



# Realización de los Casos de Uso

*(adaptado a METRICA 3, versión 00)*

**Félix García**

**Universidad de Castilla-La Mancha**

**Escuela Superior de Informática de Ciudad Real**

**Departamento de Tecnologías y Sistemas de Información**



- **Introducción**
  - ¿Qué es la realización de los casos de uso?
  - Etapas en la Realización
- **El Modelo de Análisis**
  - Elementos
  - Clases de Análisis
  - Realización de Casos de Uso – Análisis
- **El Modelo de Diseño**
  - Elementos
  - Clases de Diseño
  - Realización de Casos de Uso – Diseño
- **Un Caso de Ejemplo: Cajero Automático**



- Jacobson, Booch, y Rumbaugh (2000): El Proceso Unificado de Desarrollo de Software.
- CSAE(): Metodología METRICA v3. Técnicas y Prácticas. Ministerio de Administraciones Públicas, Consejo Superior de Administración Electrónica.
  - <http://www.csaemap.es/csi/metrica3/index.html>.

### ¿Qué es la Realización de Casos de Uso?

- Una Realización en UML es la relación existente entre una especificación y su implementación
  - Se aplica a los casos de uso y a las relaciones entre clases/componentes e interfaces
- Realización de un Caso de Uso:
  - El caso de uso describe **qué** comportamiento del sistema es deseado pero no describe **cómo implementarlo**
  - Se lleva a cabo a través de las **Colaboraciones**
    - **Describen los objetos que implementan el comportamiento del caso de uso y la forma en que interactúan para implementar dicho comportamiento**



- **Colaboración:**
  - Disposición general de objetos y enlaces que interactúan dentro de un contexto para implementar un comportamiento
  
  - Está formada por:
    - Parte estática
      - Roles que pueden desempeñar los objetos y enlaces en una instancia de la colaboración
      - Modelado UML: Diagramas de Clases y de Objetos
  
    - Parte dinámica
      - Una o más interacciones dinámicas que muestran flujos de mensajes a través del tiempo
      - Modelado UML: Diagramas de Interacción



- **Análisis, Diseño e Implementación para Realizar los Casos de Uso**
  - Durante el análisis y diseño, se transforma el modelo de casos de uso en un modelo de análisis y en un modelo de diseño
    - Estructura de clasificadores y realizaciones de cdu
  
  - **Objetivo:** Realizar los casos de uso de una forma económica de manera que el sistema ofrezca un rendimiento adecuado y pueda evolucionar en el futuro



- El modelo de análisis crece incrementalmente a medida que se analizan los casos de uso
- Dado uno o un conjunto de casos de uso, el modelo de análisis incluye las clases de análisis y sus relaciones que realizan los casos de uso.
- Elementos:
  - Clases de Análisis
  - Realizaciones de Casos de Uso - Análisis



- Se centran en el tratamiento de requisitos funcionales (los no funcionales se posponen al diseño)
- Su comportamiento se suele describir de forma general mediante responsabilidades (descripción textual)
  - Raramente se incluyen métodos con signatura
- Su estado se describe con atributos, pero a un nivel de abstracción alto
- Participan en relaciones (más conceptuales que en diseño e implementación)
  - Por ejemplo, la navegabilidad de una asociación a este nivel no es importante



- Tipos:

- Clases de Interfaz

- Modelan la relación del Sistema con sus Actores
    - Son generalmente las abstracciones de Ventanas, Formularios, Paneles, Interfaces de comunicaciones, Interfaces de Impresoras, APIs.
    - Cada clase de interfaz debe asociarse como mínimo a un actor y viceversa
    - En UML se representan con el estereotipo <<Boundary>> o el icono |—○
    - Ejemplo:

- La siguiente clase de interfaz "IU Solicitud de Pago" se utiliza para cubrir la interacción entre el actor Comprador y el caso de uso "Pagar Factura"




- Clases de Entidad

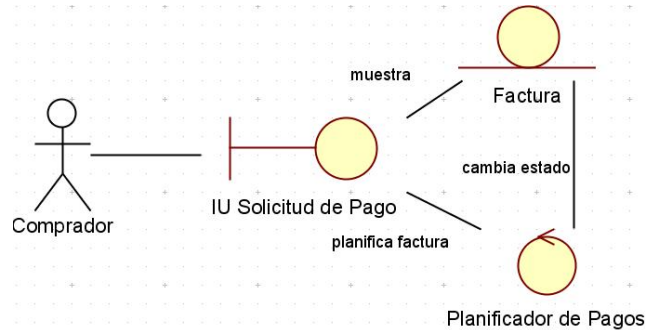
- Se utilizan para modelar información persistente y su comportamiento asociado
      - Persona, objeto del mundo real, ..
    - En la mayoría de casos las clases de entidad se derivan de una clase de entidad del negocio (modelo de objetos de negocio)
      - Con la diferencia de que las clases de entidad son las que maneja el sistema y no tiene por qué coincidir exactamente con su definición a nivel de negocio
    - Se representan con el estereotipo <<Entity>> o con el icono ○
    - Ejemplo:



