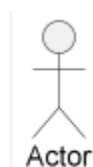


Enlace a repositorio github:

**<https://github.com/laquecu/casosDeUsoUno.git>**

A continuación vamos a mostrar una serie de diagramas de casos de uso, para comenzar con ello vamos a definir algunos de los componentes de los mismos.

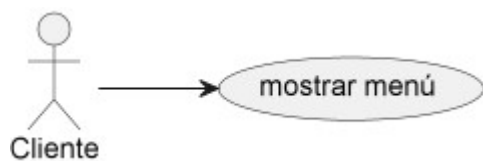
-Actor: serán lo que o quien interactua con el sistema. Usaremos “:Actor:” para crearlo con plantUML.



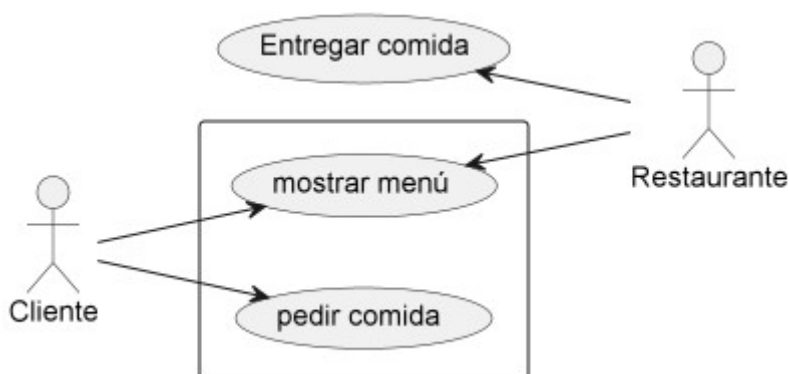
Caso de uso: corresponden a acciones generales del sistema, se representan con una elipse. Usaremos “(mostrar menú)” para crearlo en plantUML.



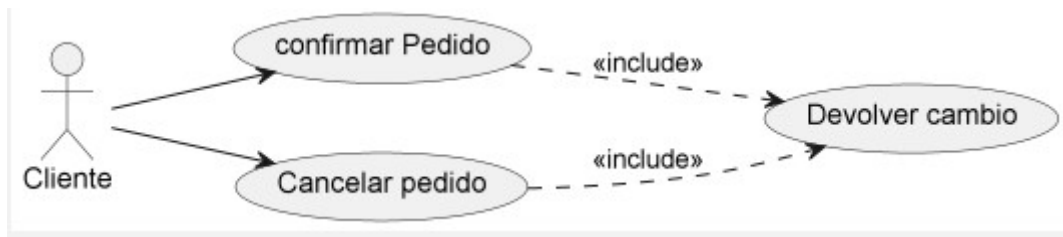
Asociación: Es la interacción entre los actores y los casos de uso. Usaremos “-->” para crearlo en plantUML.



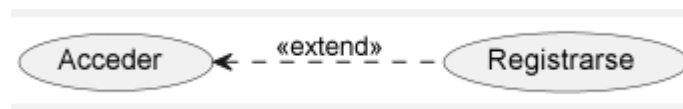
Sistema: Es el software que vamos a desarrollar, se representa como una caja rectangular. Usaremos la palabra rectangle y entre llaves pondremos lo que queramos incluir en el rectángulo para crearlo en plantUML.



Inclusión: el comportamiento de un caso de uso se incluye dentro del comportamiento de otro. Esto se hará mediante unas flechas discontinuas con la palabra <<include>> , que las crearemos con “..>” en plantUML.



Extensión: Cuando un caso aporta un comportamiento adicional en determinadas circunstancias. Se representa igual que la inclusión pero con la palabra <<extend>>.



Generalización: sirve para expresar que se puede llegar a un caso más general a través de un caso de uso más especializado. Se representa con una flecha con la punta en blanco. Esto lo haremos con “--|>” en nuestro plantUML.



