



DESARROLLO DE SOFTWARE

Gestión de Requerimientos

QUE SON?

Los requerimientos de software son las **necesidades y expectativas** que deben cumplirse para que un software sea considerado exitoso. Estos requerimientos pueden provenir de diferentes partes interesadas, como **clientes, usuarios, desarrolladores y gerentes**.

Si una compañía desea otorgar un contrato para un gran proyecto de desarrollo de software, tiene que definir sus necesidades de una **forma suficientemente abstracta para que una solución no esté predefinida**. Los requerimientos deben redactarse de tal forma que muchos proveedores liciten en pos del contrato, ofreciendo, tal vez, diferentes maneras de cubrir las necesidades de organización del cliente. Una vez otorgado el contrato, el proveedor tiene que escribir con más detalle una definición del sistema para el cliente, de modo que éste comprenda y valide lo que hará el software. Estos documentos suelen nombrarse documentos de requerimientos para el sistema.



IMPORTANCIA DE LOS REQUERIMIENTOS



Asegurar que el software cumpla con su propósito: Al definir claramente los requerimientos desde el inicio, se evita la ambigüedad y se reduce el riesgo de desarrollar un software que no satisfaga las necesidades de los usuarios.

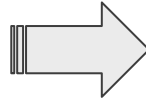
Mejorar la comunicación entre las partes interesadas: Un proceso de gestión de requerimientos bien definido fomenta la comunicación efectiva entre todas las partes involucradas en el proyecto, lo que ayuda a evitar malentendidos y conflictos.

Optimizar el uso de recursos: Al tener una comprensión clara de los requerimientos, se pueden planificar y asignar los recursos de manera más eficiente.

Facilitar el mantenimiento del software: Un buen conjunto de requerimientos documentados facilita la comprensión del código y su posterior mantenimiento.

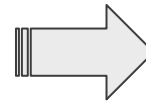
NIVELES DE REQUERIMIENTOS

Requerimientos de Usuario



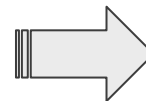
Administradores clientes,
usuarios finales del sistema,
ingenieros clientes,
arquitectos contratistas,
arquitectos del sistema.

Requerimientos de Sistema



usuarios finales del sistema,
ingenieros clientes,
arquitectos del sistema,
**desarrolladores de
software**

Especificación del diseño de
software



ingenieros clientes,
arquitectos del sistema,
**desarrolladores de
software**

REQUERIMIENTOS FUNCIONALES

Los requerimientos funcionales describen **lo que el software debe hacer**. Especifican las funciones, características y comportamientos del software desde la perspectiva del usuario.

- Requerimientos Funcionales de usuario
- Requerimientos Funcionales de sistema
- Requerimientos Funcionales de especificación del diseño del software

El software debe permitir al usuario registrarse en una cuenta.

El software debe mostrar una lista de productos con sus precios y detalles.

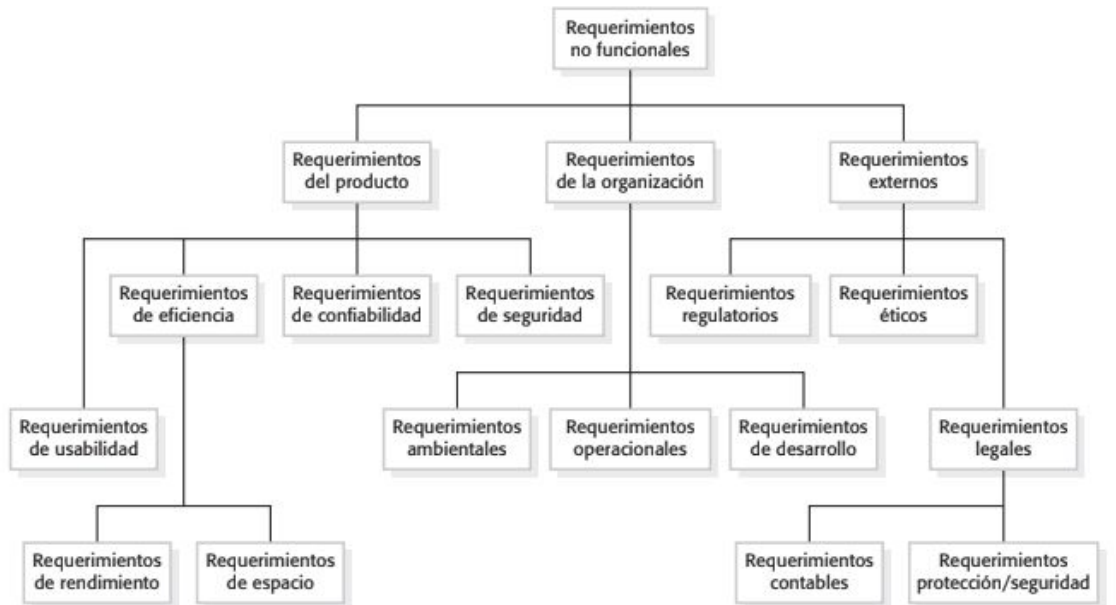
El software debe permitir al usuario agregar productos al carrito de compras.

