



El futuro digital
es de todos

MinTIC

Ciclo: Programación Básica
Diseño Orientado a Objetos

»
Misión TIC 2022

xxx



Misión
TIC 2022



El futuro digital
es de todos

MinTIC



Programación Básica

x
x
x

- **Modelamiento Orientado a Objetos**
- **Lenguaje de Modelado Unificado (UML)**
- **Diagramas de Clases**
- **Diagramas de Casos de Uso**





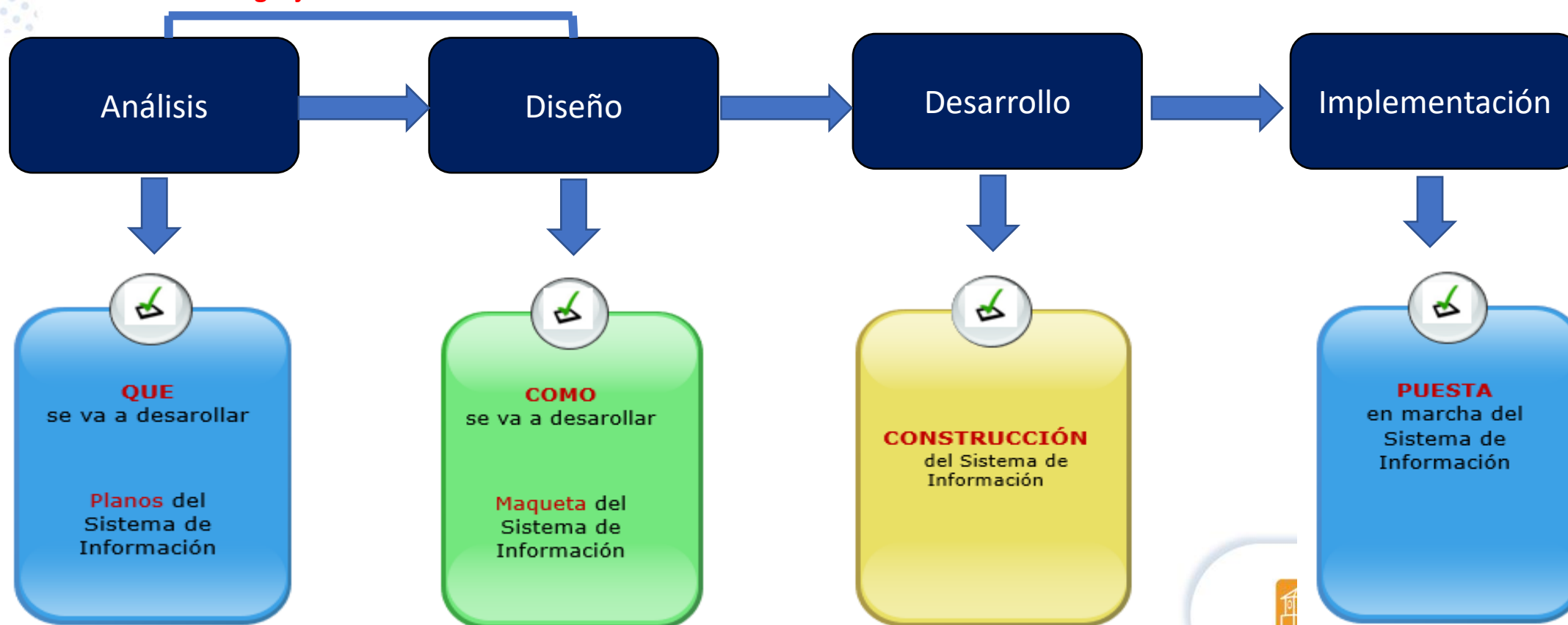
Tema: Ciclo de Vida del Software



Ciclo de Vida de un sistema de información

x
x
x

UML=>Lenguaje de Modelado Unificado





Tema: Ciclo de Vida del Software



Ciclo de Vida de un sistema de información

x
x
x

UML=>Lenguaje de Modelado Unificado

Análisis



QUE
se va a desarrollar

Planes del
Sistema de
Información

- **Reconocimiento del Problema:** Contacto con el usuario y la situación problema.
- **Síntesis:** Recopilación de información y procesos
- **Definición (Análisis) de Requerimientos:** Los **requerimientos** especifican **qué** es lo **que** el sistema debe hacer (sus funciones) y sus propiedades esenciales y deseables. Se realiza a través de los diagramas que ofrece el Lenguaje de Modelado Unificado (Clases, casos de uso, secuencias, estados, actividades, componentes, entre otros.)
- **Socialización y Aprobación** por parte del usuario



El futuro digital
es de todos

MinTIC



Tema: Ciclo de Vida del Software

Definición de Requerimientos



| Código del Requerimiento | RF01 |
|--------------------------|---|
| Nombre | Registro de Usuario |
| Propósito | Crear un nuevo usuario capaz de presentar encuestas en el aplicativo web PETIC. |
| Descripción | Una vez ubicados en la página de inicio de sesión, el potencial usuario debe dirigirse al hipervínculo que indica la creación de una nueva cuenta, diligenciar el formulario con sus respectivos campos obligatorios y proceder a finalizar el proceso pulsando en el botón indicado. |
| Entrada | Formulario de Registro diligenciado con información de la empresa y usuario. |
| Salida | <ul style="list-style-type: none">- Mensaje de bienvenida al correo electrónico registrado.- Redirección a la página de la primer Encuesta. |
| Prioridad | Alta |

Requerimiento funcional

| Código del Requerimiento | RNF04 |
|--------------------------|---|
| Nombre | Confidencialidad |
| Descripción | Toda la información otorgada por los usuarios se manipulará únicamente con fines corporativos y de manera limpia. |
| Prioridad | Alta |

Requerimiento NO funcional



Tema: Ciclo de Vida del Software

Diagramas de Casos de uso

En un **sistema de información para el manejo de un Aula Virtual de Economía Solidaria**, se especifica que el estudiante inicialmente debe iniciar sesión, lo que incluye la validación de su usuario y contraseña y el registro en el Log de auditoría. Una vez haya ingresado al sistema podrá realizar varias actividades como: Revisar Agenda de Actividades, Revisar Contenidos, Participar en los foros, Revisar y contestar tareas, Revisar y contestar evaluación, Revisar lista de recursos, que incluye visualizar videos, revisar libros electrónicos multimediales y revisar documentos de apoyo. Además, puede Verificar porcentaje de progreso del curso y Consultar notas del curso.

Adicionalmente, se especifica que el docente, inicialmente debe iniciar sesión, lo que incluye la validación de su usuario y contraseña y el registro en el Log de auditoría. Una vez el docente ingrese al sistema podrá Administrar contenidos del curso, revisar en foros, Administrar tareas, Enviar mensajes, Administrar recursos, que incluye la gestión de videos, incluir libros electrónicos multimediales e incluir documentos de apoyo. Además, puede Publicar notas del estudiante y Administrar evaluación de los cursos, que incluye la generación de evaluaciones y la calificación de las evaluaciones.

