de software típico, los malentendidos en los requerimientos pueden representar hasta el 60% de los defectos del producto final. UML ayuda a mitigar estos riesgos proporcionando una representación visual clara y precisa de lo que se va a construir.

En esta actividad, exploraremos los fundamentos de UML desde una perspectiva práctica y aplicada. Comenzaremos entendiendo su definición y propósito, para luego sumergirnos en los conceptos de orientación a objetos que forman su base conceptual. A través de ejercicios prácticos y casos de estudio, desarrollaremos las habilidades necesarias para crear diagramas de clases efectivos y establecer relaciones apropiadas entre los componentes de un sistema.

## Objetivos de Aprendizaje

### Objetivo General

Comprender los fundamentos del Lenguaje Unificado de Modelado (UML) y su aplicación en el desarrollo de sistemas de software, identificando sus elementos principales y las relaciones entre clases.

## Objetivos Específicos

Definir qué es UML y su propósito en el desarrollo de sistemas  
Identificar el alcance y los beneficiarios del estándar UML  
Relacionar UML con los conceptos de Programación Orientada a Objetos  
Representar clases utilizando la notación UML estándar  
Distinguir y aplicar los diferentes tipos de relaciones entre clases  
utilizar correctamente la terminología específica de UML

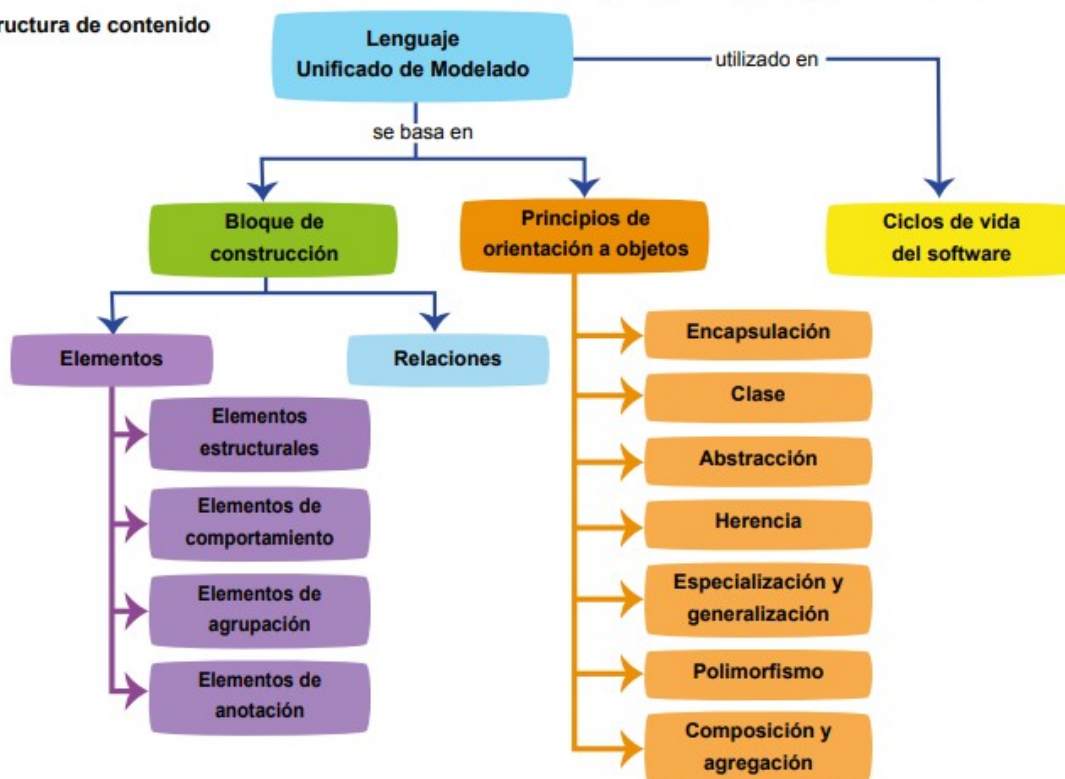
## Marco Teórico

### 1. Definición y Propósito de UML

El Lenguaje Unificado de Modelado (UML) es un estándar gráfico internacional utilizado para especificar, visualizar, construir y documentar sistemas complejos. Su importancia radica en proporcionar un conjunto de buenas prácticas de ingeniería que facilitan la comunicación entre equipos de desarrollo y stakeholders.