El software debe procesar pagos con tarjeta de crédito.

REQUERIMIENTOS NO FUNCIONALES

Los requerimientos no funcionales describen cómo debe funcionar el software. Especifican las características y atributos del software que no están directamente relacionados con sus funciones

- Requerimientos NO Funcionales de usuario
- Requerimientos NO Funcionales de sistema
- Requerimientos NO Funcionales de especificación del diseño del software



ELEMENTOS PARA GESTIÓN REQUERIMIENTOS

Introducción: Describe el propósito del software y el alcance del proyecto.

Descripción de los requerimientos: Detalla los requerimientos funcionales y no funcionales del software.

Casos de uso: Describen cómo los usuarios interactúan con el software para realizar tareas específicas.

Priorización de requerimientos: Establece la importancia relativa de cada requerimiento.

Glosario de términos: Define los términos técnicos utilizados en la documentación.

ESPECIFICACIÓN DE REQUERIMIENTOS

Notación	Descripción
Enunciados en lenguaje natural	Los requerimientos se escriben al usar enunciados numerados en lenguaje natural. Cada enunciado debe expresar un requerimiento.
Lenguaje natural estructurado	Los requerimientos se escriben en lenguaje natural en una forma o plantilla estándar. Cada campo ofrece información de un aspecto del requerimiento.
Lenguajes de descripción de diseño	Este enfoque usa un lenguaje como un lenguaje de programación, pero con características más abstractas para especificar los requerimientos al definir un modelo operacional del sistema. Aunque en la actualidad este enfoque se usa raras veces, aún tiene utilidad para especificaciones de interfaz.
Anotaciones gráficas	Los modelos gráficos, complementados con anotaciones de texto, sirven para definir los requerimientos funcionales del sistema; los casos de uso del UML y los diagramas de secuencia se emplean de forma común.
Especificaciones matemáticas	Dichas anotaciones se basan en conceptos matemáticos como máquinas o conjuntos de estado finito. Aunque tales especificaciones sin ambigüedades pueden reducir la imprecisión en un documento de requerimientos, la mayoría de los clientes no comprenden una especificación formal. No pueden comprobar que representa lo que quieren y por ello tienen reticencia para aceptarlo como un contrato de sistema.

