

# Analisis y Diseño

- El análisis se centra en la representación del problema, es decir, en la identificación de las abstracciones que representen el significado de las especificaciones y de los requisitos del sistema. El énfasis del diseño se centra en la definición de la solución.

## Analisis Y Diseño Orientado a Objetos

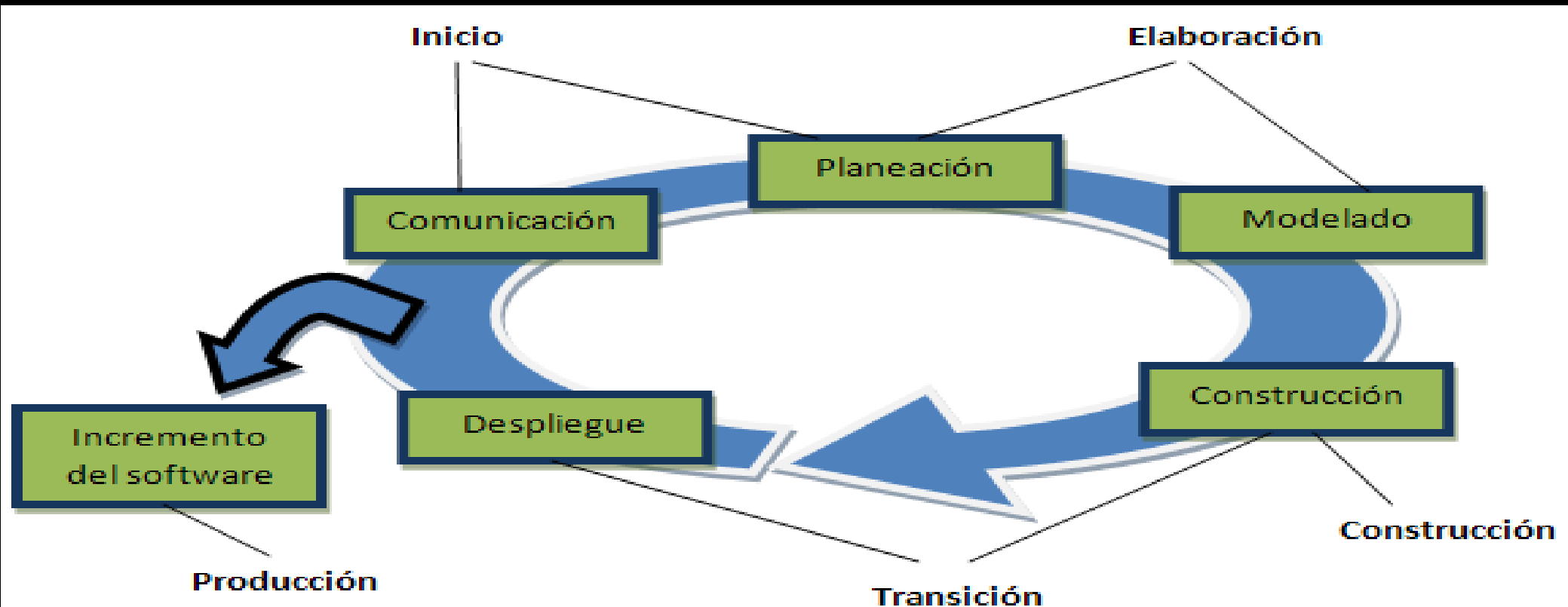
Es un enfoque de análisis que modela un sistema como un grupo de objetos que interactúan entre sí.

## Modelado de software

El modelado de software es un proceso esencial en la ingeniería de software que implica la creación de representaciones abstractas de un sistema de software para comprender, diseñar, construir y mantener aplicaciones complejas.

# El Proceso Unificado (UP)

Es una metodología y se compone de varias fases, incluyendo la concepción, el desarrollo, la transición y la producción. Cada fase está compuesta por iteraciones que producen entregables concretos y verificables. Además, el UP permite dividir el desarrollo en ciclos iterativos, donde cada ciclo produce una versión del sistema que va evolucionando y mejorando con el tiempo.