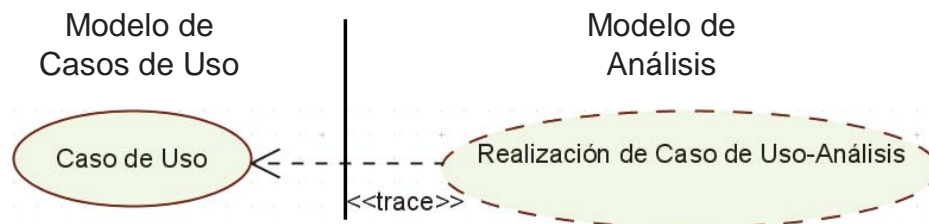


#### Clases de Control

- Representan coordinación, secuencia, transacciones y control de otros objetos
- Se usan para encapsular el control de un caso de uso
- También se usan para representar acciones que no se pueden asociar con ninguna información concreta (clase de entidad)
- Coordinan las acciones y flujos de control principales y delegan el trabajo a otros objetos (de interfaz y entidad)
- Se representan con el estereotipo <<Control>> o el icono 
- Ejemplo:

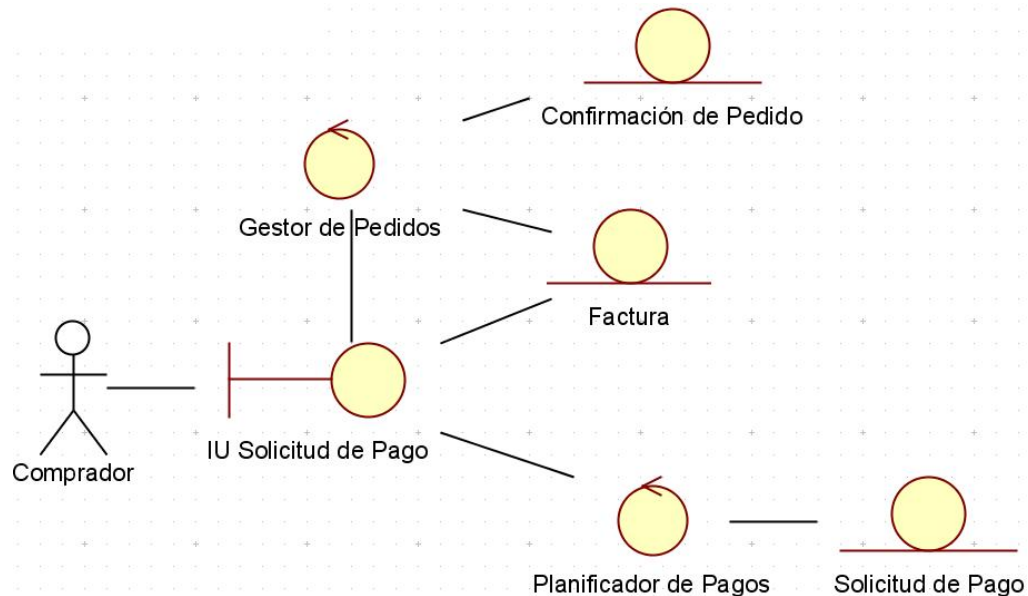


- Colaboración entre clases y objetos de análisis en interacción para llevar a cabo un caso de uso



- Se representan mediante:
  - Diagramas de Clases (parte estática)
  - Diagramas de Interacción (parte dinámica)

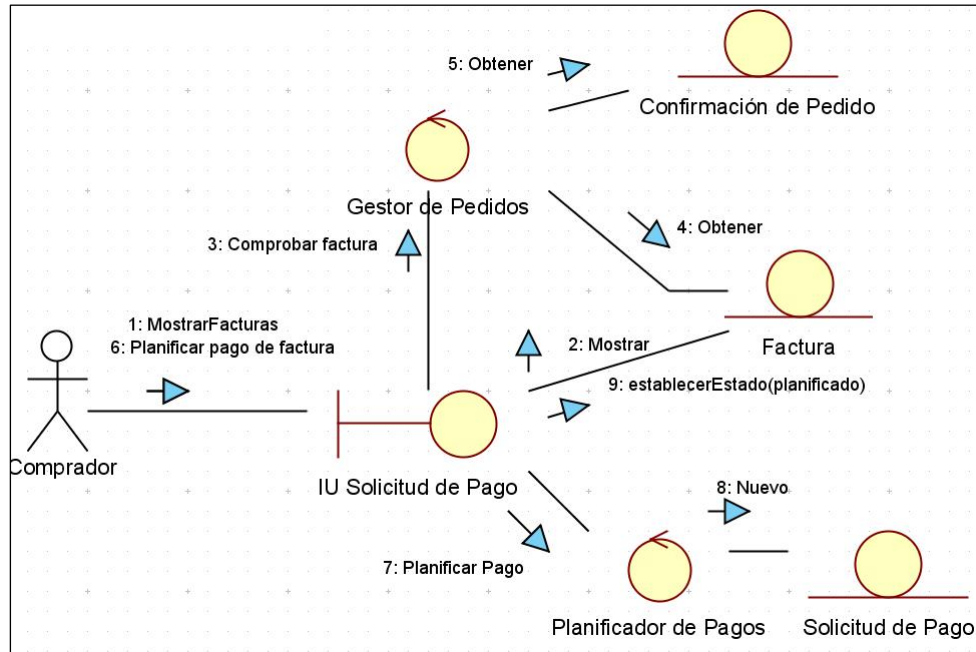
• Diagrama de Clases (Análisis)



