

## Actividad | #3 | Modelo UML

### Lenguaje Unificado de Modelado

Ingeniería en Desarrollo de Software

---



TUTOR: Eduardo Israel Castillo García

---

ALUMNO: Pilar Barajas Cervantes

---

FECHA: 30/08/2023

---

## Índice

Introducción.....	3
Descripción .....	4
Justificación .....	5
Diagramas de Actividades .....	6
Diagrama de Componentes .....	7
Conclusión .....	8
Referencias .....	9

## **Introducción**

**En esta Actividad, proyecto final hablaremos un poco de los diagramas de Actividades y Diagramas de Componentes en sí diagramas UML. La especificación UML 2, aplica de manera considerable las funciones y alcance de los diagramas de actividades, más allá de su clasificación anterior como un tipo especial de diagramas de estado. En la actualidad los diagramas de actividades pueden entenderse como diagramas de flujo para el XXI y se utilizan para escribir reglas de negocio, determinamos caso de uso, series complejas de casos de uso y procesos con punto de decisión y flujos alternos operaciones que ocurren en paralelo o flujos de software y estructuras de control lógicas. Por otro lado, los diagramas de componentes UML asignan la vista lógica de las clases del proyecto a los archivos que contienen el código fuente en el que se implementa la lógica. Cuando UModel genera código los diagramas de componentes representan la ubicación de los archivos de código de fuente Java, C++, C# o Visual Basic para sus clases. Al realizar ingeniería inversa en un proyecto ya existente, los diagramas de componentes pueden ayudarle a establecer relaciones entre cada diagrama de clases UModel y los archivos de código fuente. Un diagrama de componentes muestra los componentes, el cableado y los artefactos como parte de un sistema físico. Se suele utilizar para ilustrar como los sistemas complejos, por ejemplo, sistemas de software, servidores Etc. Están conectados entre sí al mundo físico.**

## **Descripción**

**En UML un diagrama de actividades se usa para mostrar la secuencia de actividades. Los diagramas de actividades muestran el flujo de trabajo desde el punto de inicio y el punto final detallando muchas de las rutas de decisiones que existen en el proceso de eventos contenidos en la actividad. Estos también pueden usarse para detallar situaciones donde el proceso paralelo puede ocurrir en la ejecución de algunas actividades. Los diagramas de actividades son útiles para el modelo de negocios donde se usa para detallar el proceso involucrado en a las actividades del negocio.**

**Los diagramas UML se utilizan para modelar los componentes de un sistema y describen sus funcionalidades. En esta actividad también veremos que es un diagrama de componentes, se utilizan para visualizar la organización de los componentes de un sistema y las relaciones de dependencia entre ellos. Proporcionan una visión de alto nivel de los componentes de un sistema, los componentes pueden ser un componente de software como; una base de datos o una interfaz de usuario o un componente de hardware, como un circuito, un microchip, un dispositivo o una unidad de negocio; como, un proveedor, una nómina o un envío. También se utilizan en el desarrollo basado en componentes para describir sistemas con arquitectura orientada a servicios.**