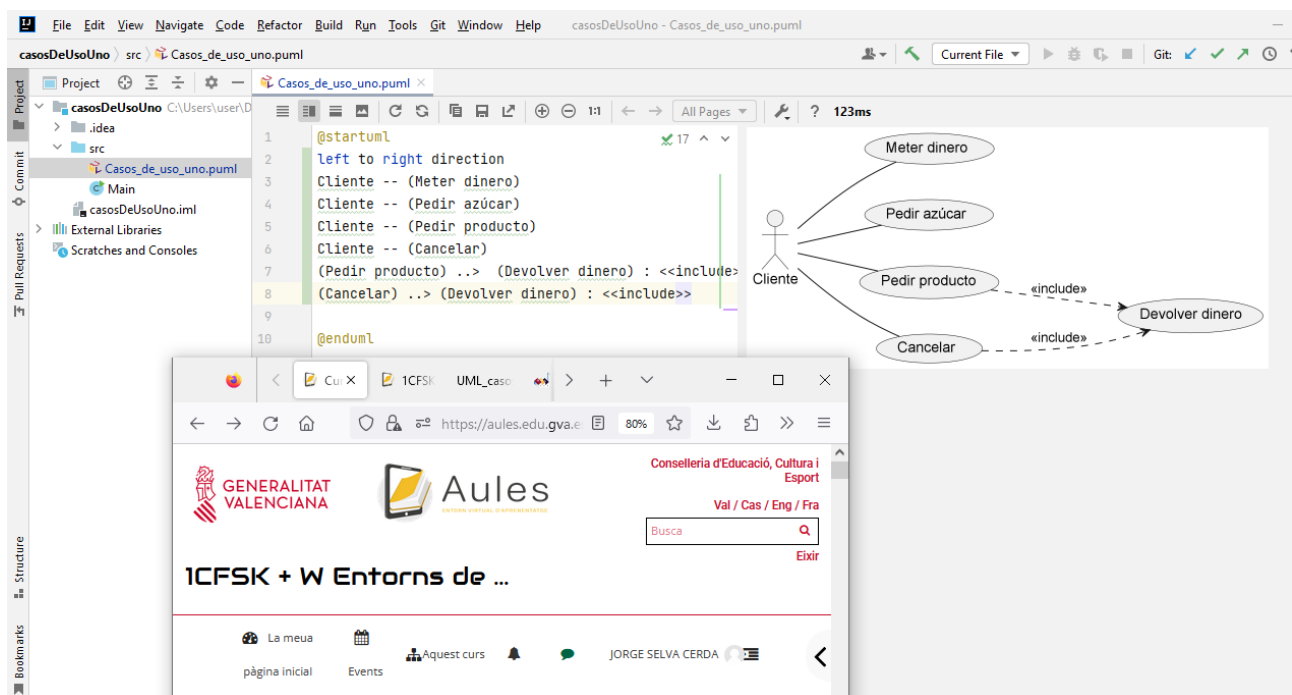
Ejemplo: La máquina de café.

Supongamos que se requiere desarrollar el control de una máquina de entrega de café automática.

La máquina debe permitir a una persona introducir dinero, escoger uno de los productos de acuerdo a su precio, escoger un nivel de azúcar y entregar el producto y las vueltas.

El usuario puede en cualquier momento antes de escoger el azúcar cancelar la operación, mediante un botón existente para este objetivo.

En la imagen siguiente se muestra el digrama de casos de uso además se puede ver el código que hemos empleado para conseguirlo.



Ejemplo: Tienda en Internet.

El cliente debe identificarse mediante su dirección de correo. Si es un nuevo cliente se le debe registrar en el sistema previamente, pidiéndole los datos personales.

Una vez identificado al cliente, éste podrá elegir el medio de pago: por transferencia bancaria o con tarjeta de crédito. Según el medio de pago se le solicitarán unos datos u otros.