# SCRUM

En Scrum, el trabajo se organiza en iteraciones cortas y regulares llamadas "sprints", que típicamente duran de una a cuatro semanas. Durante cada sprint, un equipo multifuncional trabaja en la implementación de una lista de elementos de trabajo priorizados, llamada "backlog del sprint". Esta lista se crea a partir del "product backlog", que es una lista priorizada de todas las características, mejoras y correcciones necesarias para el producto.



# Por que esta compuesto el Scrum?

- **Product Owner:** Es responsable de representar las necesidades de los clientes y las partes interesadas, definir los elementos del backlog del producto y priorizarlos en función del valor que aportan al producto.
- **Scrum Master:** Actúa como facilitador y protector del equipo Scrum. Se encarga de eliminar obstáculos, facilitar la comunicación y asegurar que el equipo siga los principios y prácticas de Scrum.
- **Equipo de Desarrollo:** Es un equipo autoorganizado y multifuncional que trabaja en la entrega del incremento del producto durante cada sprint. Son responsables de decidir cómo llevar a cabo el trabajo y de cumplir con los compromisos del sprint.

## ***En el Sprint***

**Reunión de planificación del sprint:** En esta reunión, el equipo selecciona los elementos del backlog del sprint que se compromete a completar durante el sprint.

**Daily Scrum:** Una breve reunión diaria en la que el equipo comparte el progreso, identifica cualquier obstáculo y ajusta el plan para alcanzar el objetivo del sprint.

**Revisión del sprint:** Al finalizar el sprint, el equipo muestra el trabajo completado y obtiene retroalimentación de los stakeholders. Se revisa el backlog del producto y se actualiza según sea necesario.

**Retrospectiva del sprint:** Una reunión para reflexionar sobre el sprint, identificar áreas de mejora y planificar ajustes para el siguiente sprint.

# Requerimientos Funcionales y no Funcionales

## Requisitos Funcionales:

Son las cosas que el sistema debe hacer. Son las funciones o acciones que el sistema necesita llevar a cabo para funcionar correctamente. Por ejemplo, si estás construyendo un sitio web de comercio electrónico, un requisito funcional sería que los usuarios puedan agregar productos a su carrito de compras.

## Requisitos No Funcionales:

Son como las reglas que el sistema debe seguir. No describen lo que el sistema hace, sino cómo lo hace y qué tan bien lo hace. Por ejemplo, si estás construyendo una aplicación móvil, un requisito no funcional podría ser que la aplicación se cargue rápidamente, incluso en conexiones lentas.

ID. Requisito	Nombre del Requisito	Descripción del Requisito	Usuario	Medio	Proceso Asociado
RF-001	Registrar datos de la institución.	El sistema debe permitir el registro la información de la institución educativa, tales como identificación del director, datos de dirección, etc	Personal del departamento de evaluación	Pantalla	Institución
RF-002	Actualización de datos de la institución.	El sistema permite actualizar los datos de la institución que ya se ha registrado con anterioridad. (Principalmente la información del director).	Personal del departamento de evaluación	Pantalla	Institución
RF-003	Registro de nuevo años escolares	El sistema debe permitir que se registre un nuevo año escolar según el formato que se maneja por el departamento de evaluación de la institución.	Personal del departamento de evaluación	Pantalla	Años escolares
RF-004	Finalizar año escolar	El sistema debe permitir que un año escolar incorporado sea determinado como finalizado.	Personal del departamento de evaluación	Pantalla	Años escolares

ID. Requisito	Descripción del Requisito
RNF-001	La interfaz del sistema deberá ser implementada como una aplicación web.
RNF-002	Cada usuario que desee ingresar al sistema, deberá introducir en la página principal un código de usuario y una contraseña, la cual será validada por el sistema, dándole acceso al sistema o enviándole un mensaje para que introduzca nuevamente sus datos.
RNF-003	El sistema deberá tener una interfaz gráfica sencilla y amigable, basada en menús, ventanas, listas desplegables y botones de acción.
RNF-004	El sistema deberá ser desarrollado bajo software libre, utilizando el lenguaje de programación PHP y utilizará el estándar HTML para el diseño de las páginas web del sistema. De esta forma se garantizaría que el código HTML generado pueda ser interpretado por cualquier de los navegadores comerciales existentes en el mercado.
RNF-005	El sistema debe ser diseñado según la arquitectura cliente/servidor de tres capas.
RNF-006	El sistema debe utilizar los servicios de la red interna de la institución, para establecer comunicación entre las coordinaciones y departamentos de directivos, el servidor y el manejador de base de datos.
RNF-007	La organización, manipulación, consulta y almacenamiento de los datos estará bajo la responsabilidad del sistema manejador de base de datos relacional de Mysql.