ESPECIFICACIÓN DE REQUERIMIENTOS

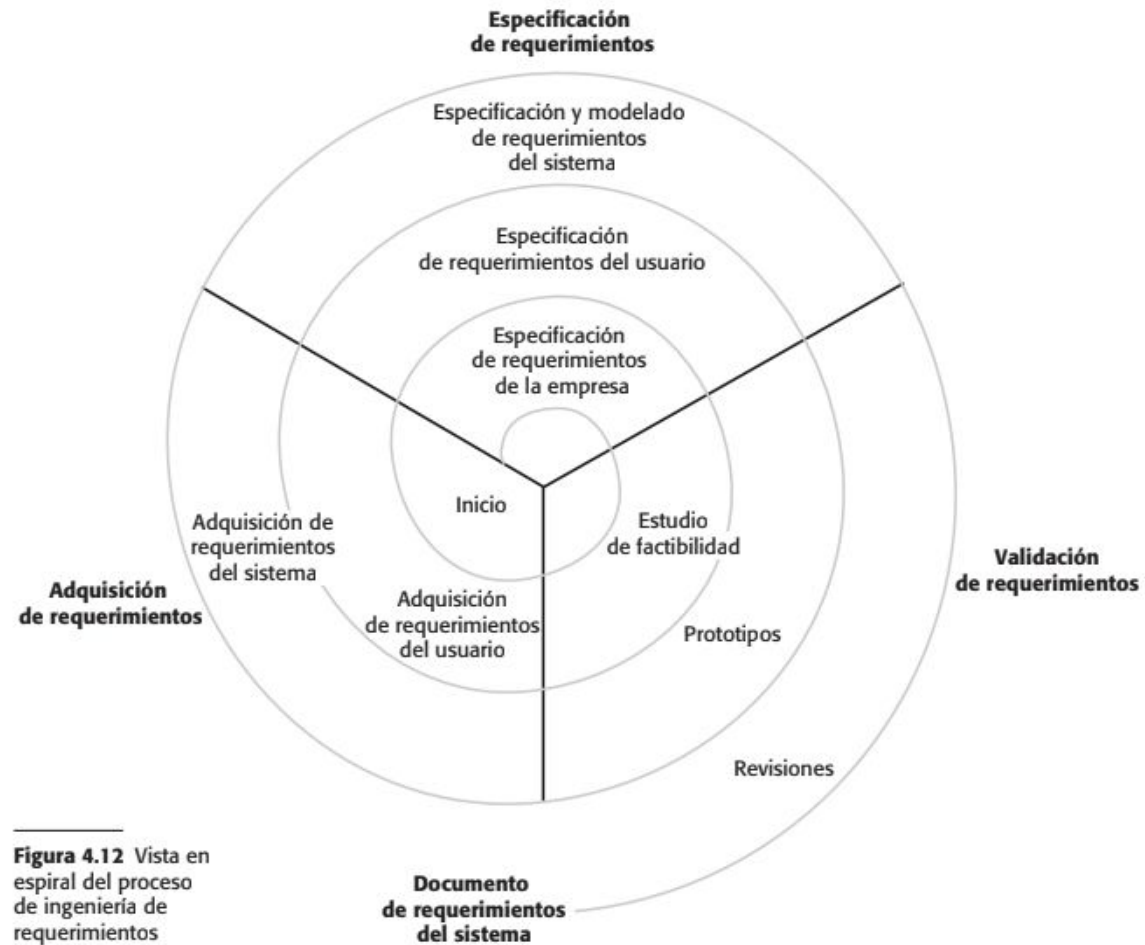


Figura 4.12 Vista en espiral del proceso de ingeniería de requerimientos

UML - Unified Modify Language

Lenguaje visual que se utiliza para diseñar y documentar sistemas de software. Proporciona una forma estándar de representar los diferentes aspectos de un sistema, incluida su estructura, comportamiento e interacciones.

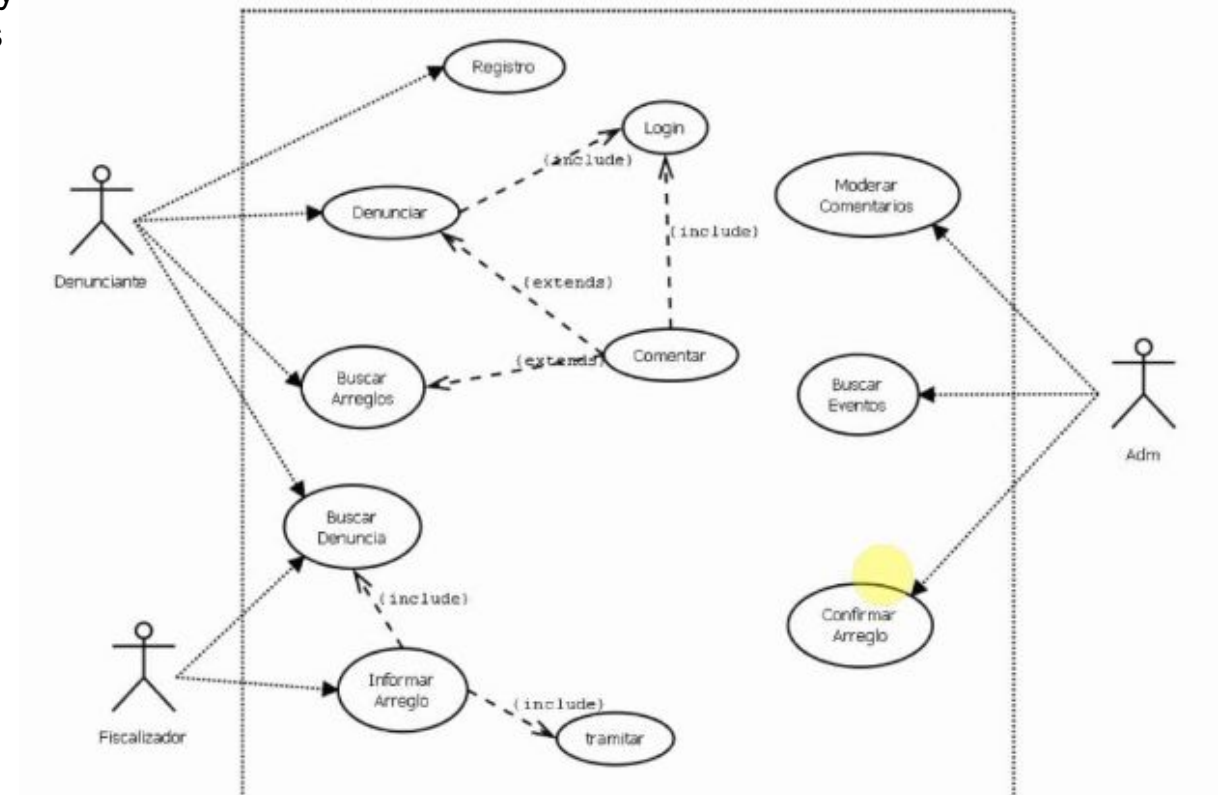
Diagramas: UML utiliza una colección de diferentes diagramas, cada uno con un propósito específico. Estos diagramas pueden mostrar elementos como clases, objetos, interacciones y procesos.