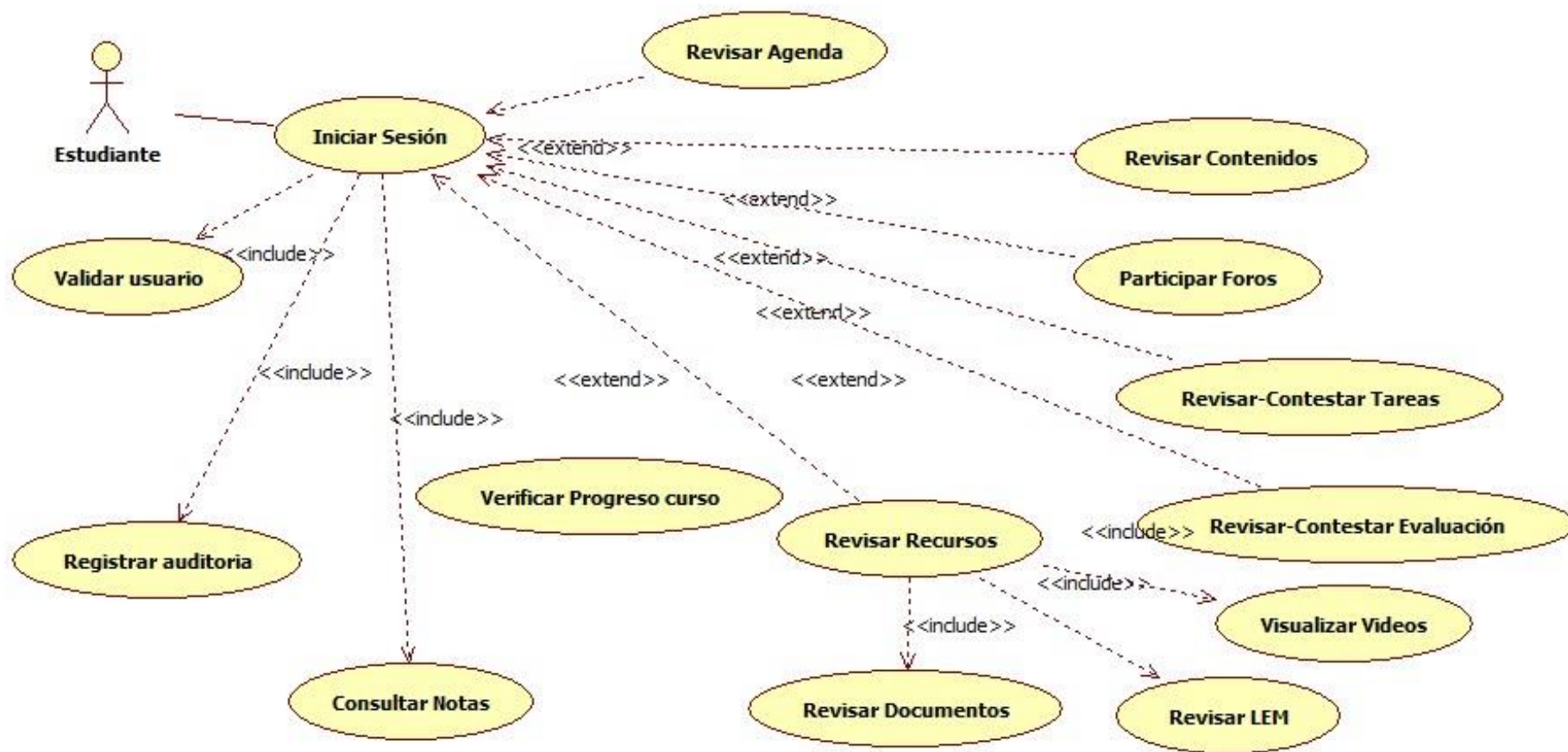
Finalmente, se especifica al administrador del sistema, quien inicialmente debe iniciar sesión, lo que incluye la validación de su usuario y contraseña y el registro en el Log de auditoría. Posteriormente, puede gestionar la base de datos, lo que incluye las operaciones de incluir, modificar y eliminar. Además, realizar los procesos de gestionar la seguridad (Usuario, contraseñas) y gestionar el proceso de copias de respaldo. También es el encargado que generar los informes estadísticos sobre los resultados de los estudiantes.



Tema: Ciclo de Vida del Software



Diagrama de Casos de Uso Estudiante





Tema: Ciclo de Vida del Software



Ciclo de Vida de un sistema de información



UML=>Lenguaje de Modelado Unificado

Diseño



COMO
se va a desarrollar

Maqueta del
Sistema de
Información

- **Modelado Arquitectura de los Datos:** Modelado de la Base de Datos.
- **Modelamiento de los procesos:** Identificación y diseño de los módulos y procesos del sistema de información.
- **Seguridad y Respaldo:** Esquema de seguridad del sistema de información, procesos de auditoría y copias de seguridad.
- **Interfaz – Tecnologías aplicadas.** Diseño de la interfaz o capa de presentación (Front-End)

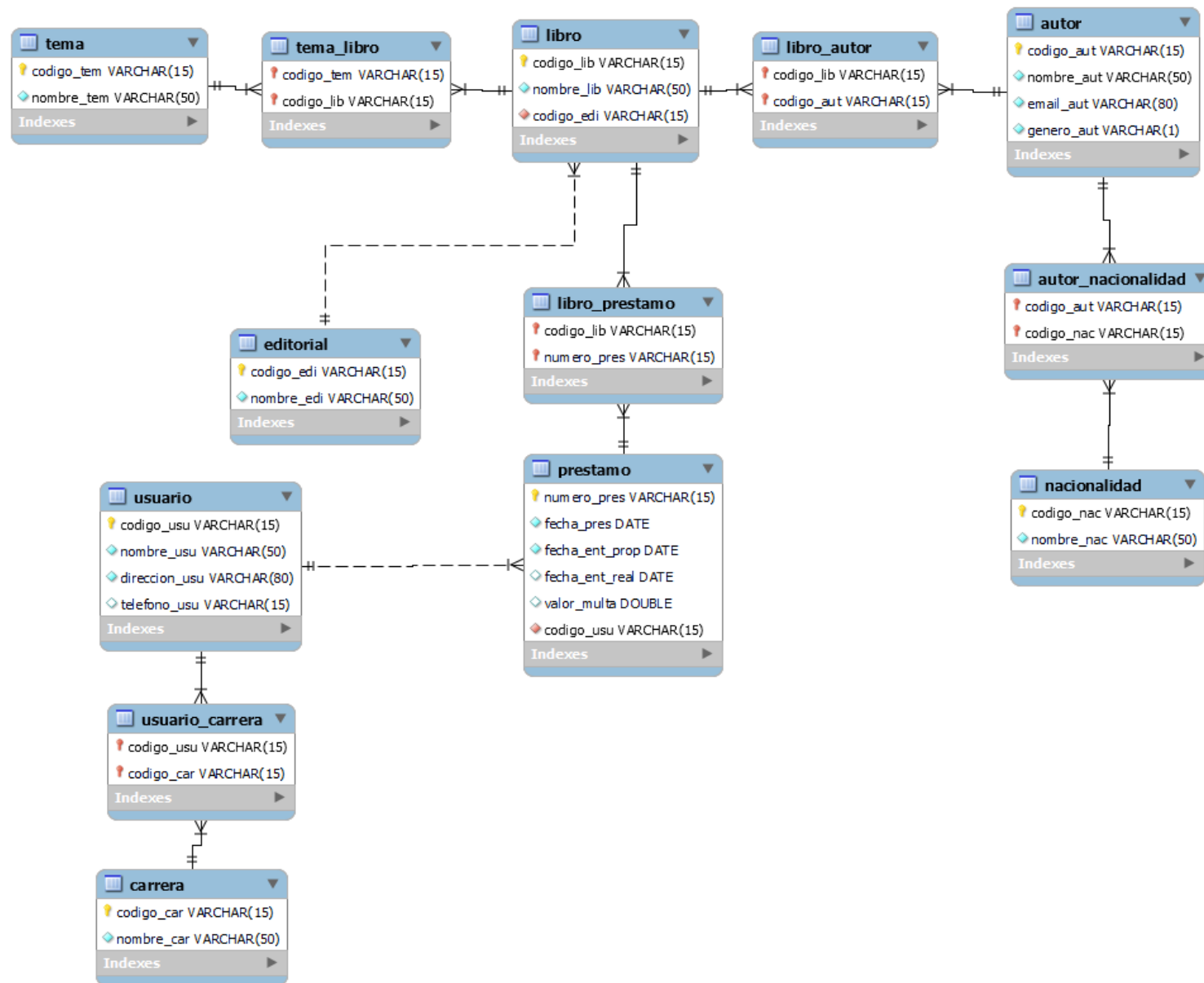


El futuro digital
es de todos

MinTIC

Tema: Ciclo de Vida del Software

Modelo Conceptual Base de Datos



inab
EDUCACIÓN

Mision
TIC2022



Tema: Ciclo de Vida del Software

Diagramas de Clase: Ejemplo 1

Representa mediante un diagrama de clases la siguiente especificación.

Una aplicación necesita almacenar información sobre empresas, sus empleados y sus clientes. Las empresas tienen un identificador y nombre. Ambos (empleados y sus clientes) se caracterizan por su identificación, nombre, edad. Los empleados tienen un salario. El empleado tiene una experiencia, de la cual se conoce la descripción y tiempo y el cliente posee un historial de ventas, de las cuales se conoce una descripción y valor vendido. De los clientes, además se necesita conocer su teléfono de contacto y el correo electrónico. La aplicación necesita mostrar los datos de empresas, empleados, experiencia, clientes e historial de ventas.

Además, sobre las empresas, empleados y clientes se deben realizar las operaciones de incluir, modificar y eliminar. A los empleados se les debe realizar un proceso periódico de incremento de salario y a los clientes se les aplica un proceso de sanción, cuando no cumplen con las metas estipuladas o cometen alguna falta.