• Descripción de la Interacción:

- (1,2) El comprador consulta a través del *IU Solicitud de Pago* las facturas gestionadas por el sistema
- (3, 4, 5) El *IU Solicitud de Pago* utiliza el *Gestor de Pedidos* para comprobar las *facturas* con sus correspondientes *confirmaciones de pedido* antes de mostrar la lista de facturas al comprador. En función de la comprobación anterior el *Gestor de Pedidos* decide qué hacer actuando
- (6) El comprador selecciona una *factura* mediante el *IU Solicitud de Pago* y planifica su *pago*
- (7) El *IU Solicitud de Pago* solicita al *Planificador de Pagos* que planifique el *pago* de la *factura*
- (8) El *Planificador de Pagos* crea una *Solicitud de Pago*
- (9) El *IU Solicitud de Pago* cambia el estado de la *Factura* a “planificado”

• Diagrama de Comunicación (Interacción Análisis)



- Se modela el sistema para dar soporte a los requisitos (incluyendo los no funcionales y otras restricciones)
- La entrada esencial es el Modelo de Análisis
- Los elementos más importantes del modelo de diseño son:
  - Clase de Diseño
  - Realización de Caso de Uso-Diseño



- Es una abstracción de una clase a un nivel cercano a la implementación
  - El lenguaje para especificarla es similar a un lenguaje de programación
  - Se especifica la visibilidad de los atributos y operaciones
  - Tienen un significado directo cuando la clase es implementada
    - Sus relaciones con otras clases:
      - Ej. Generalización → *Extends* en JAVA
    - Sus métodos
      - En diseño se pueden especificar con pseudocódigo
  - Puede realizar interfaces (si tiene sentido en el lenguaje de programación, como por ej en JAVA)

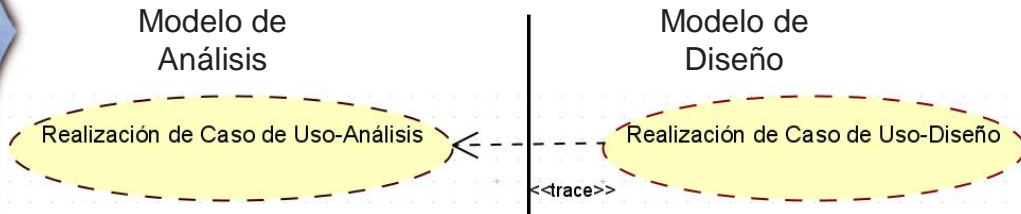


- La Clase de Diseño Factura:

Factura
-cantidad : double
-fechadepago : Date
-fechadeenvío : Date
+creacion(cantidad : double, fechapago : Date)
+envío(comprador : TipoComprador)
+planificación(horadepago : Date)
+cerrar()



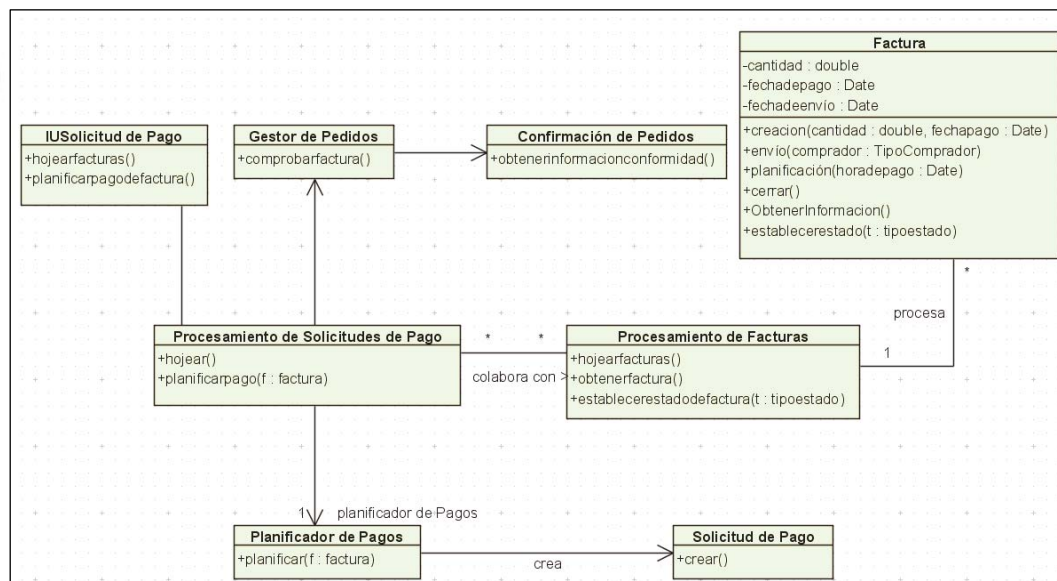
- Colaboración entre clases y objetos de análisis en interacción para llevar a cabo un caso de uso



