

# Práctica clases de uso con PlantUml

## II

### Casos de Usos:

Capturan los requisitos funcionales del sistema a desarrollar.

### Objetivos

- Modelar el comportamiento del sistema
- Aprender un subconjunto de UML

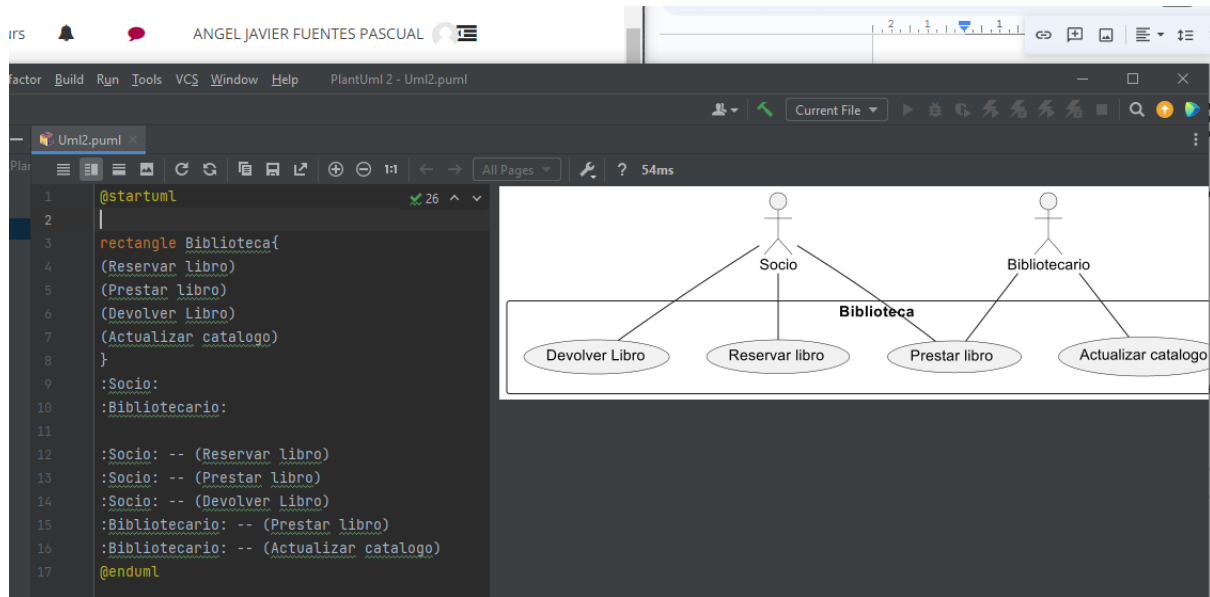
### Elementos

- Caso de Uso
- Actor
- Comunicación
- Entorno del sistema

Lo primero se ha de identificar el sistema, luego el caso de uso que son el requisito de funcionalidades que aportan a los actores externos que son los que interactúan con el sistema.

Para indicar que comunicación tiene con cada caso de uso se usan las líneas de comunicación.

Aquí tenemos un ejemplo:



Es un sistema software de gestión de una biblioteca donde tenemos dos actores.

El socio tiene 3 casos de uso que es devolver libro, reservar libro y Prestar libro, mientras que el Bibliotecario puede Prestar libro y Actualizar catálogo.

## Plantillas de Descripción

Cada caso de uso se describe utilizando plantillas en lenguaje natural.

Ejemplo.

<b>Caso de Uso</b>	Reservar Libro
<b>Actores</b>	Socio
<b>Resumen</b>	El socio puede solicitar la reserva de un libro para su posterior préstamo, a partir de una fecha determinada
<b>Precondiciones</b>	El socio no tiene ninguna reserva
<b>Postcondiciones</b>	El socio tiene una reserva y el libro tiene una nueva reserva a partir de una fecha
<b>Incluye</b>	- -
<b>Extiende</b>	- -
<b>Hereda de</b>	- -
<b>Flujo de Eventos</b>	
<i>Actor</i>	<i>Sistema</i>
1. El socio solicita la reserva (codigo libro, fecha)	2. El sistema comprueba que el socio no tiene reserva 3. El sistema comprueba que el libro esta libre para la fecha solicitada 4. El sistema solicita confirmación de la reserva
5. El socio confirma la reserva	6. El sistema realiza la reserva