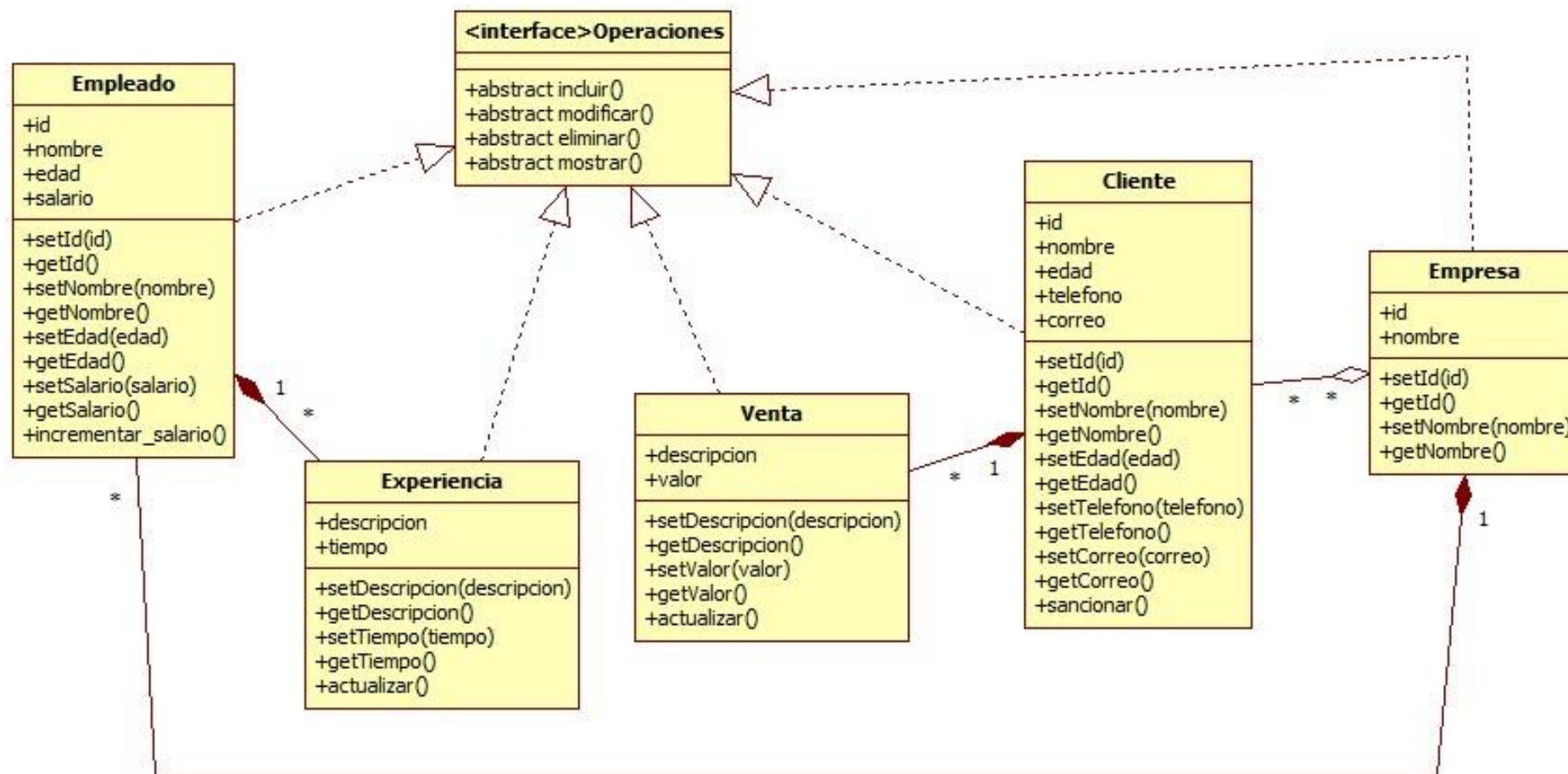
Es importante el proceso de actualización de la experiencia para los empleados y el historial de ventas para los clientes.



Tema: Ciclo de Vida del Software



Diagrama de Clases





Tema: Ciclo de Vida del Software



Ciclo de Vida de un sistema de información



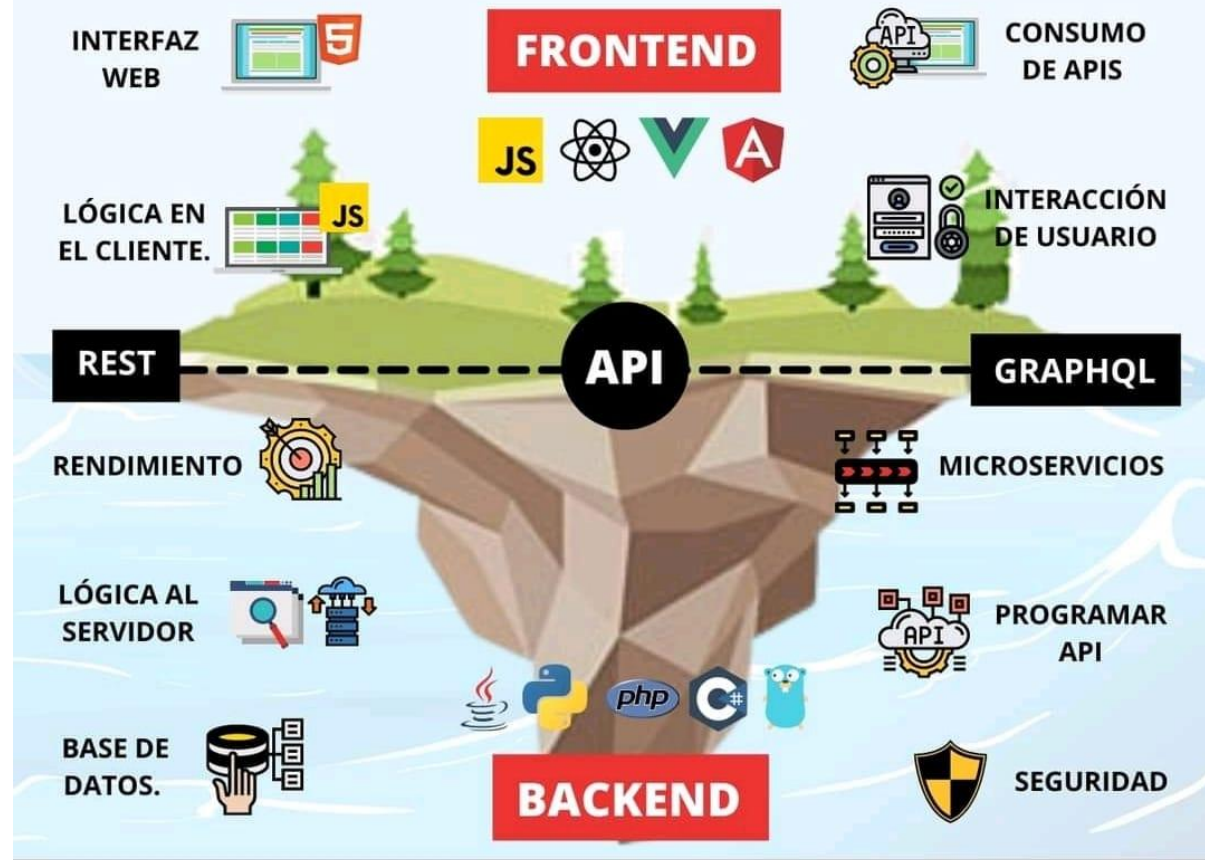
UML=>Lenguaje de Modelado Unificado

Desarrollo



- **Esquema de la Base de Datos:** Construcción de la base de Datos con un DBMS (Sistema administrador de Bases de Datos).
- **Programación:** Programación de los procesos incluyendo la seguridad y respaldo.

¿QUÉ ES EL BACKEND Y FRONTEND?



¡APRENDE LARAVEL ESTE 2021!

<https://codersfree.com>



Coders Free



Misión
TIC 2022



Tema: Ciclo de Vida del Software



Ciclo de Vida de un sistema de información



UML=>Lenguaje de Modelado Unificado

Implementación



PUESTA
en marcha del
Sistema de
Información

- **Pruebas (Pilotaje):** Prueba del sistema, a través del diseño de casos de prueba de caja blanca y caja negra. (Se definen muestras)
- **Ajustes:** Mejoras o correcciones a las inconsistencias encontradas en las pruebas.
- **Puesta en Marcha:** Puesta en funcionamiento definitiva del sistema de información.



Tipos de pruebas funcionales para el aseguramiento de la calidad



¿Qué son?

Son un proceso de control de calidad para asegurar el cumplimiento de un sistema o componente con requerimientos funcionales.