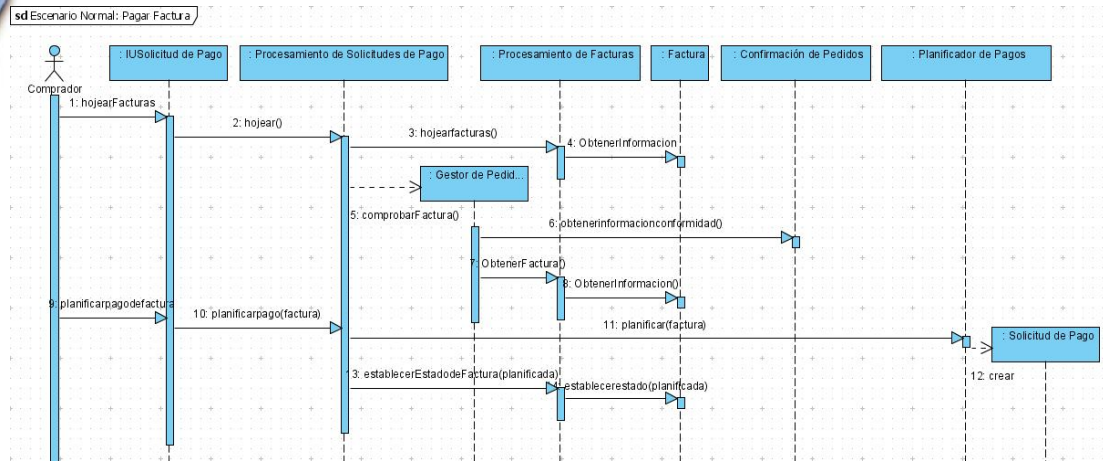
- Se representan mediante:
  - Diagramas de Clases (parte estática)
  - Diagramas de Interacción (parte dinámica)



- Diagrama de Clases (Diseño):