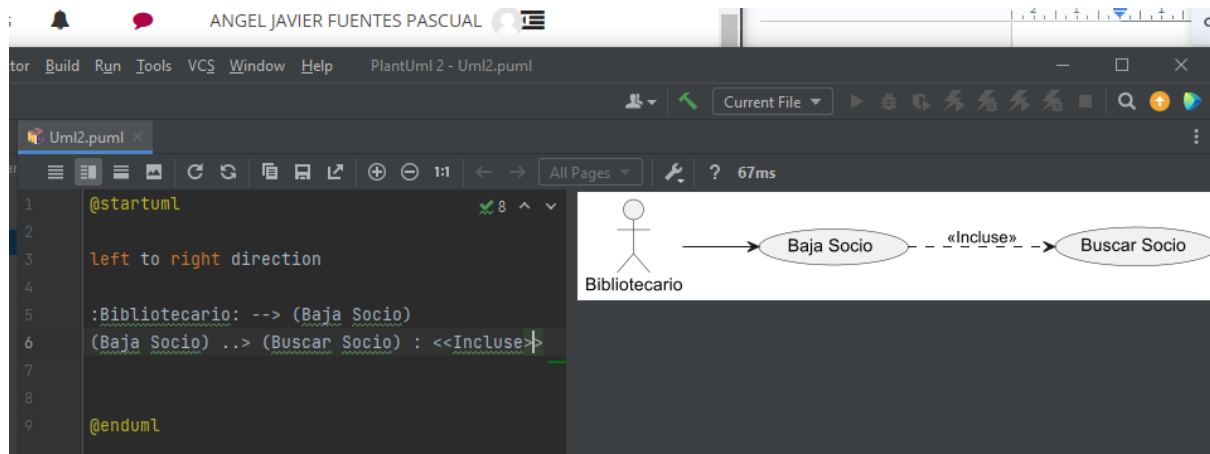
Una parte muy importante es el flujo de Eventos es decir la interacción del actor con el sistema que especifica la relación del actor con el sistema.

## Relación de casos de uso.

- Inclusión
- Extensión
- Herencia

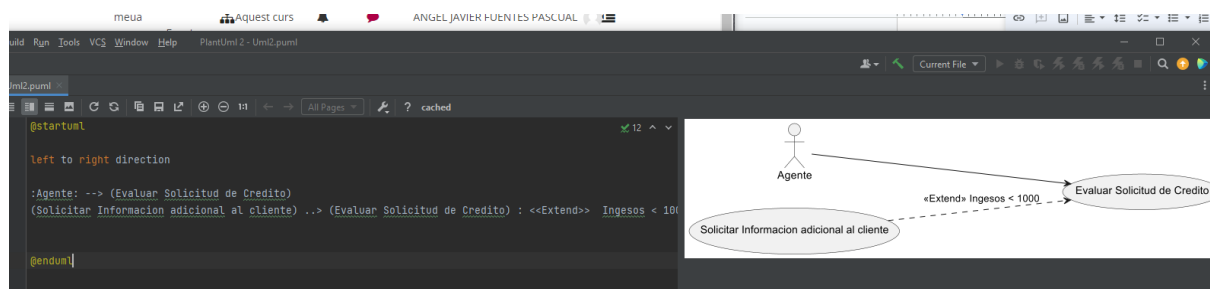
### Inclusión:

Un caso de uso incluye a otro caso de uso por lo tanto siempre que se ejecute el primer caso de uso incluirá siempre al segundo.



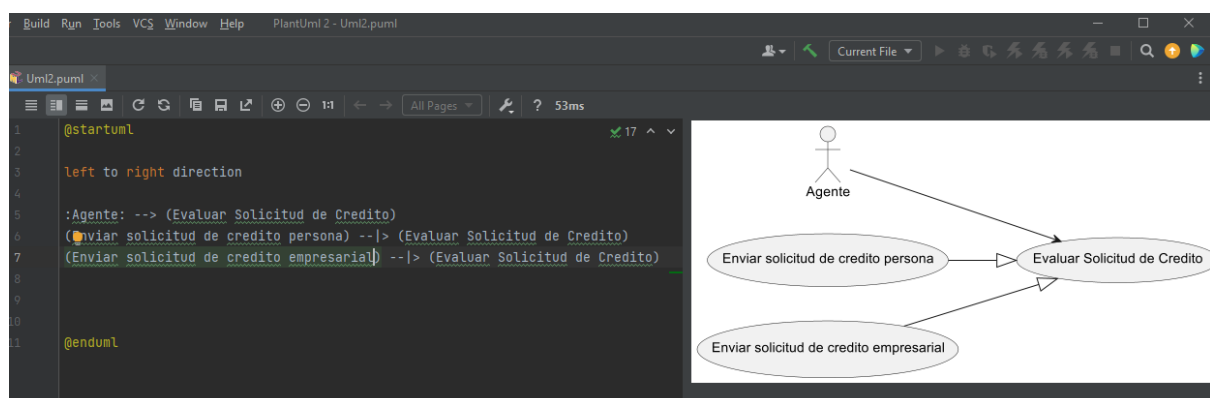
## Extensión:

Un caso de uso B extiende de un caso de uso A, en este caso **Solicitar información** extiende de **Evaluar solicitud de Crédito**, Solicitar información no se va a ejecutar siempre, solo se ejecutara si ingresos es menor de 1000.