# Especificación de requisitos de software o srs

- es una descripción completa del comportamiento del sistema que se va a desarrollar. Incluye un conjunto de casos de uso que describe todas las interacciones que tendrán los usuarios con el software.

## Tipos de requisitos

**Requisitos de Usuarios:** Necesidades que los usuarios expresan verbalmente

**Requisitos del Sistema:** Son los componentes que el sistema debe tener para realizar determinadas tareas

**Requisitos Funcionales:** Servicios que el sistema debe proporcionar al finalizar el sistema

**Requisitos No Funcionales:** Requisitos que debe cumplir el sistema no relacionados con funcionamiento o actuaciones de usuario, sino con atributos como rendimiento, seguridad, portabilidad, y otros.

# UML

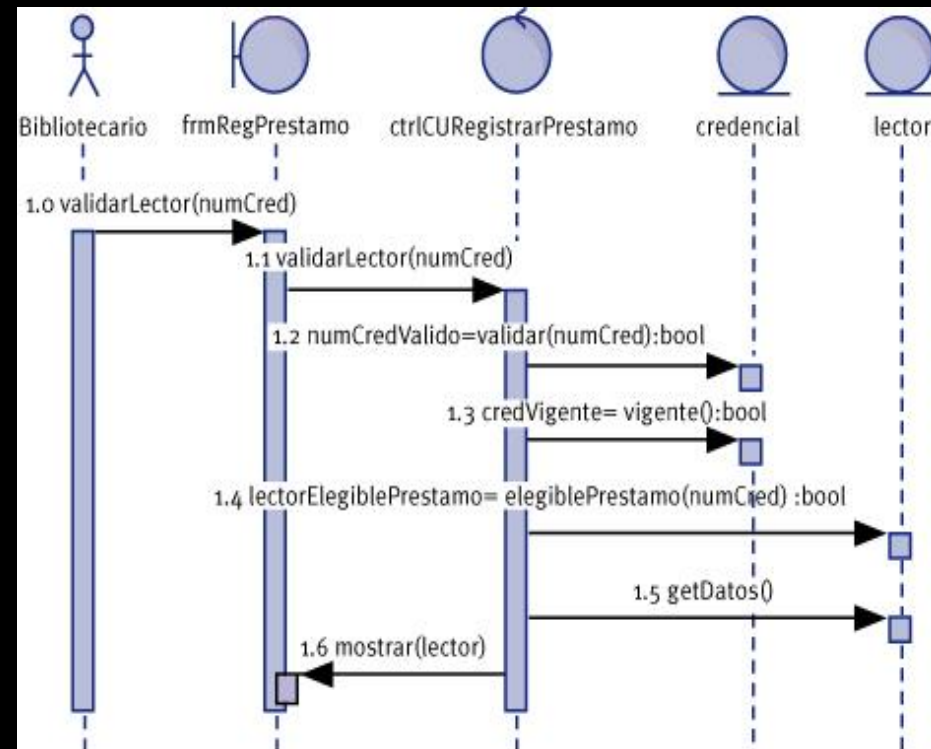
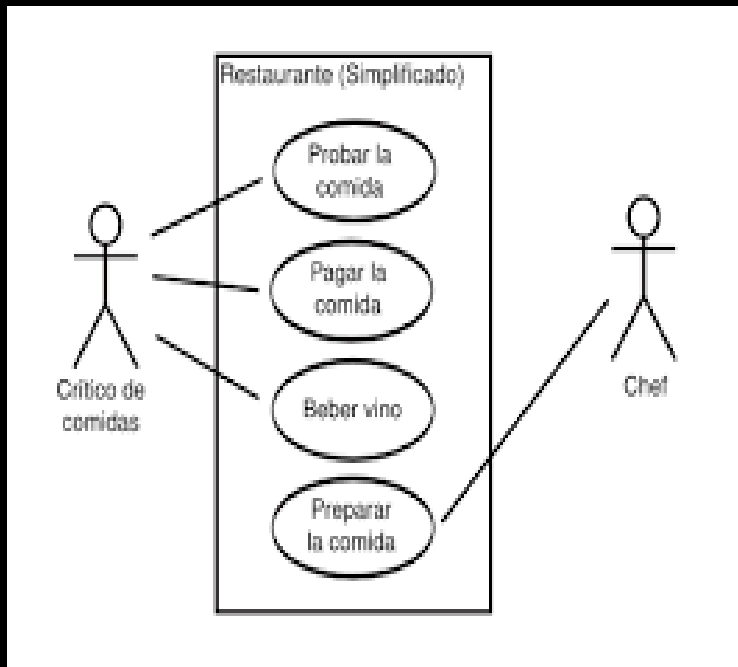
- (UML o lenguaje de modelado unificado) es un estándar que proporciona un lenguaje visual para la especificación, construcción y documentación de los componentes de software y otros sistemas. Fue creado para ayudar a los desarrolladores de software a visualizar y diseñar los sistemas de manera más efectiva y clara.