Estas pruebas pueden realizarse durante:

- la fase de desarrollo: individualmente para secciones específicas desarrolladas por el equipo
- al final: cuando las diferentes secciones del proyecto están unidas.



Unitarias

Asegura que cada célula del código desarrollado en un componente brinde los resultados adecuados.

Se evalúan:

- interfaz
- la especificación de un componente



de Componentes

Verifica las funcionalidades y/o usabilidades de los componentes cumplan con lo planeado, aunque no solo se limite a ello. Por ejemplo:

- Prueba de UI para usabilidad y accesibilidad
- Prueba de carga para asegurar el rendimiento
- Prueba de login con credenciales válidas e inválida



de Humo

Verifica si las funcionalidades más significativas de la aplicación funcionan o no, de forma que lo más básico del software se ejecute de forma correcta con pruebas sencillas y rápidas.

Encuentra más información en: <https://trycore.co/transformacion-digital/tipos-de-pruebas-funcionales/>



Modelamiento Orientado a Objetos



Todo gira en torno de una visión. Un **sistema complejo** toma forma cuando alguien tiene la visión de **cómo la tecnología puede mejorar las cosas**. Los desarrolladores tienen que entender completamente la idea y mantenerla en mente mientras crean el sistema que le dé forma.

El éxito de los proyectos de desarrollo de aplicaciones o sistemas se debe a que sirven como enlace entre quien tiene la idea y el desarrollador. **El UML** (*Lenguaje Unificado de Modelado*) es una herramienta que cumple con esta función, ya que le **ayuda a capturar la idea de un sistema para comunicarla posteriormente a quien esté involucrado en su proceso de desarrollo**; esto se lleva a cabo mediante un **conjunto de símbolos y diagramas**. Cada diagrama tiene fines distintos dentro del proceso de desarrollo.



Modelamiento Orientado a Objetos



En los principios de la computación, los programadores no realizaban análisis muy profundos sobre el problema por resolver. Si acaso, garabateaban algo en una servilleta. Con frecuencia comenzaban a escribir el programa desde el principio, y el código necesario se escribía conforme se requería. Aunque anteriormente esto agregaba un aura de aventura y atrevimiento al proceso, en la actualidad es inapropiado en los negocios de alto riesgo.

Hoy en día, es necesario contar con un plan bien analizado. Un cliente tiene que comprender qué es lo que hará un equipo de desarrolladores; además tiene que ser capaz de señalar cambios si no se han captado claramente sus necesidades (o si cambia de opinión durante el proceso). A su vez, el desarrollo es un esfuerzo orientado a equipos, por lo que cada uno de sus miembros tiene que saber qué lugar toma su trabajo en la solución final (así como saber cuál es la solución en general).



Tema: Modelamiento orientado a objetos



Modelamiento Orientado a Objetos

La clave está en organizar el proceso de diseño de tal forma que los analistas, clientes, desarrolladores y otras personas involucradas en el desarrollo del sistema lo comprendan y convengan con él. El UML proporciona tal organización.



Tema: Modelamiento orientado a objetos



Principios del Enfoque Orientado a Objetos

Enfoque orientado a objetos

Principios

Abstracción

Encapsulamiento

Modularidad

Herencia

OTRAS CARACTERISTICAS

Polimorfismo
Enlace Dinámico
Persistencia
Concurrencia



Tema: Modelamiento orientado a objetos



Principios del Modelado Orientado a Objetos

x
x
x

Modelar el dominio de la información

Describir la función del módulo

Representar el comportamiento del módulo

Dividir el modelo para más detalles
(Modelos de Datos, funcional y comportamiento)

Modelos iniciales son la esencia y los finales
aportan detalles para implementación

Principios
Modelo
Orientado a
Objetos



Tema: Lenguaje de Modelado Unificado (UML)



Definición.



El lenguaje de modelado unificado (UML) es un estándar para la **representación visual de objetos, estados y procesos dentro de un sistema**. Por un lado, el lenguaje de modelado puede servir de modelo para un proyecto y garantizar así una arquitectura de información estructurada; por el otro, ayuda a los desarrolladores a **presentar la descripción del sistema de una manera que sea comprensible para quienes están fuera del campo**. UML se utiliza principalmente en el desarrollo de software orientado a objetos. Al ampliar el estándar en la versión 2.0, también es adecuado para visualizar procesos empresariales.

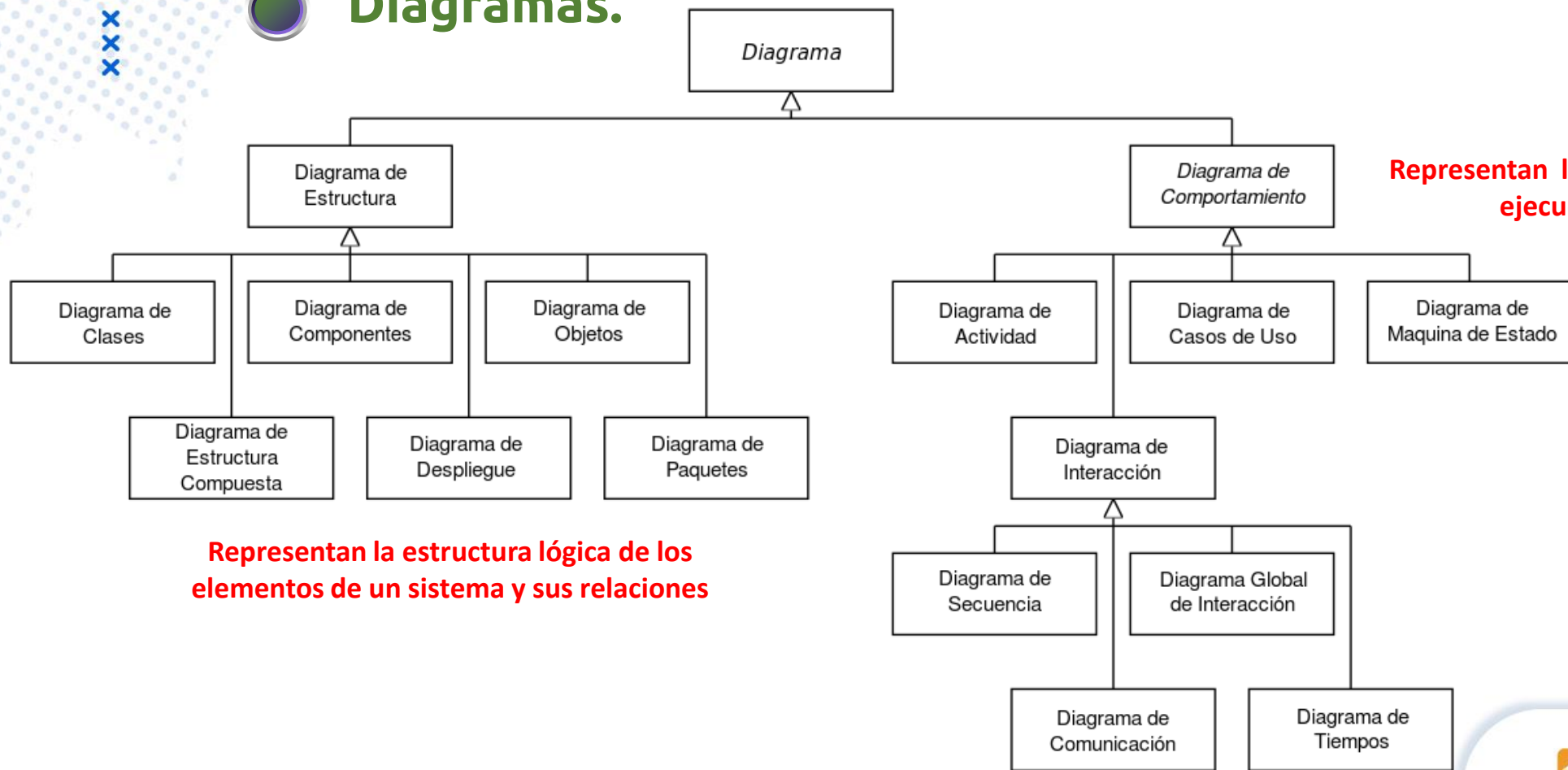
Desde finales de la década de 1980 hasta la de 1990 se desarrolló un gran número de métodos y lenguajes para la representación de la programación orientada a objetos. El resultado fue una confusa abundancia de métodos que apenas eran comparables entre sí. Para unificarlos, los tres desarrolladores **James Rumbaugh, Grady Booch e Ivar Jacobson** decidieron fusionar varios lenguajes existentes en un estándar común.



Tema: Lenguaje de Modelado Unificado (UML)



Diagramas.