## Herencia:

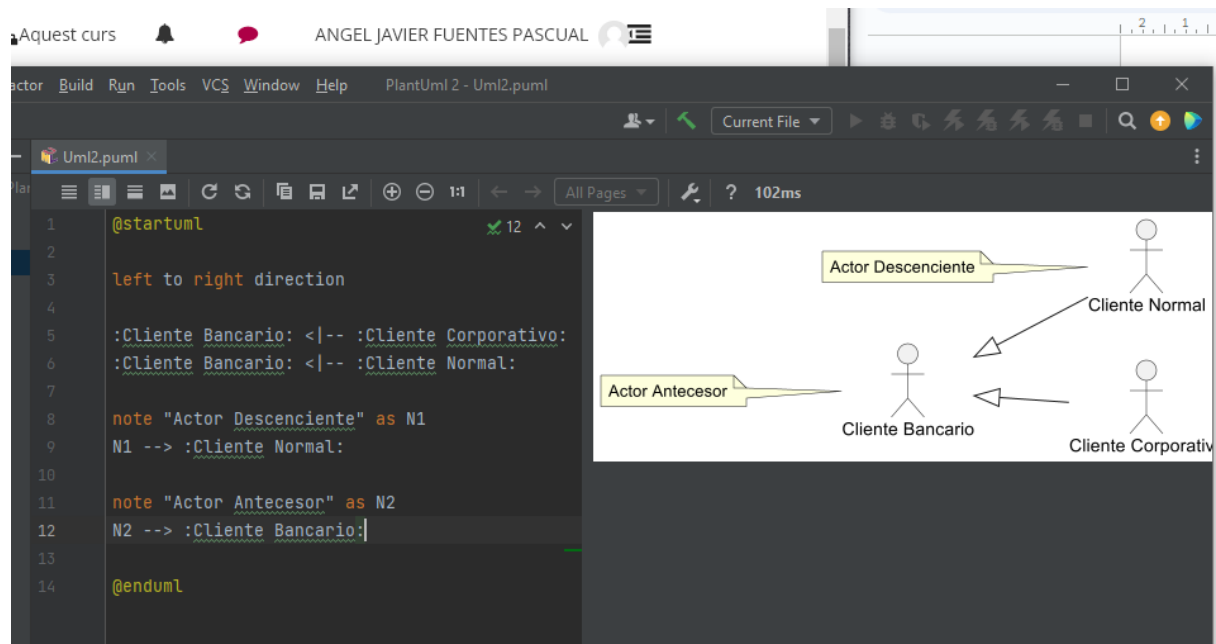
La Herencia es una especialización de casos de uso, los casos de usos especializados son refinamientos del caso uso base.



## Relación entre actores.

- Herencia

Un actor descendiente puede jugar todos los roles del actor antecesor.