## Qué se puede hacer con UML

### Modelar el Comportamiento del Sistema:

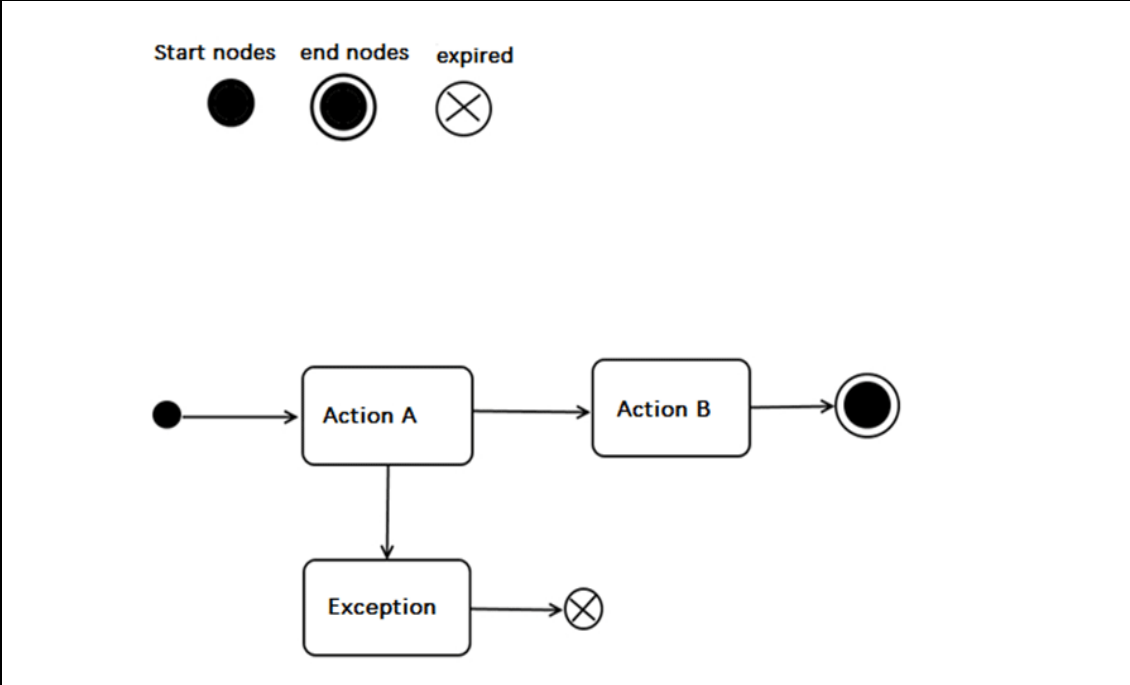
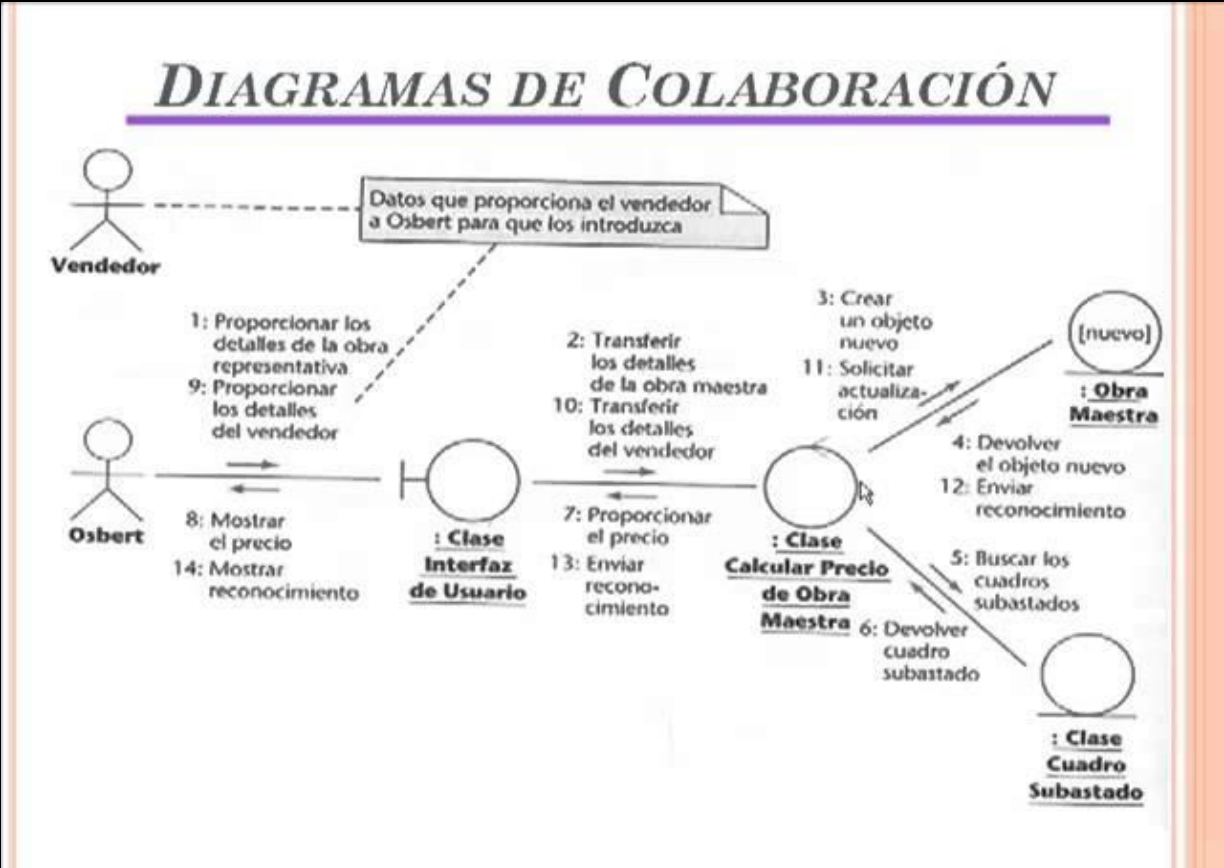
Diagramas de casos de uso: Representan las funcionalidades del sistema desde la perspectiva del usuario final. Muestran los actores (usuarios u otros sistemas) y los casos de uso (funciones o servicios que el sistema proporciona).

Diagramas de secuencia: Describen cómo los objetos interactúan en un proceso específico, mostrando el orden de los mensajes intercambiados entre los objetos.