Representan la conducta en tiempo de ejecución del sistema

Representan la estructura lógica de los elementos de un sistema y sus relaciones



Tema: Diagramas de Clases



Definición.



El **diagrama de clases** es uno de los diagramas incluidos en UML, clasificado dentro de los diagramas de estructura y, como tal, se utiliza para representar los elementos que componen un sistema de información desde un punto de vista estático.

El diagrama de clases es un **diagrama puramente orientado al modelo de programación orientado a objetos**, ya que define las clases que se utilizarán cuando se pase a la fase de construcción y la manera en que se relacionan las mismas. Se podría equiparar, salvando las distancias, al famoso diagrama de modelo Entidad-Relación (E/R), no recogido en UML, tiene una utilidad similar: la representación de datos y su interacción. Ambos diagramas muestran el modelo lógico de los datos de un sistema.



Tema: Diagramas de Clases



Metodología para la elaboración del diagrama de Clases.



Paso 1: Del enunciado se deben extraer los sustantivos, que serían las clases, los actores o personajes de la situación problema; se listan en singular con la primera letra en mayúscula.

Paso 2: De las clases identificadas, se deben excluir aquellas que no aportan nada para el modelamiento de la situación problema, que se llamarán las clases candidatas.

Paso 3: Se deben identificar los atributos de las clases, analizando las características valores o la información que suministran de cada una de las clases.

Paso 4: Se deben identificar las operaciones o métodos de las clases, a través de las acciones que se menciona que puede realizar cada una de las clases.

Paso 5: Se deben identificar las posibles relaciones entre las clases, señalando el tipo de relación (Asociación, Herencia – Generalización, Dependencia, Composición, Realización).



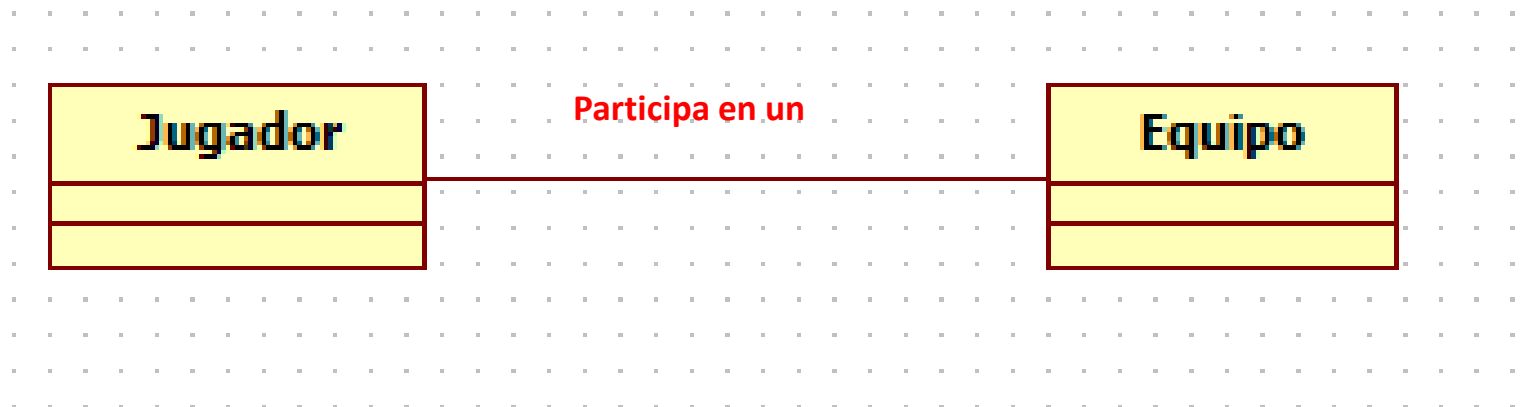
Tema: Diagramas de Clases



Tipos de Relaciones entre clases (Envío de mensajes). **Asociación**



Conexión conceptual de dos clases independientes para participar en un proceso.





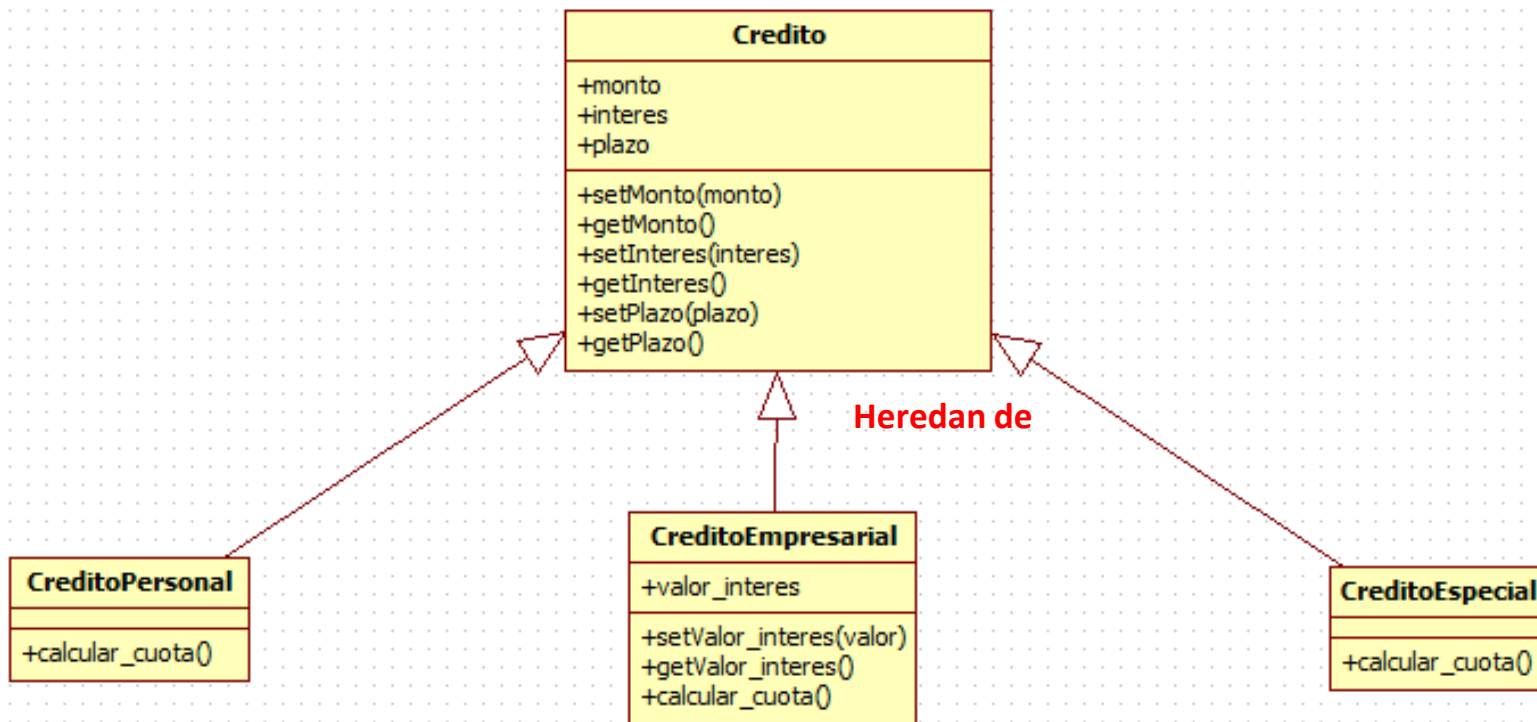
Tema: Diagramas de Clases



Tipos de Relaciones entre clases (Envío de mensajes).

Herencia - Generalización

Conexión entre clases a través de un procedimiento de Herencia, donde la Subclase toma o hereda los atributos y métodos de una Superclase.





Tema: Diagramas de Clases

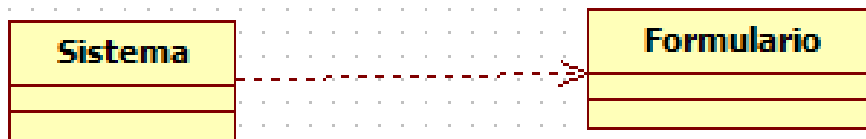
Heredan de



Tipos de Relaciones entre clases (Envío de mensajes). Dependencia

x
x
x

Es una relación de uso, es decir una **clase** usa a otra, que la necesita para su un fin particular.





