



Bases Conceptuales Acerca Del Lenguaje Unificado De Modelado (UML) Y Patrones De Diseño

Yamid Ribero Rivero
Aprendiz

Reinaldo Martínez Luna
Instructor

Tecnología en Análisis y Desarrollo de Software
Ficha 2722305
Sena, Virtual

INTRODUCCIÓN

En este documento se realizará un informe sobre el Lenguaje Unificado De Modelado (UML), es un lenguaje estándar utilizado para visualizar, especificar, construir, y documentar los artefactos de un sistema de software. Proporciona una notación grafica para representar diferentes aspectos de un sistema, como la estructura, el comportamiento y las interacciones entre los componentes del sistema.

Informe Del Lenguaje Unificado De Modelado UML

El Lenguaje Unificado de Modelado (UML, por sus siglas en inglés) es un lenguaje de modelado visual utilizado para representar y comunicar diseños de software. El Lenguaje Unificado de Modelado (UML) es una herramienta poderosa que puede ayudar a mejorar el proceso de desarrollo de software. Fue desarrollado en la década de 1990 por un consorcio de empresas de software y se ha convertido en el estándar de facto para el modelado de sistemas de software.

El UML proporciona una serie de diagramas que permiten representar diferentes aspectos de un sistema, como su estructura, comportamiento, interacciones y procesos. Algunos de los diagramas más comunes en UML son:

Diagrama de clases: muestra la estructura estática del sistema, con sus clases, atributos y relaciones.

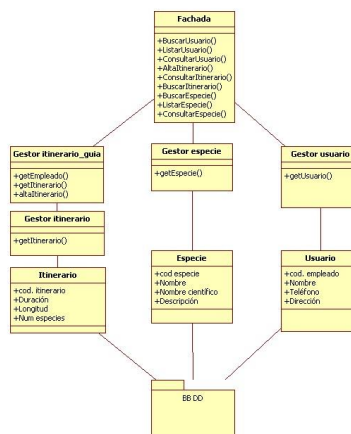


Diagrama de objetos: Representa instancias de clases y sus relaciones en un momento específico del tiempo.

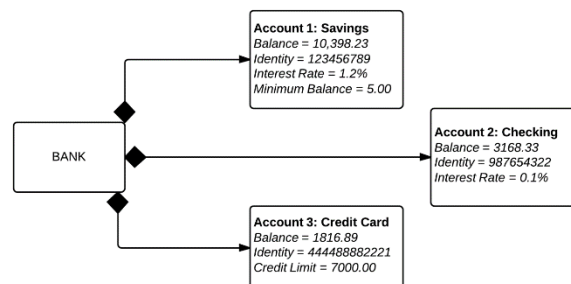


Diagrama de secuencia: muestra la interacción entre diferentes objetos en una secuencia de mensajes.

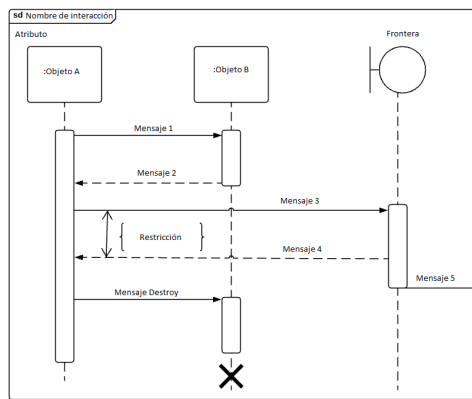


Diagrama de actividad: Describe el flujo de actividades o procesos dentro del sistema.

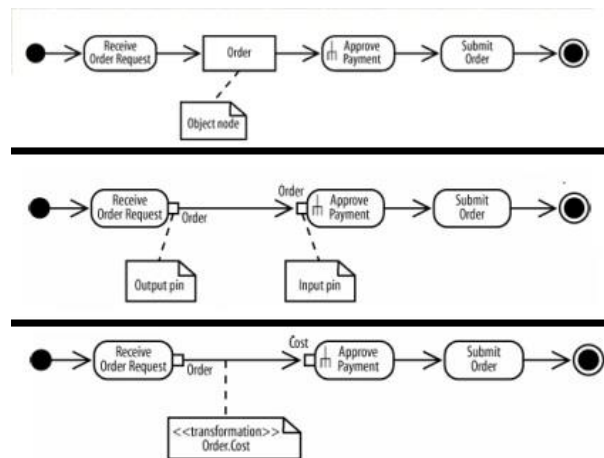


Diagrama de estado: representa los diferentes estados en los que puede estar un objeto y las transiciones entre ellos.