**Diagramas de colaboración:** Muestran las interacciones entre objetos y su organización en un sistema.

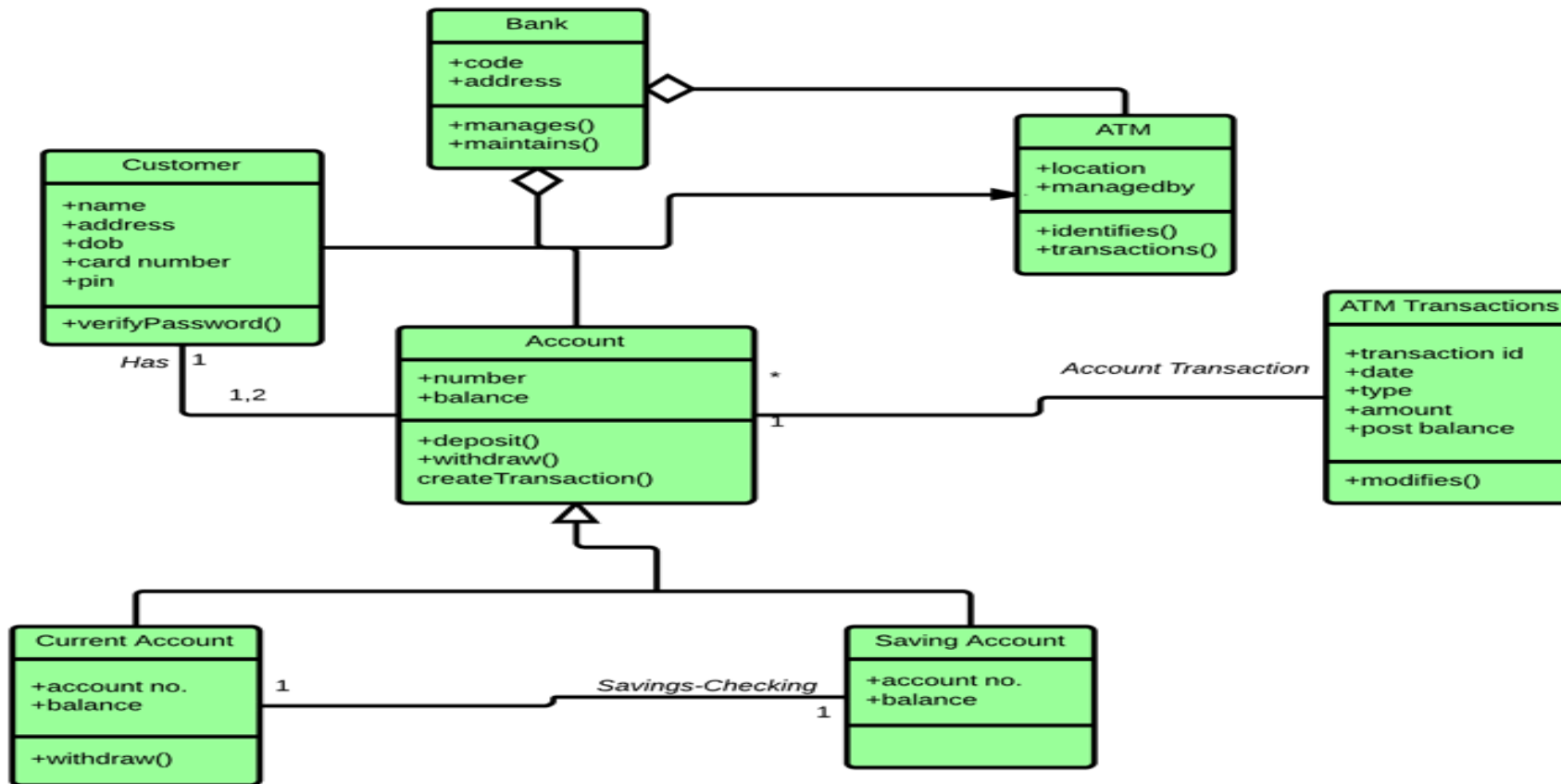
**Diagramas de actividad:** Representan el flujo de actividades y decisiones dentro de un proceso o sistema.





# UML

- **Modelar la Estructura del Sistema:**
- **Diagramas de clases:** Describen la estructura del sistema mostrando sus clases, atributos, métodos y las relaciones entre las clases (herencia, asociación, composición, etc.).