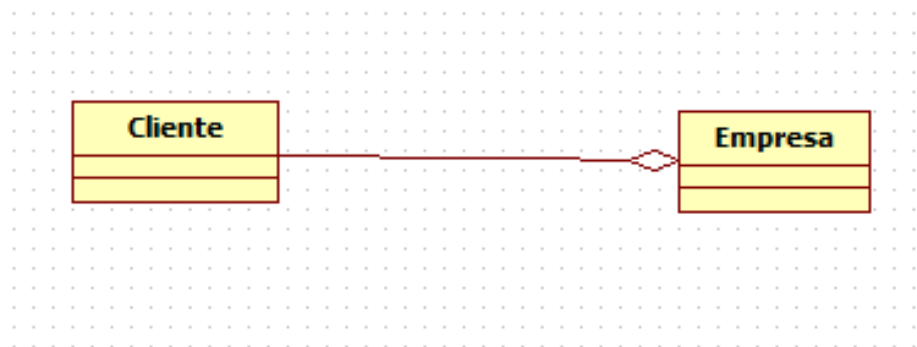
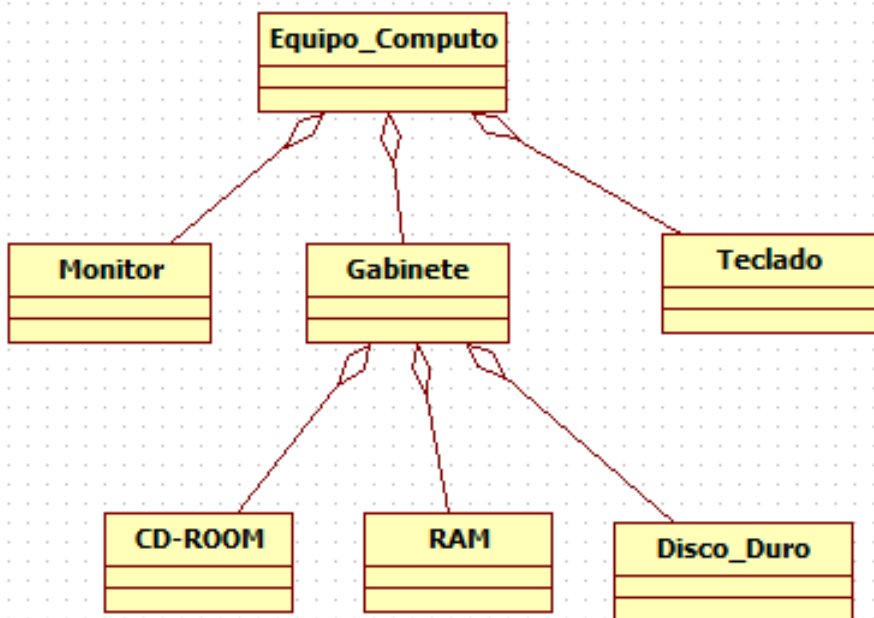
Tema: Diagramas de Clases



Tipos de Relaciones entre clases (Envío de mensajes).

Agregación o Acumulación

Es un tipo de **asociación** que indica que una **clase** es parte de otra **clase** (composición débil)





Tema: Diagramas de Clases

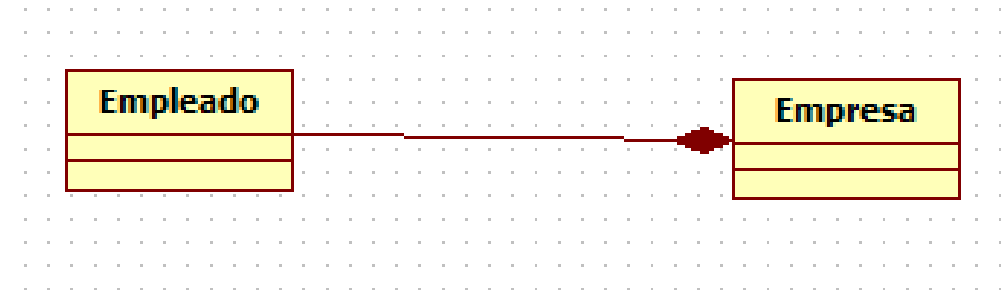
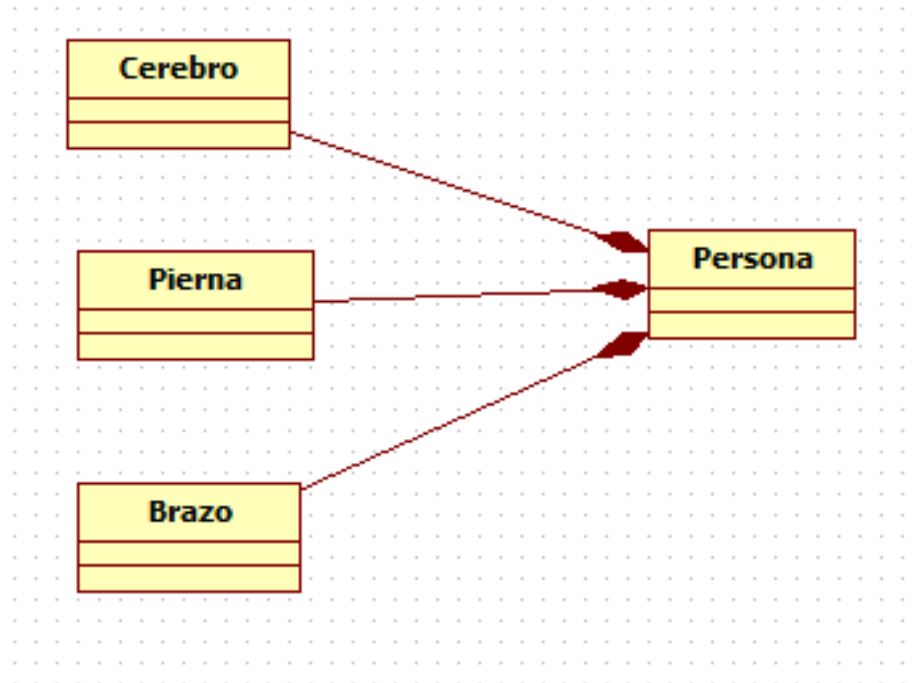


Tipos de Relaciones entre clases (Envío de mensajes).

Composición

x
x
x

Composición es una forma fuerte de **Agregación** donde la vida de la **clase** contenida debe coincidir con la vida de la **clase** contenedor. En la composición fuerte (Composición), los componentes no pueden compartirse y la destrucción del objeto compuesto conlleva la destrucción de sus componentes..