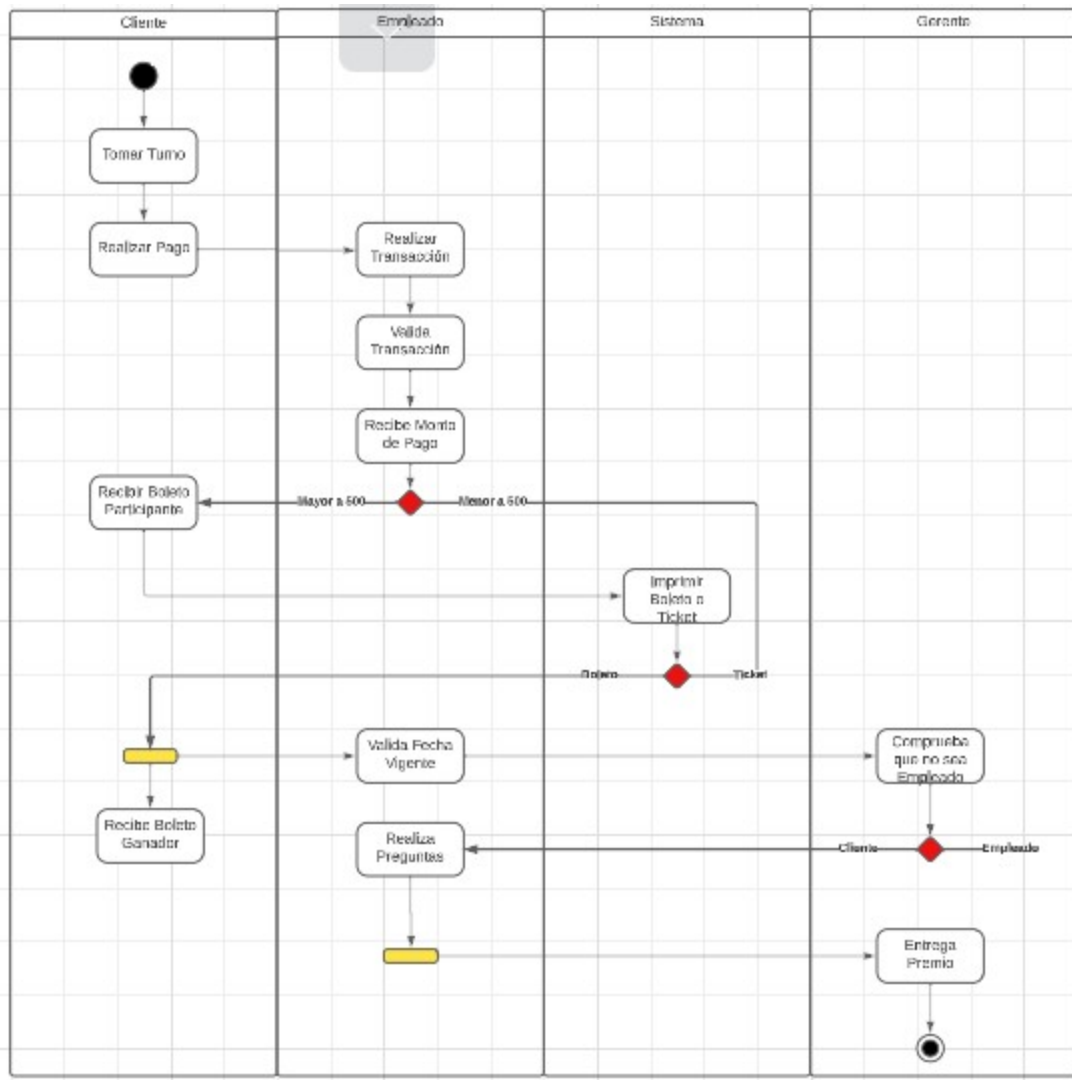
## **Justificación**

**El lenguaje unificado de modelado o UML (por sus siglas en inglés) es una forma de representar visualmente la arquitectura, diseño e implementación de un sistema de software complejos. Cuando se escribe código, hay miles de líneas en una aplicación y resulta difícil llevar un control de las relaciones y las jerarquías dentro de un sistema de software. Los diagramas UML dividen este sistema de software en componente y subcomponentes. ¿Porque deberíamos usar diagramas UML?**

**El UML es un lenguaje de modelado que se puede utilizar en diferentes lenguajes de programación y proceso de desarrollo, por lo que la mayoría de desarrolladores de software lo pueden entender y aplicar en su trabajo.**