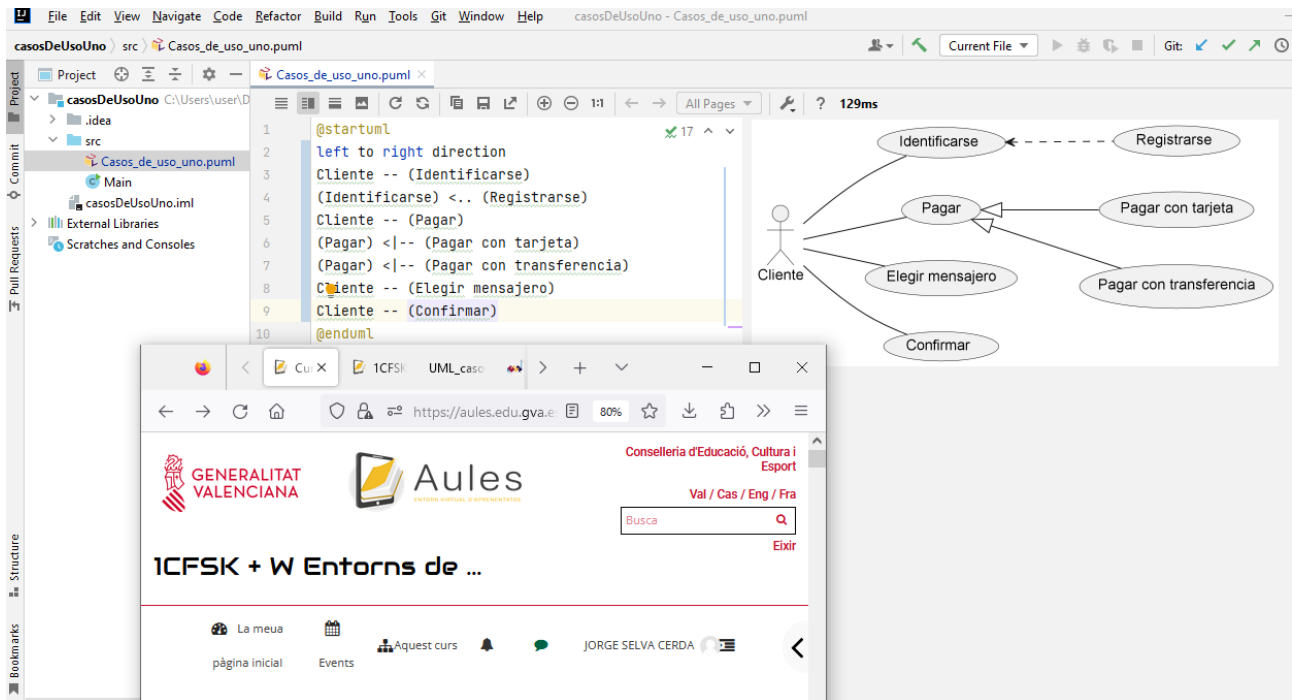
El cliente también deberá elegir el método de envío.

Finalmente se le mostrarán todos los datos del pedido para pedirle que confirme.

En este diagrama que se muestra a continuación se puede también observar la descomposición del caso general “Pagar” en los casos específicos “Pagar con tarjeta” y “Pagar con transferencia” mediante una generalización.

El caso “Registrarse” extiende a “Identificarse” porque está sujeto a la condición “si es un nuevo cliente”.

En la imagen posterior se observa el diagrama de casos de uso así como el código que usamos para conseguirlo en plantUML.