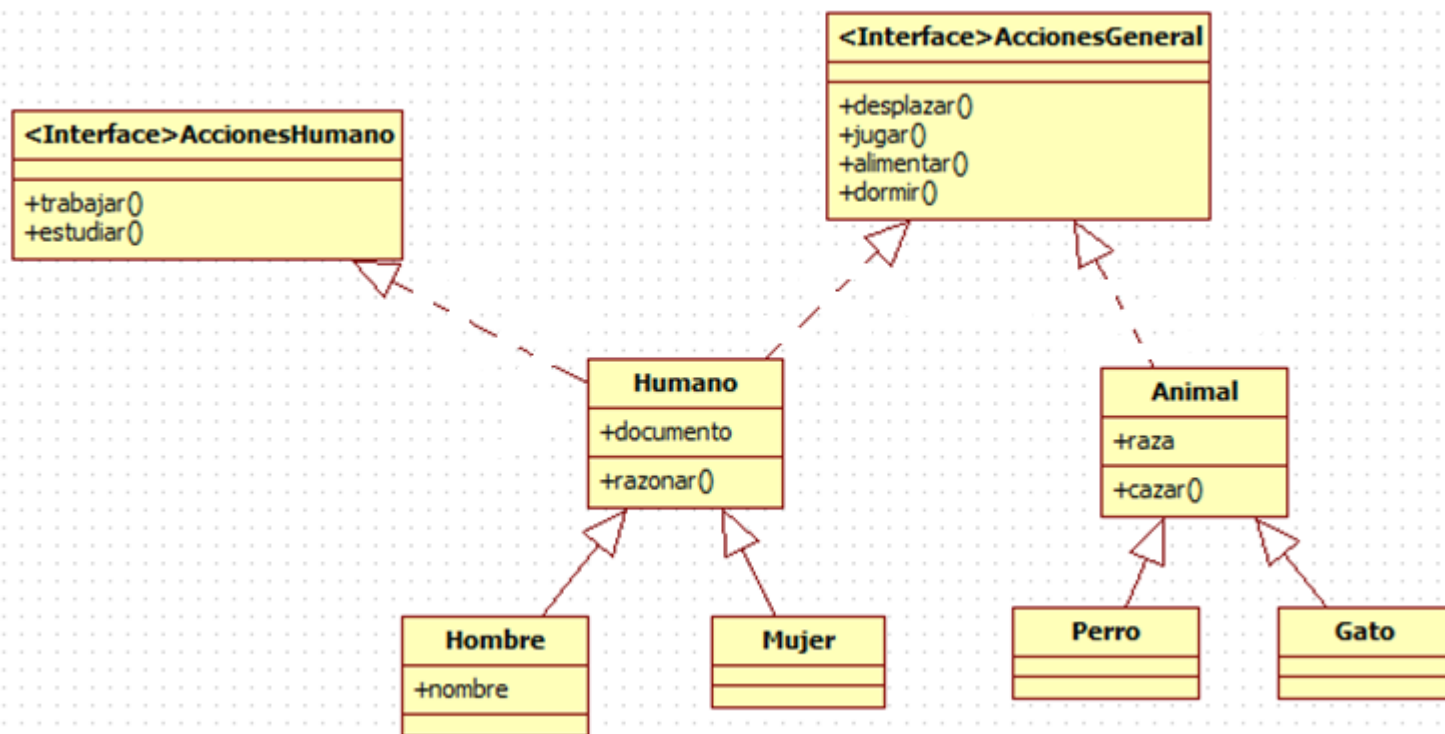
Tema: Diagramas de Clases



Tipos de Relaciones entre clases (Envío de mensajes). Realizaciones - Interfaces

x
x
x

Es una relación que se implementa a través de una **interfaz**, concepto nos permite especificar operaciones en una clase y su forma de presentarlas a otras.





Tema: Diagramas de Clases

Diagramas de Clase: Ejemplo 1

Representa mediante un diagrama de clases la siguiente especificación.

Una **aplicación** necesita almacenar información sobre **empresas**, sus **empleados** y sus **clientes**. Las empresas tiene un **identificador** y **nombre**. Ambos (empleados y sus clientes) se caracterizan por su **identificación**, **nombre**, **edad**. Los empleados tienen un **salario**. El empleado tiene una **experiencia**, de la cual se conoce la **descripción** y **tiempo** y el cliente posee un **historial de ventas**, de las cuales se conoce una **descripción** y **valor vendido**. De los clientes, además se necesita conocer su **teléfono** de contacto y el **correo** electrónico. La aplicación necesita **mostrar los datos** de empresas, empleados, experiencia, clientes e historial de ventas.

Además, sobre las empresas, empleados y clientes se deben realizar las operaciones de **incluir**, **modificar** y **eliminar**. A los empleados se les debe realizar un proceso periódico de **incremento de salario** y a los clientes se les aplica un proceso de **sanción**, cuando no cumplen con las metas estipuladas o comenten alguna falta.

Es importante el proceso de **actualización** de la experiencia para los empleados y el historial de ventas para los clientes.



Tema: Diagramas de Clases



Procedimiento Diagrama de clases



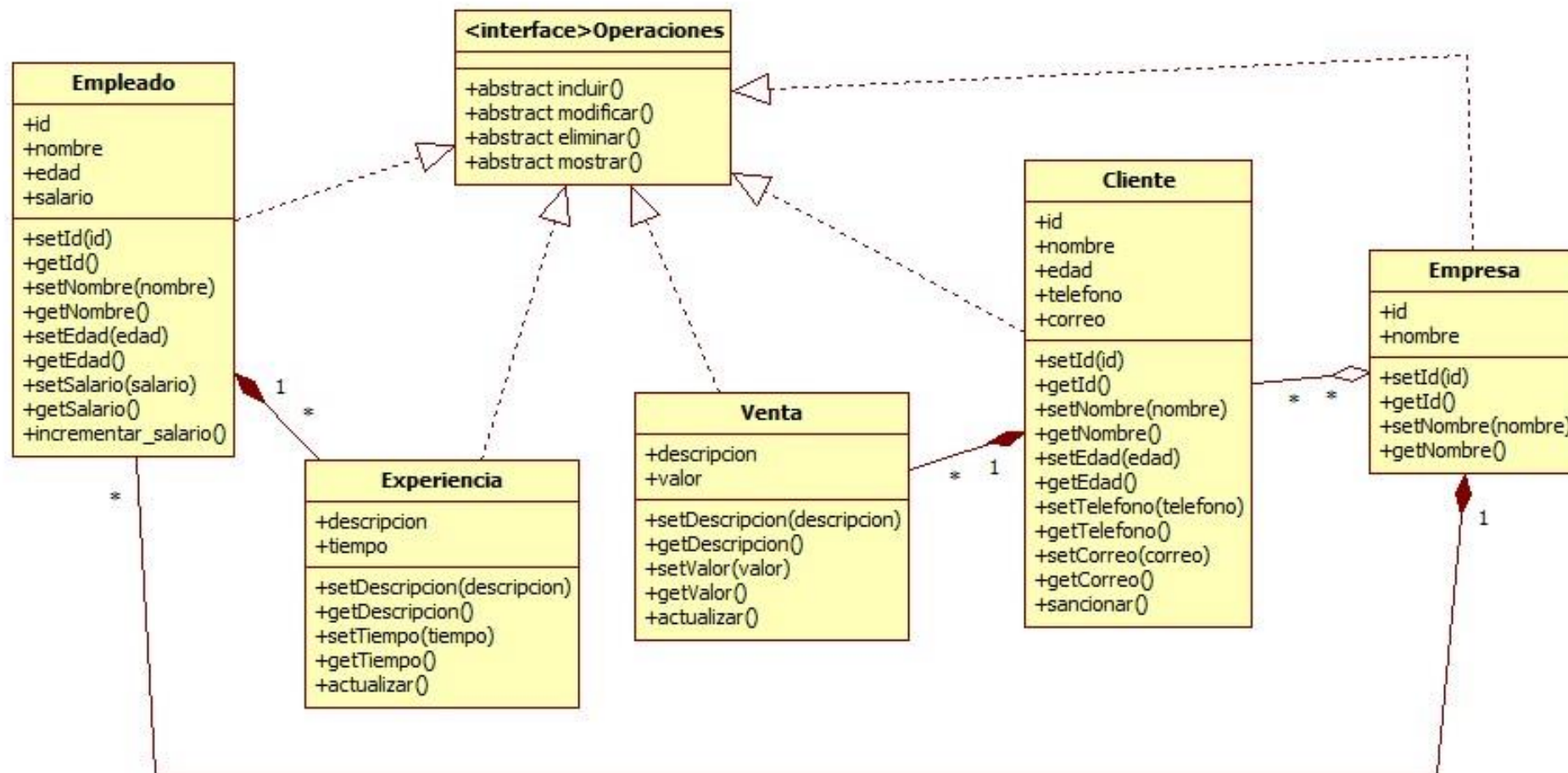
| Paso 1: Identificación de clases | Paso 2: Clases candidatos | Paso 3: Atributos | Paso 4: Métodos |
|----------------------------------|---------------------------|---|--|
| Aplicación | | | |
| Empresa | Empresa | identificador, nombre | incluir, modificar, eliminar, mostrar |
| Empleado | Empleado | identificador, nombre, edad, salario | incluir, modificar, eliminar, mostrar, incrementar_salario |
| Cliente | Cliente | identificador, nombre, edad, telefono, correo | incluir, modificar, eliminar, mostrar, sancionar |
| Experiencia | Experiencia | descripcion, tiempo | mostrar, actualizar |
| Venta | Venta | descripcion, valor | mostrar, actualizar |



Tema: Diagramas de Clases



Diagrama de Clases





Tema: Diagramas de Casos de Uso



Definición.



En el lenguaje de modelado unificado (UML), los **casos de uso** son modelos que describen como diferentes tipos de usuarios interactúan con un sistema informático para resolver un determinado problema o necesidades. Por lo tanto, describen los objetivos de los usuarios, las interacciones entre los usuarios y el sistema, así como el comportamiento que debe ejecutar el sistema para satisfacerlos.

Los casos de uso son una de las **técnicas para especificar los requerimientos funcionales del software**. Sólo describen lo que el software debe hacer y para quien. No entran en detalles sobre como el software será implementado, cuales componentes tendrá o la interacción entre ellos.

Los **casos de uso están compuestos de 3 elementos: Actores, casos de uso y relaciones**. Así mismo, una especificación de requerimientos funcionales elaborada bajo la técnica de casos de uso debe estar compuesta por:

- El diagrama de casos de uso.
- La descripción de los actores.
- La especificación de casos de uso.