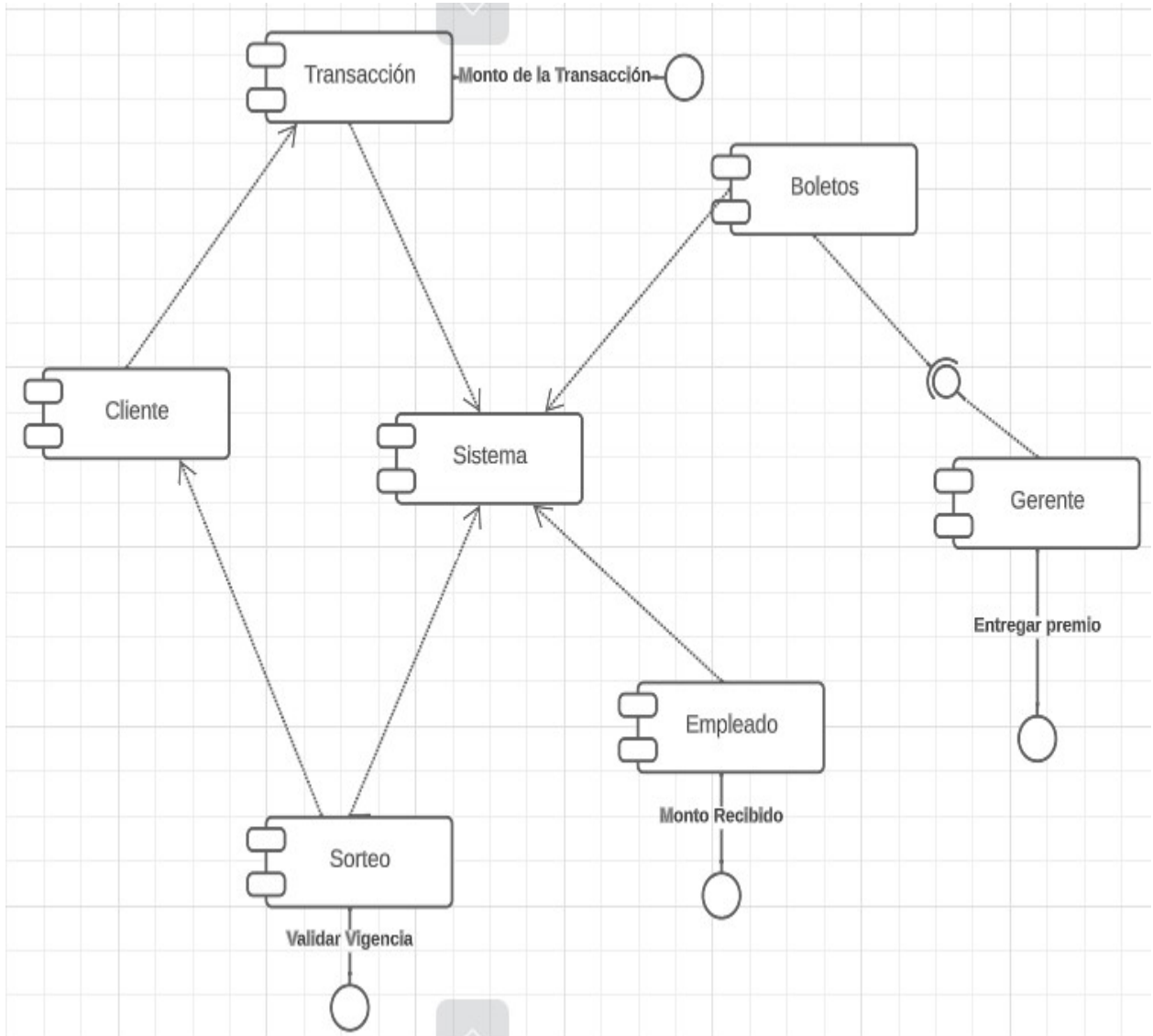
**Muchos ingenieros suelen utilizar los diagramas por que resultan útiles en un entorno de desarrollo ya que permiten mantener el desarrollo productivo y enfocado, en lugar de considerarlos como algo que es bueno tener. Un ejemplo son los diagramas de actividad ya que visualizan los pasos realizados en un caso de uso, las actividades pueden ser secuenciales, ramificadas o concurrentes este tipo de diagrama UML se utiliza para mostrar el comportamiento dinámico de un sistema, pero también puede ser utilizado en el modelo de procesos de negocio. Por otro lado, los diagramas de comportamiento, visualizan la forma en la que se comporta el sistema y como interactúa dentro de sí mismo, así como los usuarios otros sistemas y otras entidades.**

## Diagramas de Actividades



## Diagrama de Componentes

Un diagrama de componentes muestra la asignación de clases y objetos a componentes de implementación, así como también sus dependencias y compilación.



## **Conclusión**

**Existen diferentes formas de representar la funcionalidad del software sin estar terminado, una de las clases es el, lenguaje unificado de modelado UML, que es el sistema de modelado de software más conocido y utilizado en la actualidad; está compuesto por diversos elementos gráficos que se combinan para conformar diagramas, además se encuentra respaldo por el OMG (Objeto Management Group) que se dedica al cuidado y establecimiento de estándares de tecnologías orientada a objetos.**

**Dentro de UML se pueden encontrar diversos diagramas que permiten representar diversas perspectivas de un sistema como son los diagramas de actividades y diagramas de componentes a los cuales se les conoce como modelo, que es una representación simplificada de la realidad. En conclusión, también podemos decir que el diagrama de actividades ayuda al programador, diseñador y analista a entender de forma específica que ocurre dentro de una operación o un proceso del sistema donde se interactúa con un objeto; además tiene también como finalidad determinar las responsabilidades que tienen los procesos de negocios. Los diagramas de componentes nos permiten por medio de su estructura, analizar el software, así como sus relaciones entre componentes. También, así como importar o reutilizar componentes de otro sistema.**



## Referencias

Altova GmbH. (n.d.). XML, data integration, and mobile app development solutions by Altova -

Altova. <https://www.altova.com/>

Sparx Systems. (n.d.). <https://www.sparxsystems.com/>

Creately | Visual Collaboration & Diagramming Platform | CreateLy. (n.d.). <https://creately.com/>

Lucid. (n.d.). Intelligent Diagramming | LucidChart. Lucidchart. <https://www.lucidchart.com/>

Presentations and videos with engaging visuals for hybrid teams | Prezi. (n.d.). prezi.com.

<https://prezi.com/>