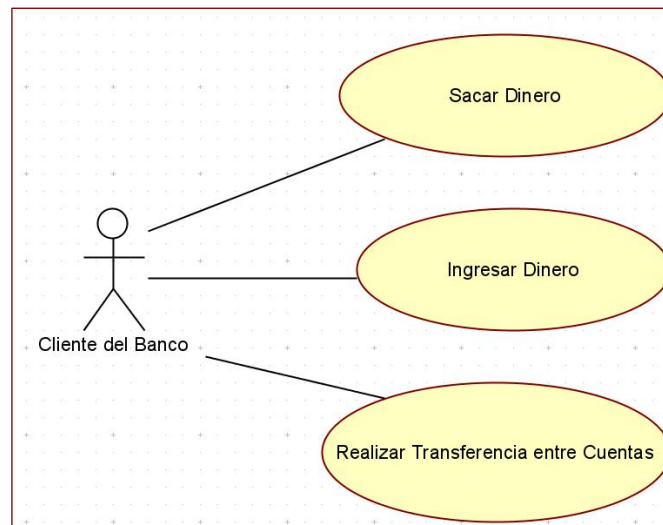


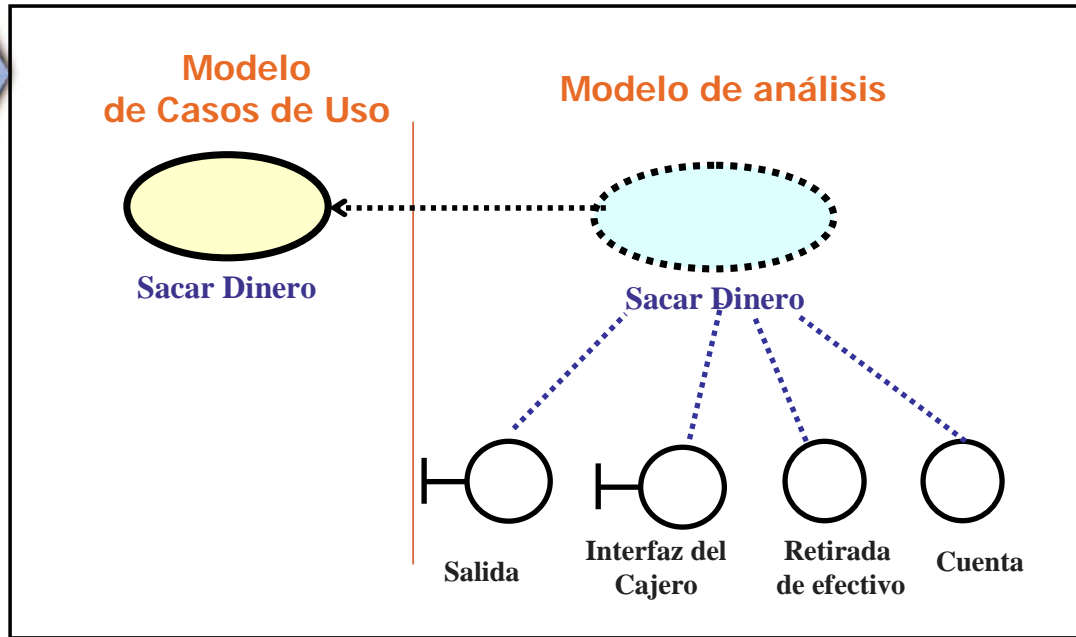
- Diagrama de Secuencia



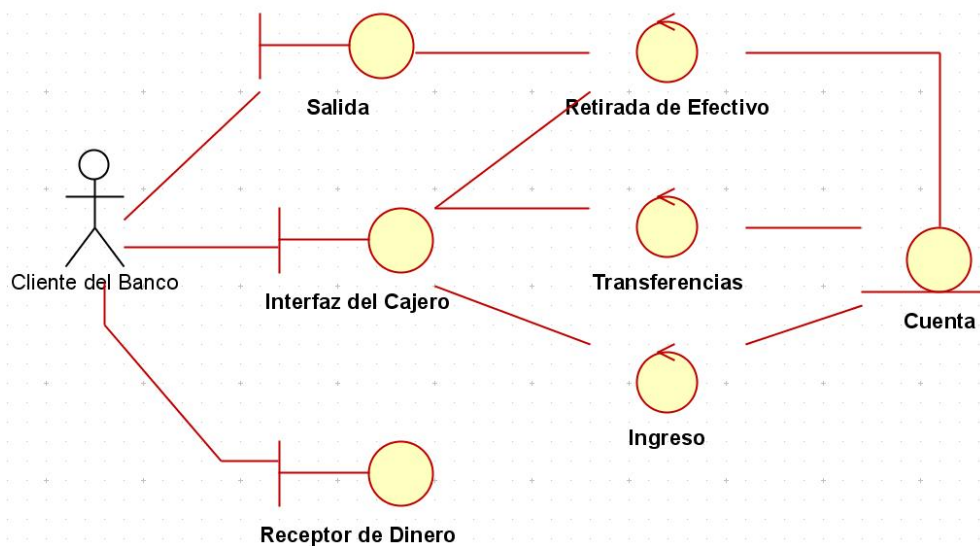
- Descripción del Sistema:

- El Cliente utiliza el Cajero Automático (CA) para retirar e ingresar dinero de sus cuentas así como para realizar transferencias entre cuentas.





- Modelo de Análisis.
  - Diagrama de Clases (parte estática)

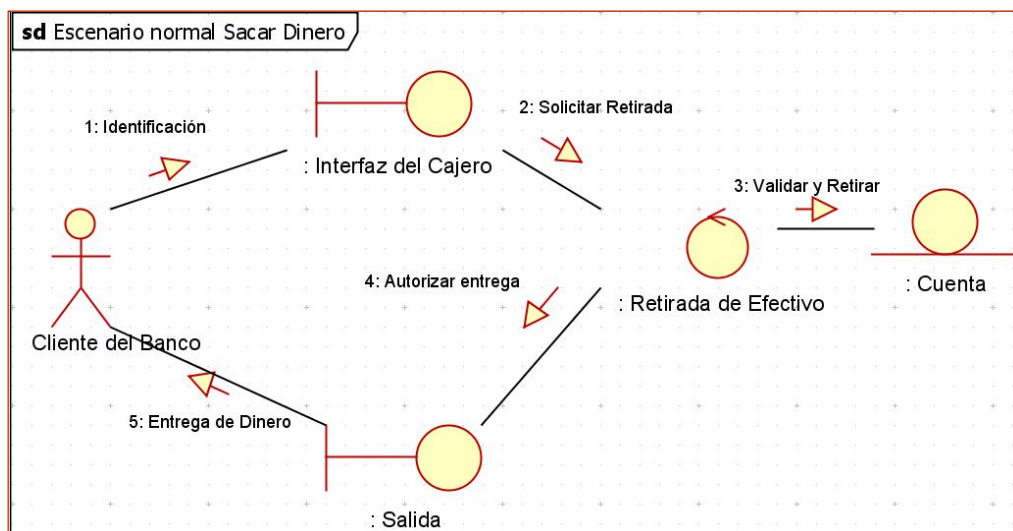




- **Modelo de Análisis.**
  - Una o más clases del modelo de análisis realizan los casos de uso del modelo de casos de uso
    - Sacar Dinero
      - Interfaz de Cajero, Salida, Retirada de Efectivo, Cuenta
    - Transferencia entre cuentas
      - Interfaz del Cajero, Transferencias, Cuenta
    - Ingresar Dinero
      - Interfaz del Cajero, Receptor de Dinero, Ingreso, Cuenta