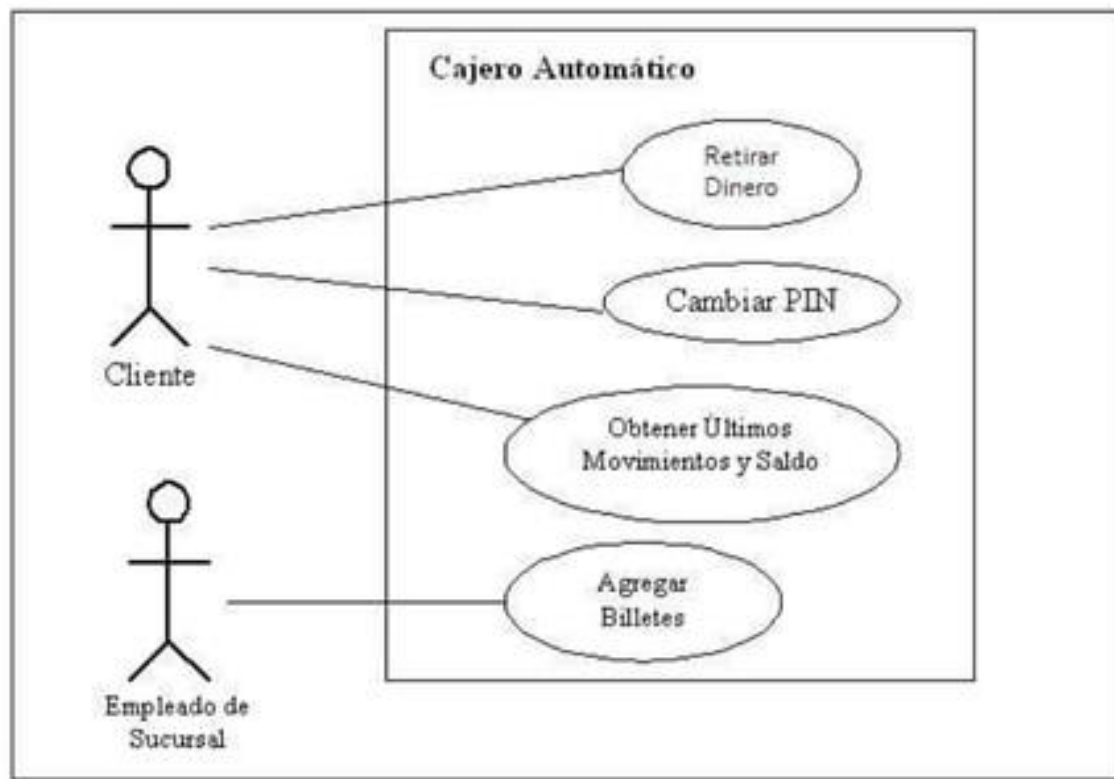
Tema: Diagramas de Casos de Uso



Representación.

x
x
x

Caso de Uso





Tema: Diagramas de Casos de Uso

Diagramas de Casos de uso: Ejemplo 1

En un **sistema de información para el manejo de un Aula Virtual de Economía Solidaria**, se especifica que el **estudiante** inicialmente debe **iniciar sesión**, lo que **incluye** la **validación de su usuario y contraseña** y el **registro en el Log de auditoría**. Una vez haya ingresado al sistema podrá realizar varias actividades como: **Revisar Agenda de Actividades**, **Revisar Contenidos**, **Participar en los foros**, **Revisar y contestar tareas**, **Revisar y contestar evaluación**, **Revisar lista de recursos**, que **incluye** **visualizar videos**, **revisar libros electrónicos multimediales** y **revisar documentos de apoyo**. Además, puede **Verificar porcentaje de progreso del curso** y **Consultar notas del curso**.

Adicionalmente, se especifica que el **docente**, inicialmente debe **iniciar sesión**, lo que **incluye** la **validación de su usuario y contraseña** y el **registro en el Log de auditoría**. Una vez el docente ingrese al sistema podrá **Administrar contenidos del curso**, **revisar en foros**, **Administrar tareas**, **Enviar mensajes**, **Administrar recursos**, que **incluye** la **gestión de videos**, **incluir libros electrónicos multimediales** e **incluir documentos de apoyo**. Además, puede **Publicar notas del estudiante** y **Administrar evaluación de los cursos**, que **incluye** la **generación de evaluaciones** y la **calificación de las evaluaciones**.

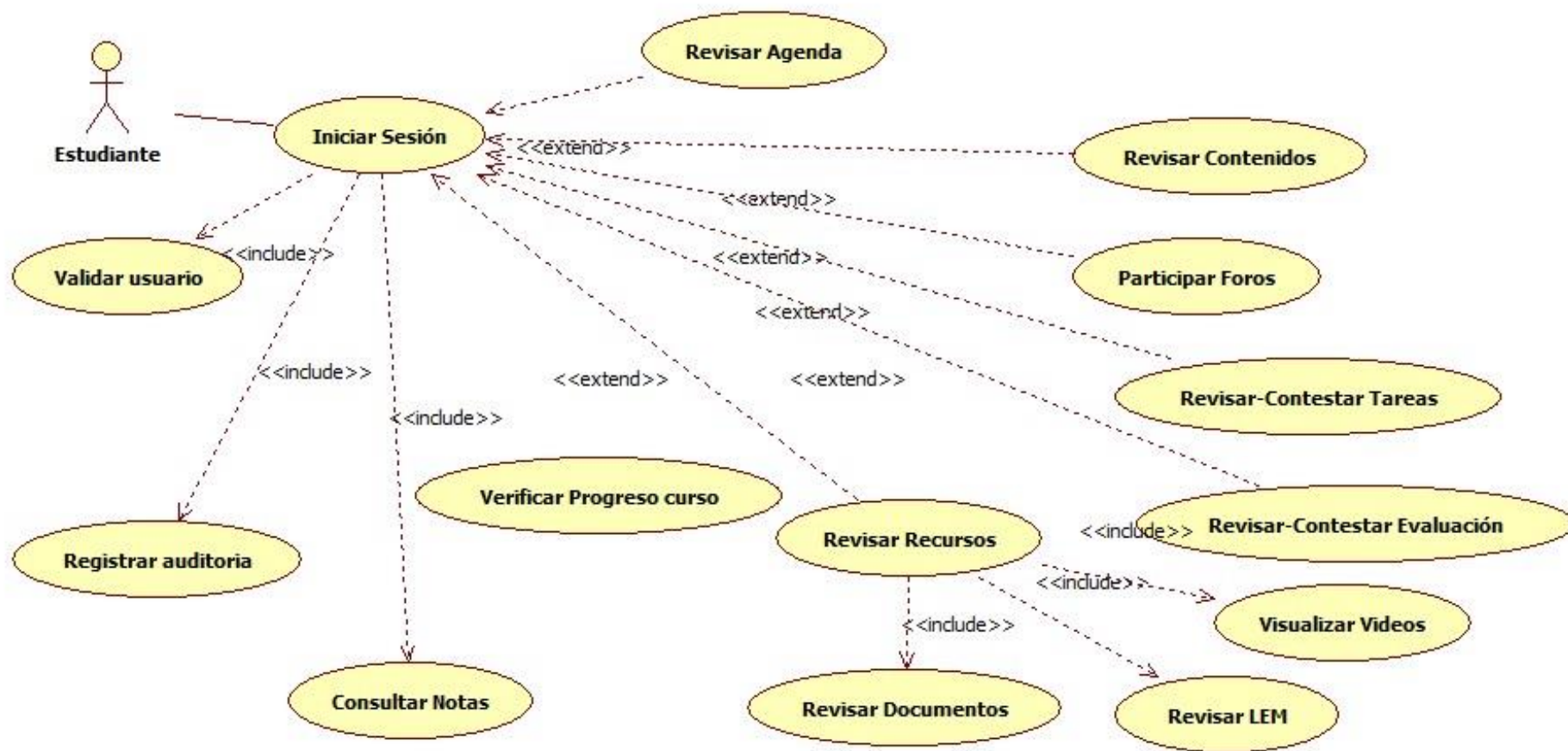
Finalmente, se especifica al **administrador del sistema**, quien inicialmente debe **iniciar sesión**, lo que incluye la validación de su usuario y contraseña y el registro en el Log de auditoría. Posteriormente, puede gestionar la base de datos, lo que incluye las operaciones de incluir, modificar y eliminar. Además, realizar los procesos de gestionar la seguridad (Usuario, contraseñas) y gestionar el proceso de copias de respaldo. También es el encargado que generar los informes estadísticos sobre los resultados de los estudiantes.



Tema: Diagramas de Casos de Uso



Diagrama de Casos de Uso Estudiante





El futuro digital
es de todos

MinTIC

Ciclo: Programación Básica
Diseño Orientado a Objetos

»
Misión TIC 2022

xxx



**Misión
TIC 2022**