Es importante recordar que UML no es un lenguaje de programación en sí. Es más como un plano que ayuda a los desarrolladores a comprender y diseñar el software antes de comenzar a codificar.

UML puede ser una herramienta valiosa para el desarrollo de software porque mejora la comunicación entre desarrolladores, partes interesadas y usuarios. También ayuda a garantizar que el software esté diseñado de una manera organizada, eficiente y fácil de mantener.

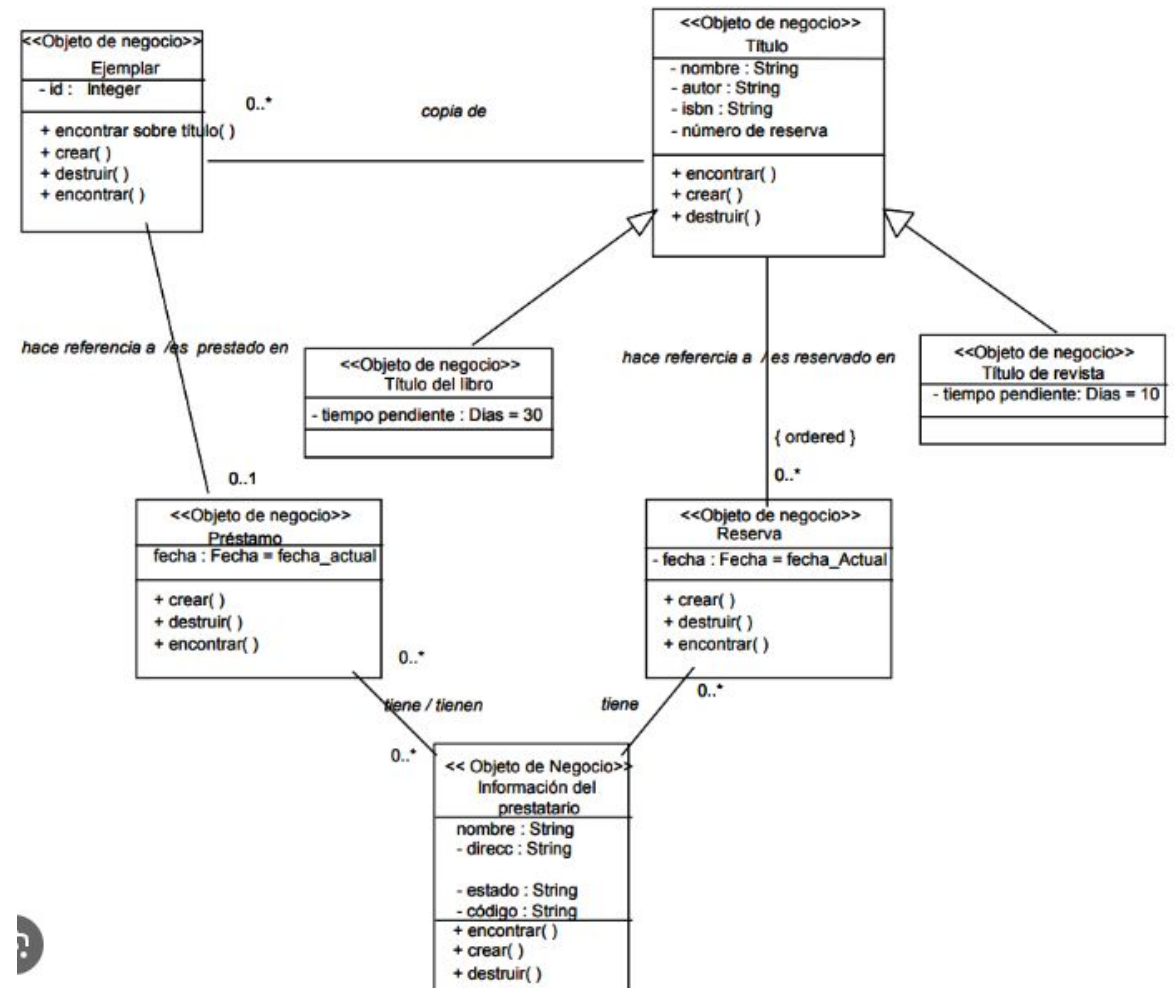
UML - DIAGRAMA DE CASOS DE USOS