- **Modelo de Análisis.**
  - Diagrama de Comunicación (parte dinámica)

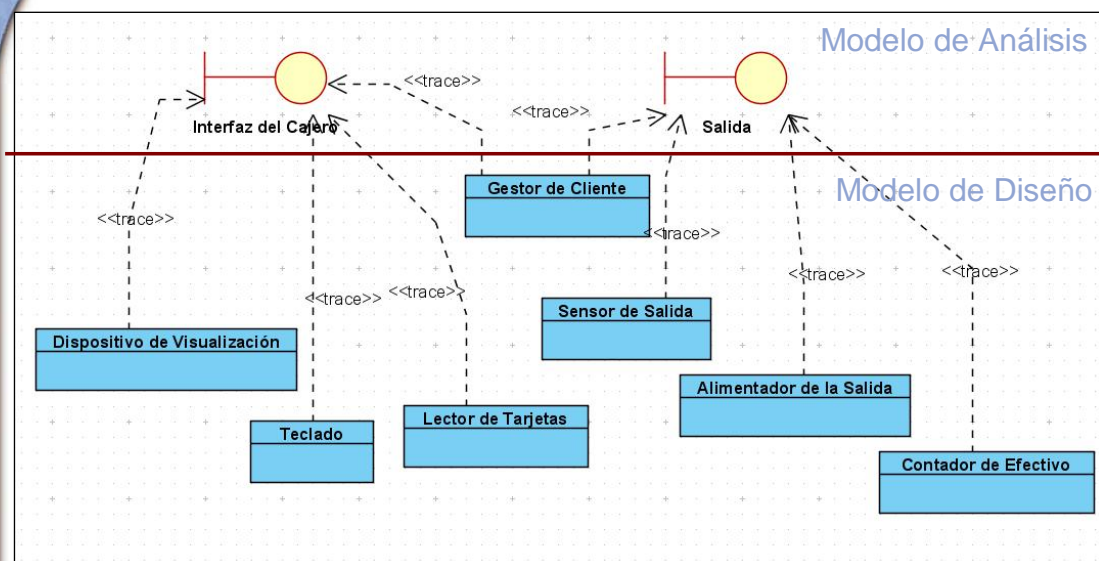




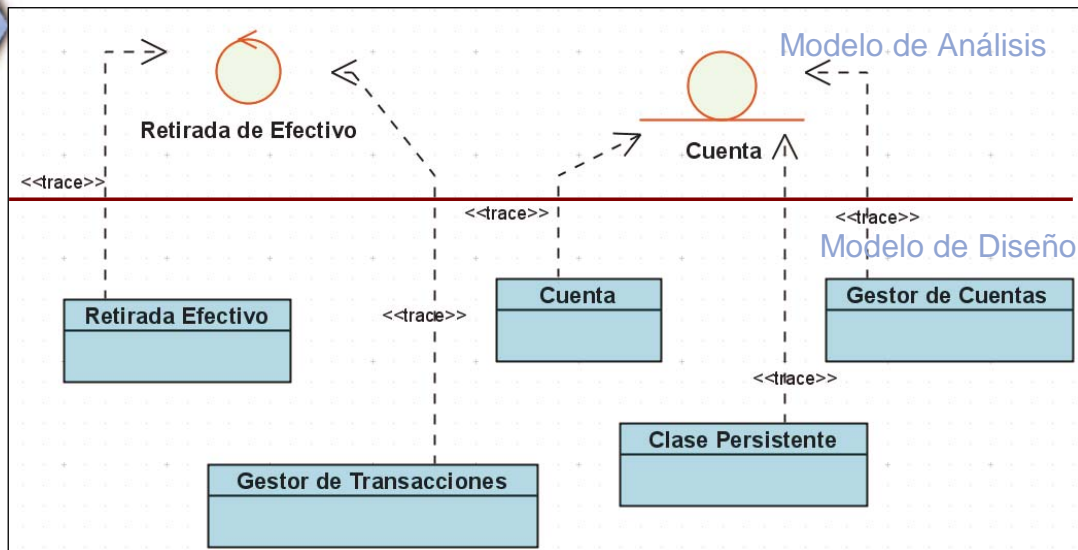
- **Modelo de Análisis.**
  - **Descripción textual del escenario normal de Sacar Dinero**
    - Un cliente del banco decide sacar dinero y activa el objeto interfaz de cajero
    - El cliente del banco se identifica y especifica la cantidad a retirar y la cuenta de la cual hacerlo
    - El interfaz del cajero verifica la identidad del cliente del banco y solicita al objeto Retirada de efectivo que lleve a cabo la transacción
    - Si la identidad del cliente del banco es válida, se le solicita al objeto Retirada de efectivo que confirme al cliente del banco que tiene derecho a sacar la cantidad especificada de la cuenta. Para ello el objeto Retirada de efectivo pide a Cuenta que valide la petición y que reste la cantidad
    - Después el objeto Retirada de efectivo autoriza a Salida que entregue al Cliente del banco la cantidad solicitada. Entonces el cliente recibe dicha cantidad