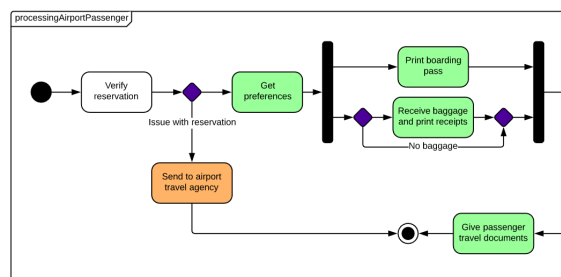


Diagrama de componentes: muestra los componentes del sistema y las dependencias entre ellos.

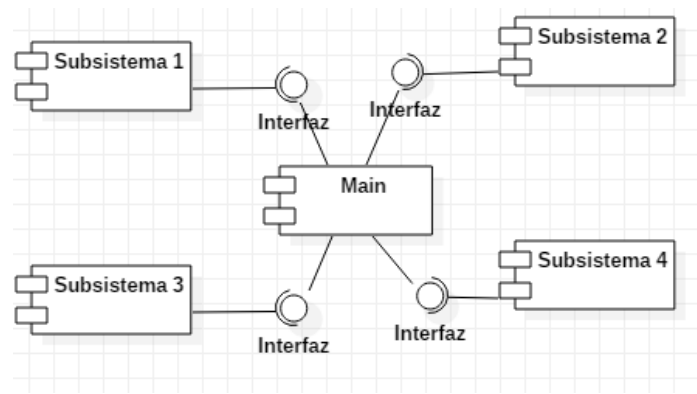
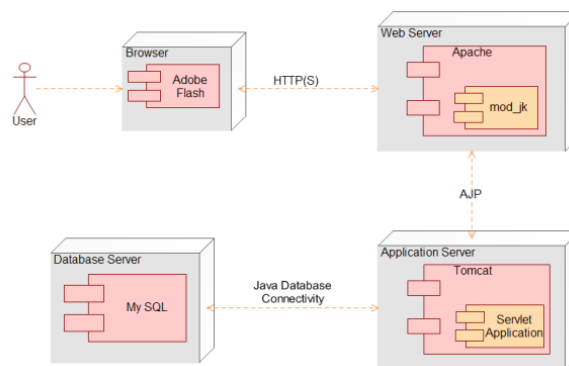


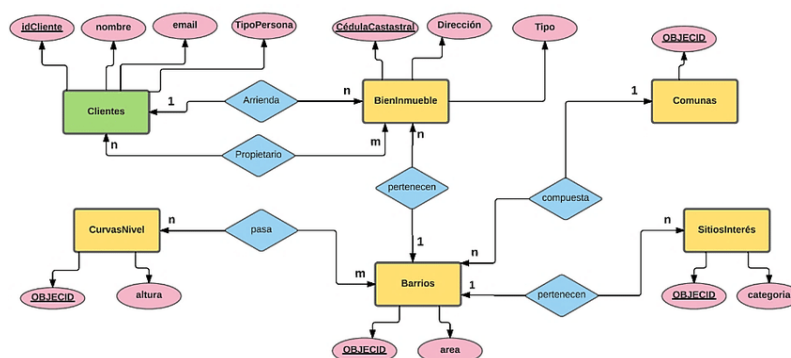
Diagrama de despliegue: representa la configuración física del sistema, incluyendo los nodos de hardware y software.



El objetivo principal del UML es proporcionar un lenguaje común y un conjunto de herramientas para que los desarrolladores, diseñadores y stakeholders puedan comunicar y entender los diseños de software de manera efectiva. El UML se utiliza tanto en el análisis y diseño de sistemas como en el desarrollo de software.