- Estos diagramas muestran cómo los usuarios interactúan con un sistema. Los casos de uso se representan como óvalos y los actores se representan como figuras de palo. Las líneas entre los casos de uso y los actores muestran cómo interactúan.



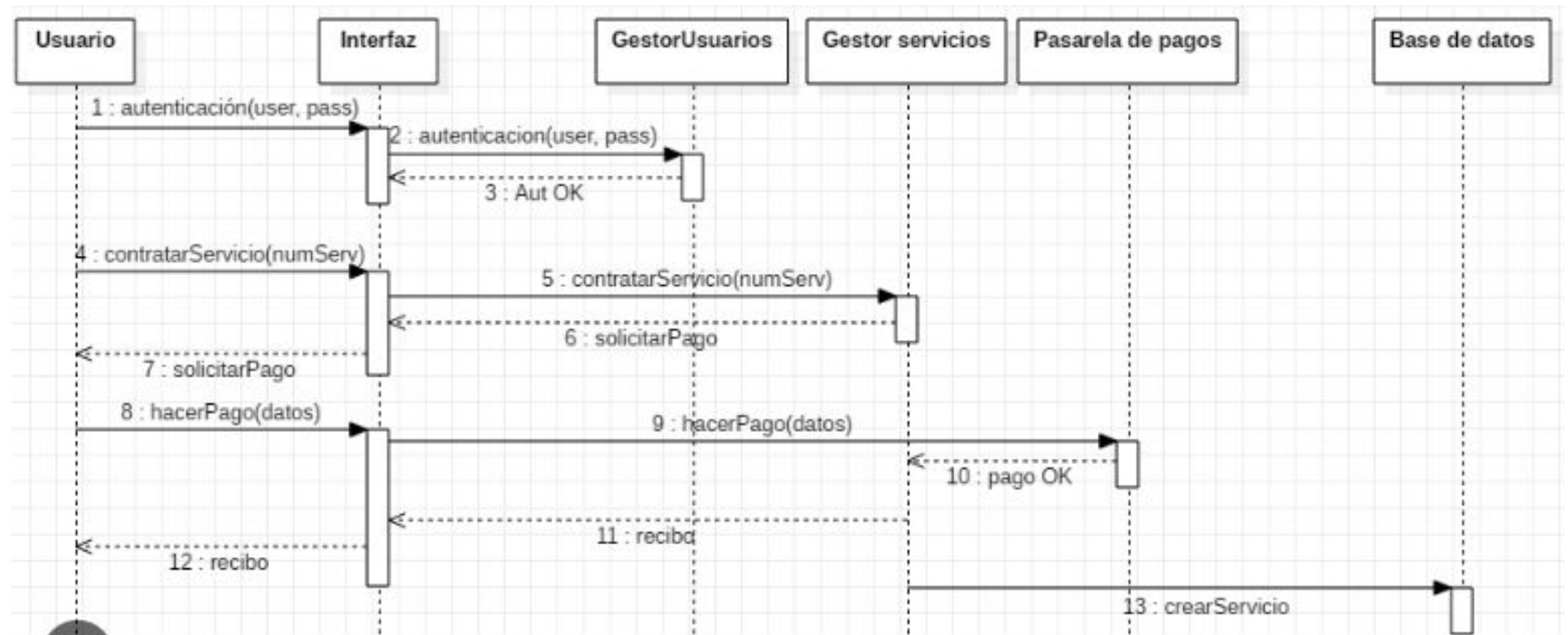
UML - DIAGRAMA DE CLASES

- Estos diagramas muestran las clases que componen un sistema y las relaciones entre ellas. Las clases se representan como rectángulos y las relaciones se representan como líneas entre los rectángulos.