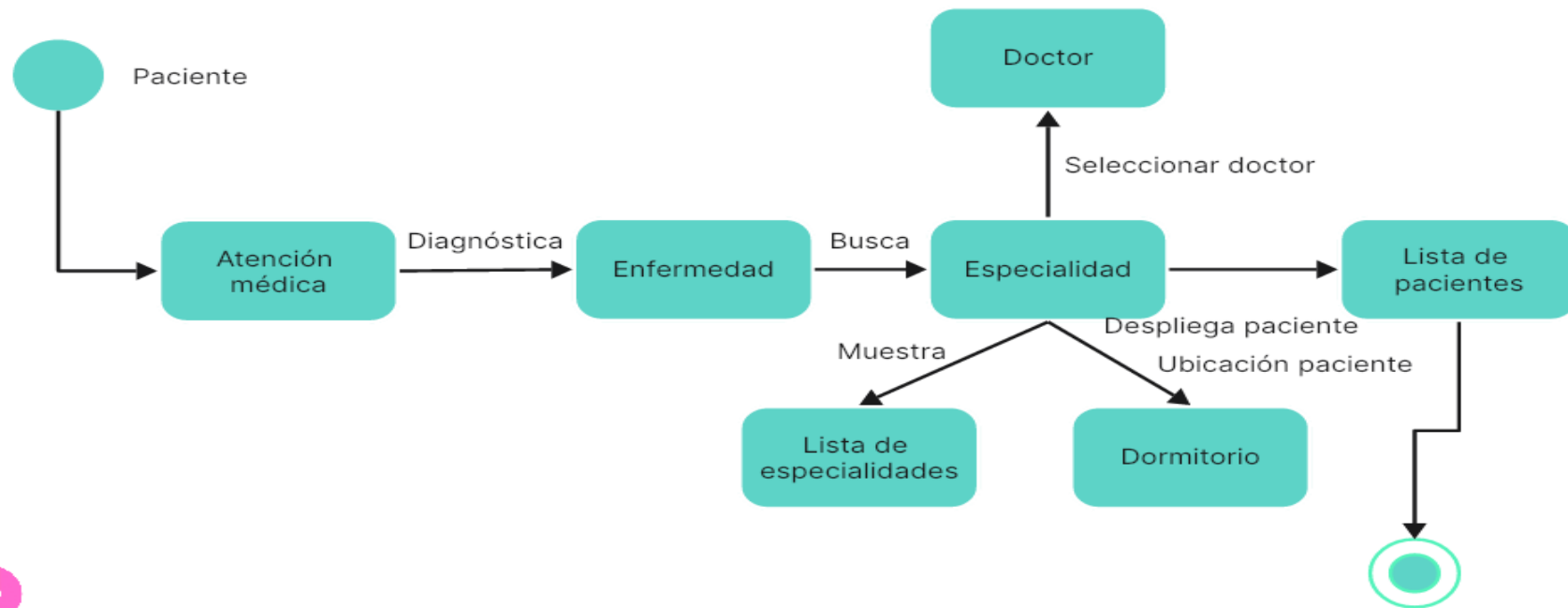


# UML

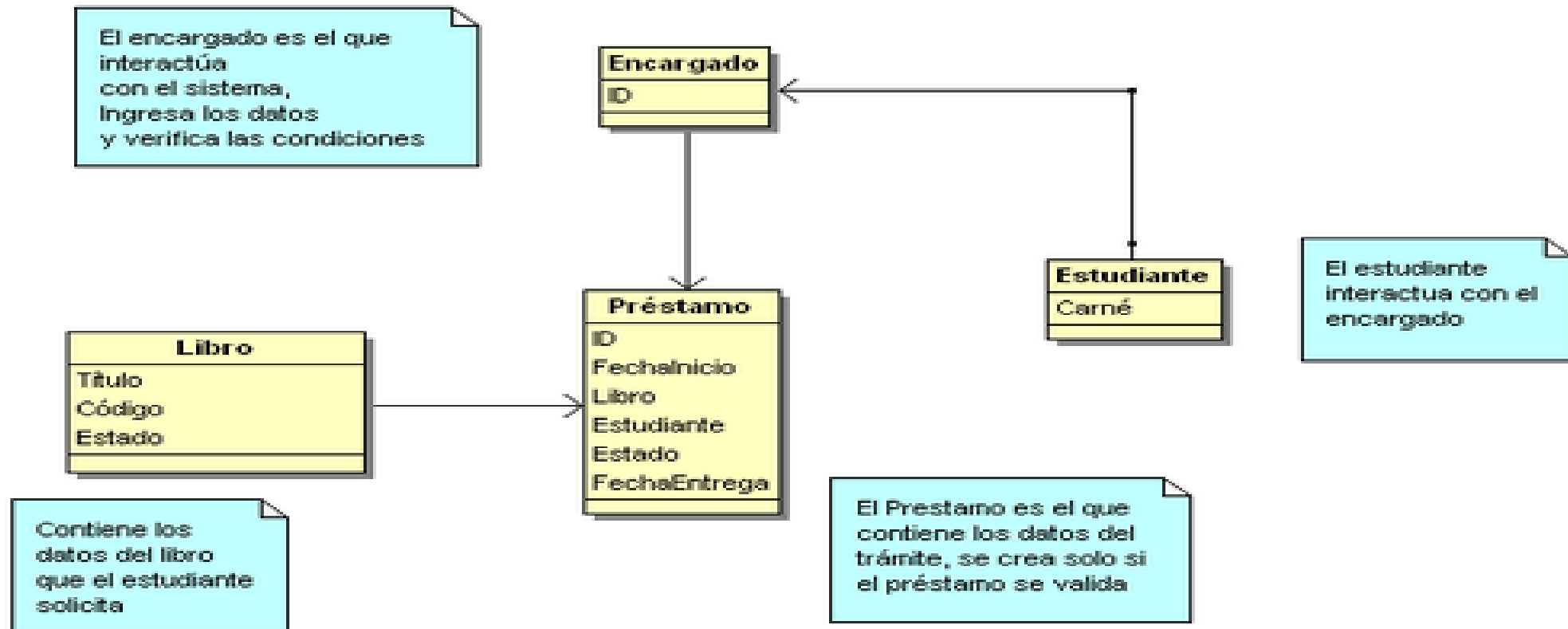
- Modelar el Estado del Sistema:

**Diagramas de estados:** Describen los diferentes estados que puede tener un objeto y las transiciones entre esos estados debido a eventos. A continuación un ejemplo de un sistema de control de un hospital

Diagrama de estados de sistema de control de hospital



**Modelo de dominio:** es una representación conceptual de los objetos, conceptos y relaciones dentro del dominio de un problema específico. Se utiliza para entender y comunicar cómo se organiza la información y cómo interactúan los diferentes elementos dentro de ese dominio. Los modelos de dominio son esenciales en el desarrollo de software, especialmente durante las fases de análisis y diseño, ya que ayudan a garantizar que el sistema construido



# Modelo Triple 8-30

- El estándar IEEE 830, formalmente conocido como IEEE Std 830-1998, proporciona directrices para crear un documento bien estructurado de Especificación de Requisitos de Software (ERS).

## Introducción

**Propósito:** Describe el propósito de la ERS y su audiencia objetivo.

**Alcance:** Define el alcance del software, incluyendo beneficios, objetivos y metas.

**Referencias:** Lista todos los documentos que se mencionan en la ERS.

**Resumen:** Proporciona una visión general del contenido y la estructura de la ERS.

## Descripción General

**Perspectiva del Producto:** Describe el contexto y el origen del producto, incluyendo interfaces, hardware, software y interfaces de comunicación.

**Funciones del Producto:** Lista las principales funciones que realizará el software.

**Características de los Usuarios:** Describe las características generales de los usuarios previstos del producto, incluyendo nivel educativo, experiencia y conocimientos técnicos.

**Restricciones:** Lista las restricciones bajo las cuales debe operar el producto, tales como políticas regulatorias, limitaciones de hardware y cualquier otra restricción de diseño.

**Suposiciones y Dependencias:** Especifica cualquier suposición que afecte los requisitos indicados en la ERS.

# Modelo Triple 8-30

- **Requisitos Específicos**
- **Requisitos Funcionales:** Descripción detallada de las funciones del sistema, como entradas, salidas y comportamiento en respuesta a las entradas.
- **Requisitos de Interfaz Externa:** Especificaciones de las interfaces externas, tales como interfaces de usuario, hardware, software y de comunicación.
- **Requisitos de Rendimiento:** Requisitos relacionados con las características de rendimiento del sistema, como velocidad, disponibilidad, tiempo de respuesta y capacidad.
- **Requisitos Lógicos de Base de Datos:** Descripción de los requisitos lógicos para cualquier información que se colocará en una base de datos.
- **Restricciones de Diseño:** Restricciones que afectan el diseño del sistema, como cumplimiento de normas y limitaciones de hardware.
- **Atributos del Sistema de Software:** Atributos del sistema, incluyendo fiabilidad, disponibilidad, seguridad, mantenibilidad y portabilidad.