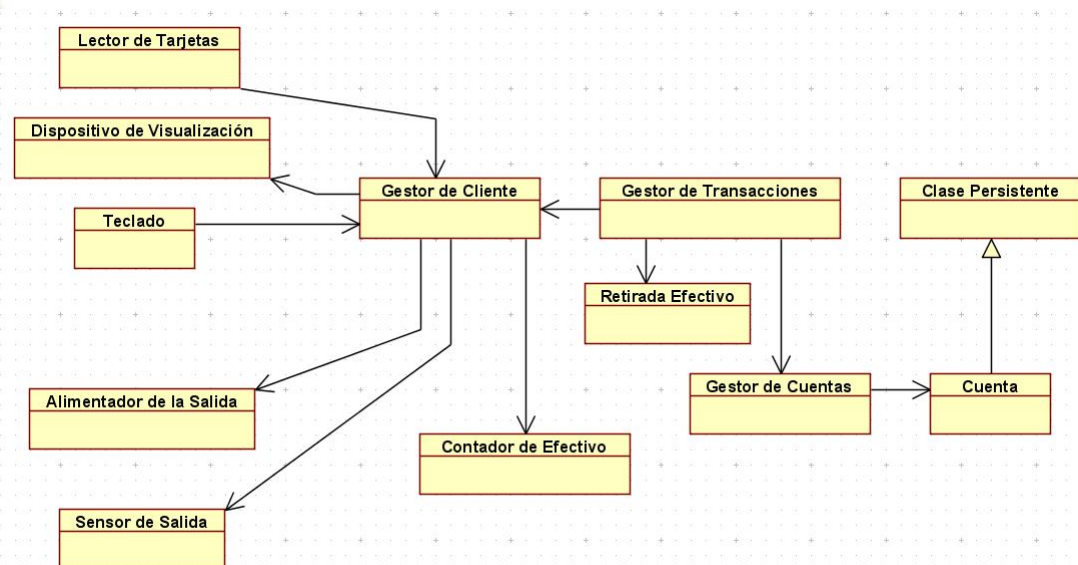
- **Modelo de Diseño vs Modelo de Análisis**
  - **Realización Caso de Uso: Sacar Dinero**



- Modelo de Diseño vs Modelo de Análisis
  - Realización Caso de Uso: Sacar Dinero

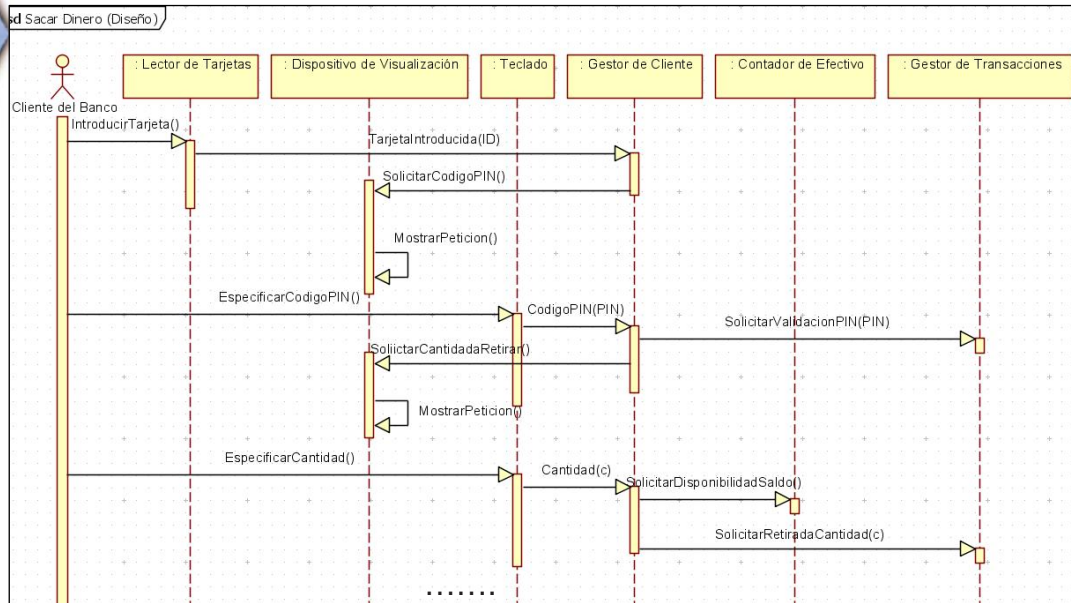


- Modelo de Diseño
  - Diagrama de Clases (alto nivel de abstracción)