DISEÑO DEL MODELO CONCEPTUAL

El diseño de un modelo conceptual es una etapa fundamental en el desarrollo de un sistema de software. Es un proceso de abstracción que permite identificar y definir los elementos básicos del sistema, como las entidades, las relaciones, las clases, los atributos y los métodos.



Objetivos Del Diseño Del Modelo Conceptual:

- Comprender los requisitos del sistema.
- Definir la estructura del sistema.
- Facilitar la comunicación entre los diferentes stakeholders del proyecto.
- Servir como base para el diseño detallado del sistema.

Pasos Para El Diseño Del Modelo Conceptual:

- **Identificar las entidades:** Las entidades son los objetos o conceptos principales sobre los que se trata el sistema. Se pueden identificar buscando sustantivos en el texto de los requisitos.
- **Identificar las relaciones:** Las relaciones son las conexiones que existen entre las entidades. Se pueden identificar buscando verbos en el texto de los requisitos que describen cómo las entidades interactúan entre sí.
- **Definir las clases:** Las clases son los tipos de objetos que se crearán en el sistema. Cada clase debe representar una entidad o un conjunto de entidades relacionadas.
- **Definir los atributos:** Los atributos son las características de los objetos. Cada atributo debe tener un nombre, un tipo de dato y un valor.
- **Definir los métodos:** Los métodos son las acciones que pueden realizar los objetos. Cada método debe tener un nombre, una lista de parámetros y un tipo de dato de retorno.
- **Definir las relaciones entre las clases:** Las relaciones entre las clases pueden ser de herencia, asociación o agregación.
- **Representar el modelo conceptual mediante un diagrama de clases:** Un diagrama de clases es una representación gráfica del modelo conceptual.

RESUMEN UML

UML (Lenguaje Unificado de Modelado), es un lenguaje gráfico para visualizar, especificar y documentar los componentes de un sistema de software. Es un lenguaje estándar que permite a los desarrolladores de software comunicarse de forma precisa y comprensible. El cual tiene beneficios de usar como: Mejorar la comunicación entre los miembros del equipo de desarrollo, Facilita la comprensión del sistema, Permite detectar errores en las primeras etapas del desarrollo y Promueve la reutilización de componentes.

Este cuenta con unos elementos principales los cuales son los: Diagramas de casos de uso, Diagramas de clases, Diagramas de secuencia, Diagramas de actividad, Diagramas de estado.

UML: Es una herramienta poderosa para el diseño y la documentación de sistemas de software. Su uso puede mejorar la calidad del software, facilitar el desarrollo y el mantenimiento, y promover la comunicación entre los miembros del equipo de desarrollo.

TERMINOLOGÍA UML

Diagrama: es una representación gráfica que se utiliza para visualizar y comprender información compleja de forma clara y concisa.

Método: Un método es un procedimiento o conjunto de pasos que se siguen para lograr un objetivo.

Agregación: Relación entre dos clases en la que una clase es parte de la otra.

Polimorfismo: Capacidad de un objeto para responder a un mensaje de diferentes maneras.

Interfaz: es un contrato entre una clase y sus clientes. Define un conjunto de operaciones que la clase debe proporcionar, pero no especifica cómo se implementan esas operaciones.

Registro: Cada una de las entradas que componen una tabla, en el mundo de las bases de datos.

StarUML: Es una herramienta CASE gratuita y de código abierto que permite modelar sistemas de software utilizando UML.

Atributo: Característica de un objeto.

Nodo: es un punto de conexión en una red. Puede ser un dispositivo físico, como un ordenador, un router o un switch, o un punto virtual, como un punto de interconexión en una red virtual.

Interacciones: es la acción que se ejerce recíprocamente entre dos o más objetos, personas, energías o entes.

Estructuras: se refiere a la organización estática de un sistema de software, y se representa utilizando diversos diagramas que muestran las clases, objetos, paquetes, componentes y sus relaciones.

CONCLUSIÓN

UML, es una herramienta valiosa para el desarrollo de software que puede proporcionar muchos beneficios a los proyectos. Sin embargo, es importante ser consciente de sus limitaciones y elegir las herramientas y técnicas adecuadas para el proyecto en cuestión.