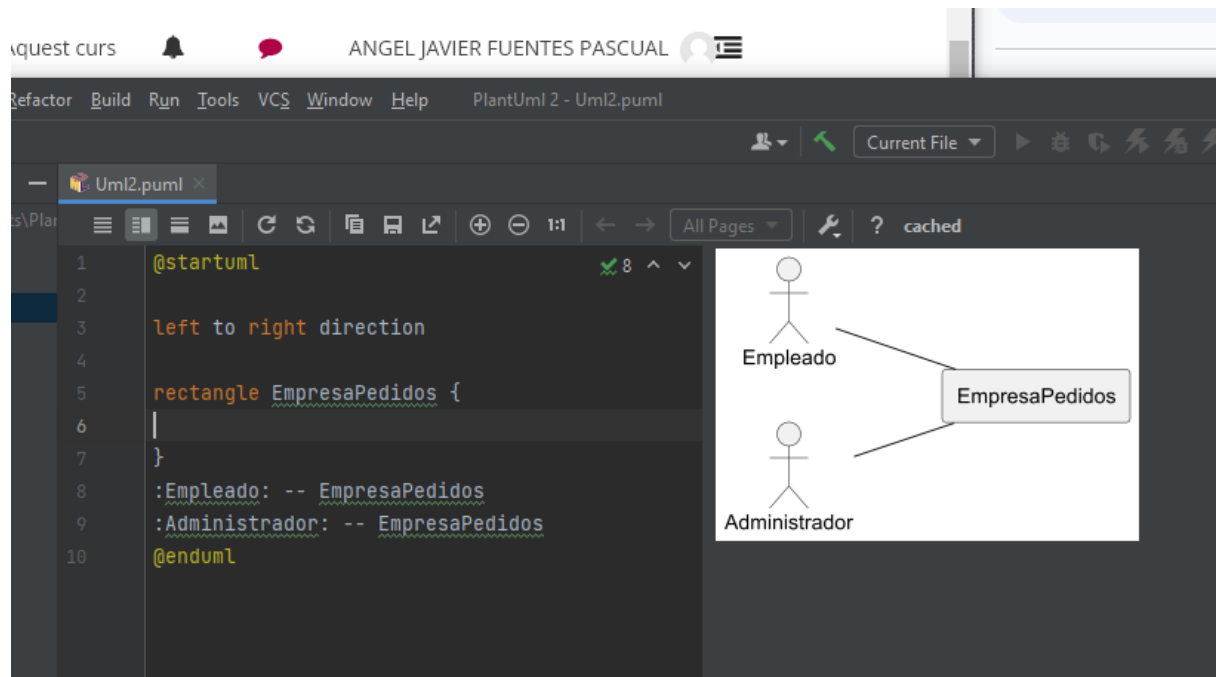
## Diagrama de Casos de Uso

- Diagrama de contexto y Diagrama inicial
- Plantillas de Descripción
- Diagrama Estructurado o Modelo de Casos de Uso

## Diagrama de Contexto

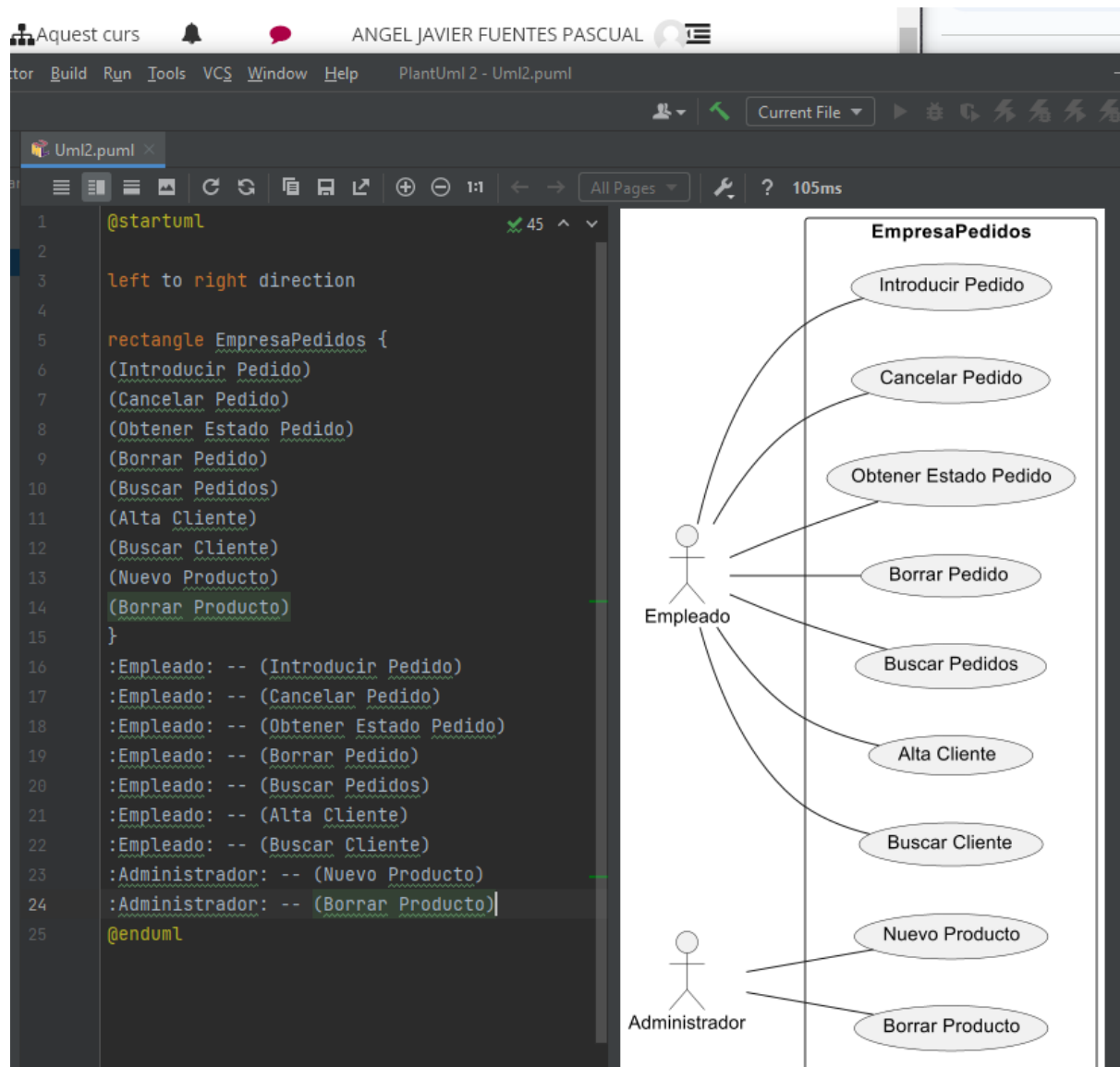
Es el que nos sirve para identificar cual es el entorno del sistema.