UML no es un lenguaje de programación, sino una notación visual que permite modelar sistemas de cualquier complejidad mediante diagramas, elementos y relaciones claramente definidos.

### 2. Estructura de contenido



## **2. Alcance y Beneficiarios**

Según la Object Management Group (OMG), organización responsable del estándar UML, este lenguaje está diseñado específicamente para:

Arquitectos de sistemas: Profesionales encargados del diseño de la estructura general del sistema

Ingenieros de software: Desarrolladores responsables de la implementación técnica Analistas de

negocios: Especialistas en modelado de procesos empresariales

Las aplicaciones principales incluyen análisis, diseño e implementación de sistemas basados en software, así como modelado de procesos de negocio, El estándar es ampliamente utilizado en diferentes procesos de modelado y desarrollo; y con otros tipos de arquitecturas gracias a su amplio alcance y conjunto de elementos, relaciones, bloques y diagramas que posibilitan la extensibilidad, flexibilidad y robustez del lenguaje. 4. Alcance del UML De acuerdo con la Organización de Administración de Objetos (en inglés, Object Management Group - OMG) el lenguaje unificado de modelado define su alcance de la siguiente manera: "El objetivo del UML es proveer a arquitectos de sistemas, ingenieros de software y desarrolladores de software con herramientas para análisis, diseño e implementación de sistemas basados en software, modelado de negocios y procesos similares" (Object Management Group, 2015) 5. Introducción a la orientación a objetos Antes de poder empezar a conocer los elementos base utilizados dentro de las buenas prácticas y el estándar UML como tal, se hace necesario abordar el tema de la orientación a objetos ya que este es más antiguo que el lenguaje (UML).

## **3. Fundamentos de Orientación a Objetos**

UML está intrínsecamente ligado a la Programación Orientada a Objetos (POO). Los conceptos fundamentales incluyen:

Objeto

Elemento identificable del mundo real que posee características específicas y puede realizar acciones.

Ejemplos: un libro, un teléfono, un estudiante.

Clase

Representación abstracta que agrupa objetos con atributos y comportamientos similares. Define la estructura y funcionalidad común de un conjunto de objetos.

Clase Conceptual

Tipo especial de clase que únicamente define atributos, sin incluir comportamientos o métodos. Se utiliza principalmente en las fases iniciales del análisis.

#### **4. Representación de Clases en UML**

La notación estándar para representar una clase en UML consiste en un rectángulo dividido en tres secciones horizontales:

Sección superior: Contiene el nombre de la clase

Sección media: Lista los atributos con su visibilidad, nombre y tipo de dato

Sección inferior: Enumera los métodos y comportamientos de la clase

#### **Terminología Esencial**

Candidato: Elemento que podría ser definitivo o modelo a seguir en el diseño

Dominio: Área específica de conocimiento donde se desarrolla el sistema  
Requerimiento: Solicitud o necesidad expresada por el cliente o usuario

Artefacto: Componente construido con un propósito específico dentro del sistema  
Estándar: Norma o modelo de referencia aceptado universalmente

Modelo: Representación simplificada que facilita la comprensión del sistema  
Notación: Sistema de símbolos y reglas para representar conceptos de manera consistente

#### **Símbolos de Visibilidad en UML**

+ (público): Accesible desde cualquier clase

- (privado): Accesible solo dentro de la misma clase

# (protegido): Accesible en la clase y sus subclases

~ (paquete): Accesible dentro del mismo paquete

#### **References**

Moreno, F. (1999). Introduccion a la OOP (pimera ed., Vol. 1).

Principio del Analisis y Dise~no orientado a Objetos Utilizando el Estandar UML (segunda ed.). (2006).

Felix Ramon Triana Gaitan.