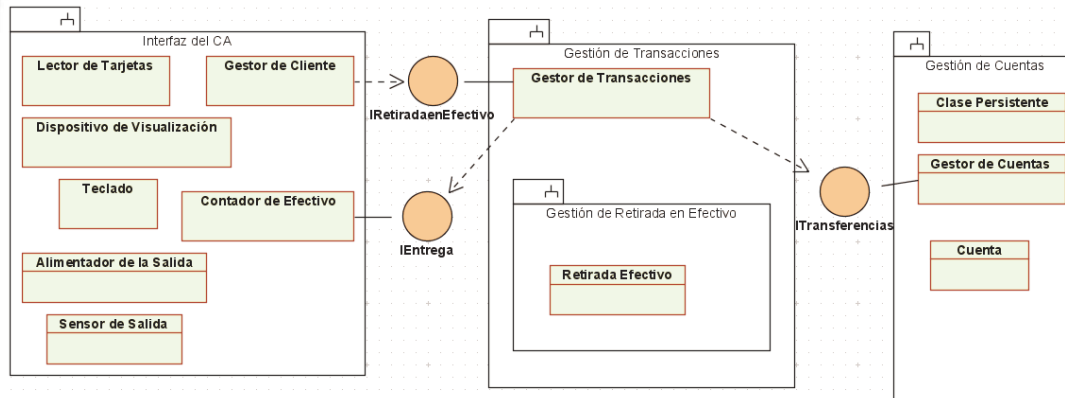
- Modelo de Diseño
  - Diagrama de Secuencia



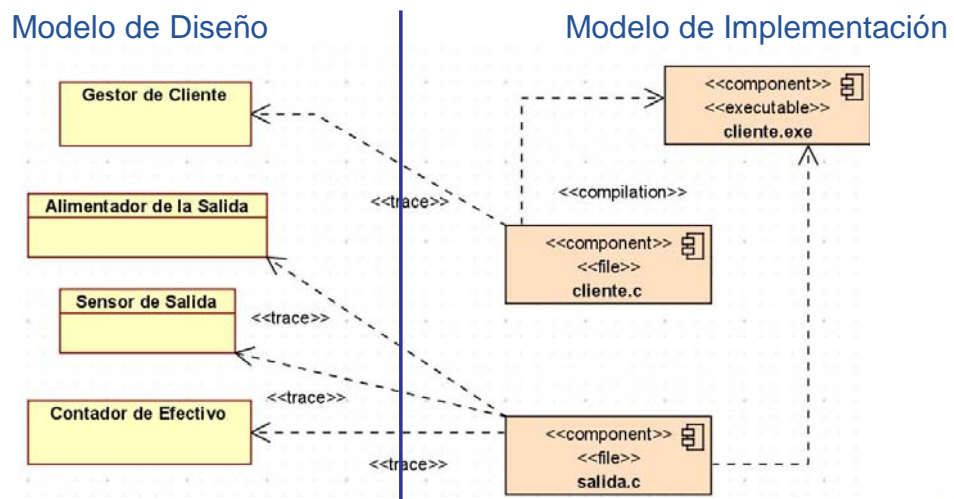
- Los subsistemas agrupan a las clases para facilitar una mayor comprensión
- Un subsistema es un agrupamiento semánticamente útil de clases o de otros subsistemas
- Posee un conjunto de interfaces que se ofrecen a los usuarios



- Separación del Sistema en tres subsistemas
  - Arquitectura de 3 capas: Presentación, Dominio, Almacenamiento



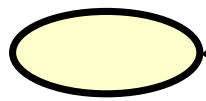
- Modelo de Implementación vs Diseño
  - El modelo de implementación incluye los elementos del sistema ejecutable: componentes ejecutables, componentes de fichero (código fuente, scripts, ..), componentes de tabla (elementos de la BBDD), etc..





- Prueba de los Casos de Uso:
  - Un modelo de prueba está formado por:
    - Casos de Prueba
      - Conjunto de entradas de prueba, condiciones de ejecución y resultados esperados
    - Procedimientos de Prueba
      - Especificación de cómo llevar a cabo la preparación, ejecución y evaluación de los resultados de un caso de prueba

#### Modelo de Casos de Uso



Sacar Dinero

#### Modelo de Prueba



Sacar Dinero – Flujo Básico

<<trace>>



- Prueba de los Casos de Uso:
  - Sacar Dinero

#### Entrada:

- La Cuenta 12-121-1211 del Cliente del Banco tiene un saldo de 350 €
- El Cliente del Banco se identifica correctamente
- El Cliente del Banco solicita la retirada de 200 € de la Cuenta 12-121-1211

#### Resultados:

- El saldo de la Cuenta 12-121-1211 del Cliente del Banco disminuye a 150 €
- El Cliente del Banco recibe 200 € del Cajero Automático

#### Condiciones:

- No se permite que otras instancias casos de uso accedan a la cuenta 12-121-1211 durante la ejecución del caso de prueba