Una vez identificado el sistema que vamos a desarrollar buscamos los actores que van a interactuar con el sistema.



## Diagrama inicial:

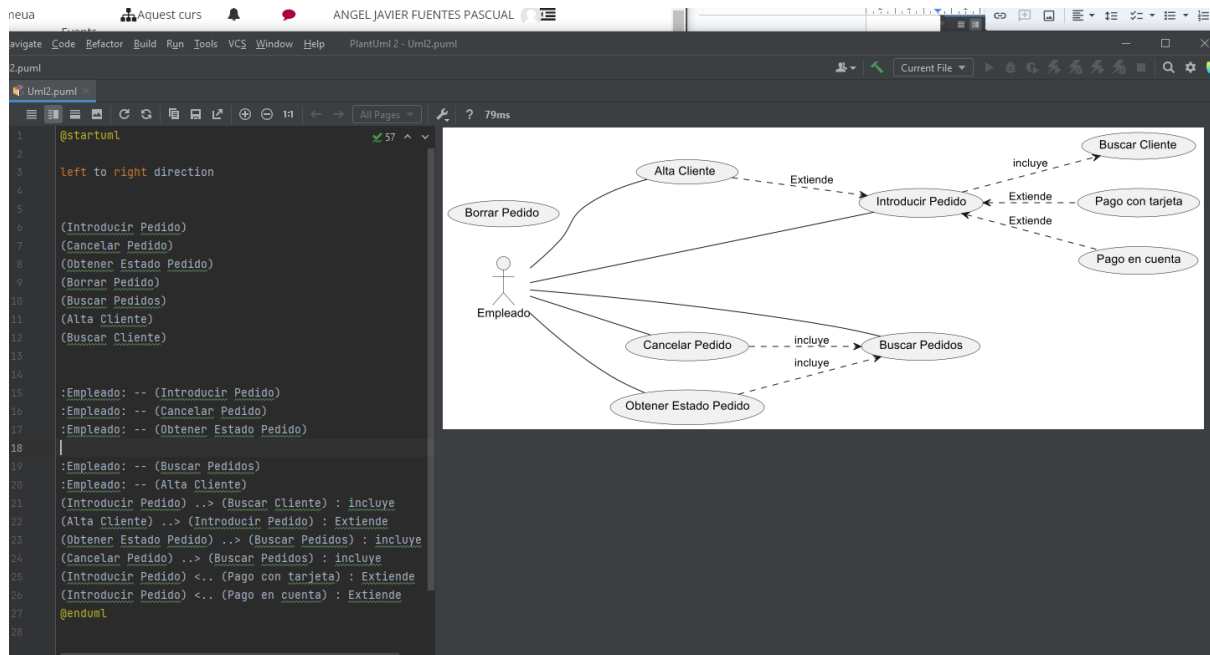
Refina el modelo anterior indicando los principales casos de uso.



## Diagrama Estructurado o Modelo de Casos de Uso

Refina aún más el diagrama anterior añadiendo las relaciones de inclusión y extensión.

Aquí se puede ver los actores, los principales casos de uso y sus relaciones.



No está completo porque no está el actor ni sus casos de uso.

## Conclusión

- El diagrama de casos de uso permite especificar las principales funcionalidades que el sistema ofrece a los actores
- UML proporciona una notación para representar dicho diagrama
  - Casos de Uso y Actores
  - Relaciones entre casos de uso
  - Relaciones entre actores
- Cada caso de uso se especifica en una plantilla de descripción textual.