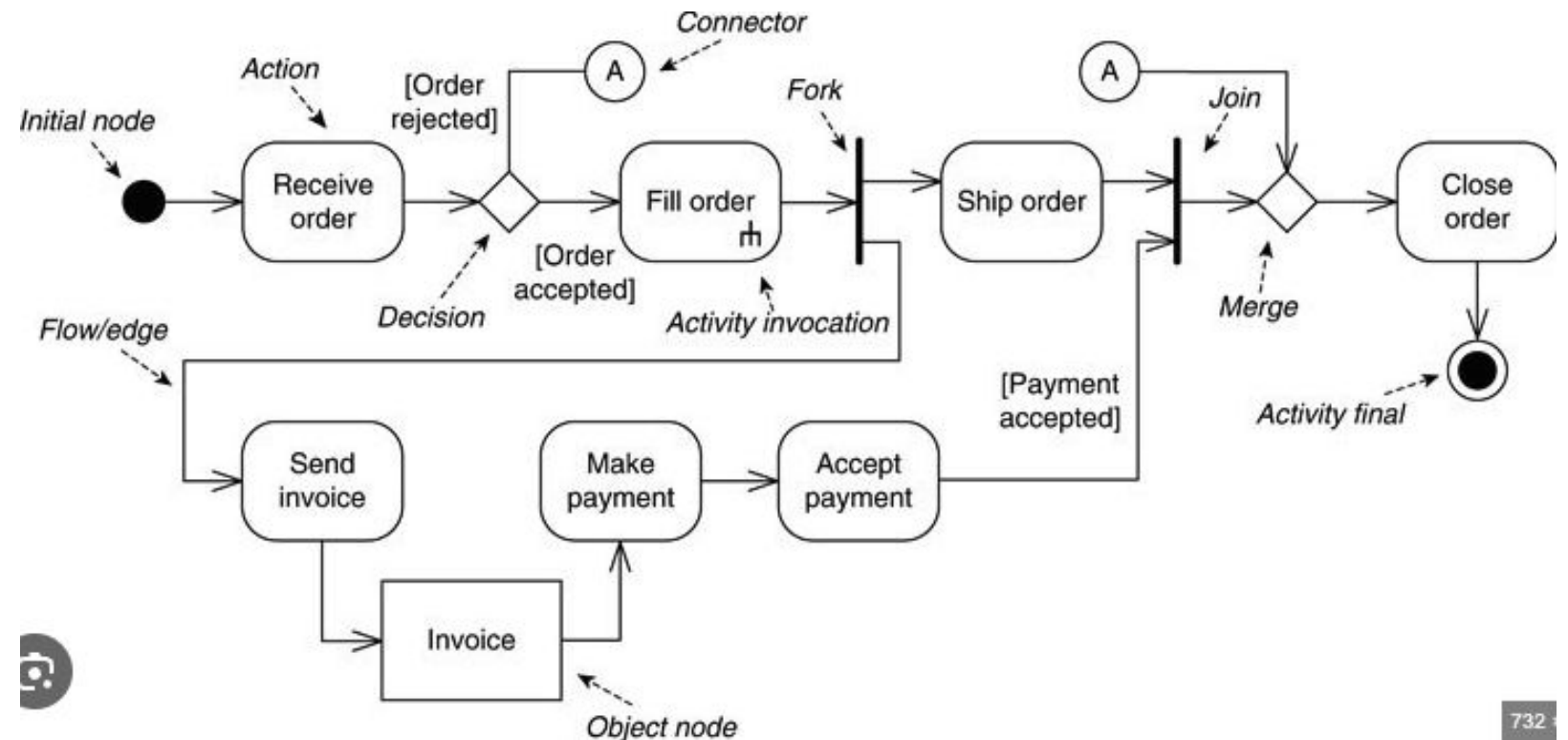
UML - DIAGRAMA DE SECUENCIAS

muestran la secuencia de interacciones entre los objetos en un sistema. Los objetos se representan como líneas de vida verticales y las interacciones se representan como líneas horizontales entre las líneas de vida.



UML - DIAGRAMA DE ACTIVIDADES

Estos diagramas muestran los pasos involucrados en un proceso. Los pasos se representan como rectángulos redondeados y las transiciones entre los pasos se representan como flechas.



TRELLO - PROYECTOS

A TRABAJAR!