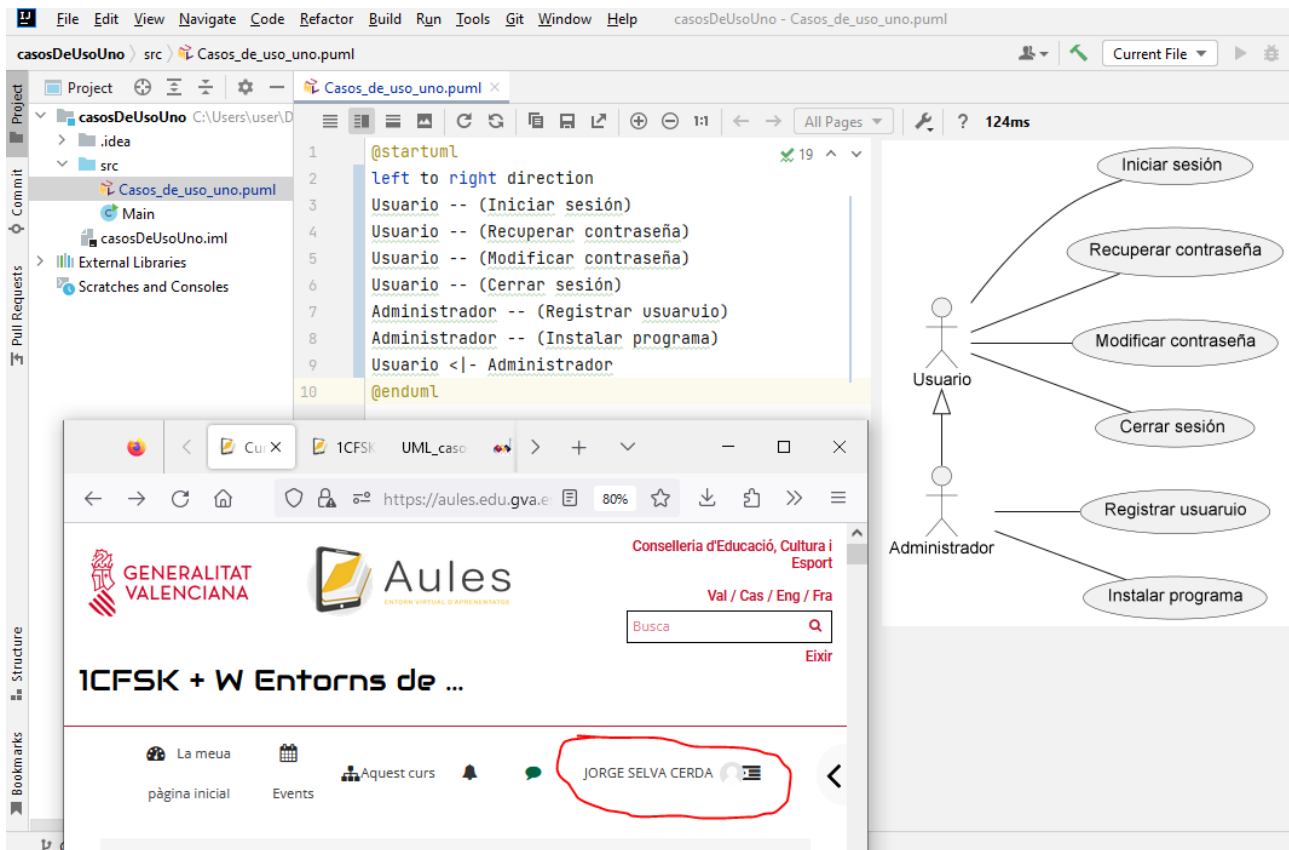
Ejemplo: Usuarios y administradores.

Queremos modelar un sistema en el que hay usuarios. Los usuarios pueden iniciar sesión, modificar su contraseña, recuperar su contraseña y cerrar sesión.

Los administradores tienen los mismos permisos que los usuarios, pero además, pueden registrar usuarios e instalar programas.

En este ejemplo nuevamente se utiliza la generalización, esta vez entre actores, para indicar que los administradores son un tipo específico de usuario.

A continuación se muestra imagen dónde se puede ver tanto el código como el diagrama de uso del caso del usuario y administrativo.

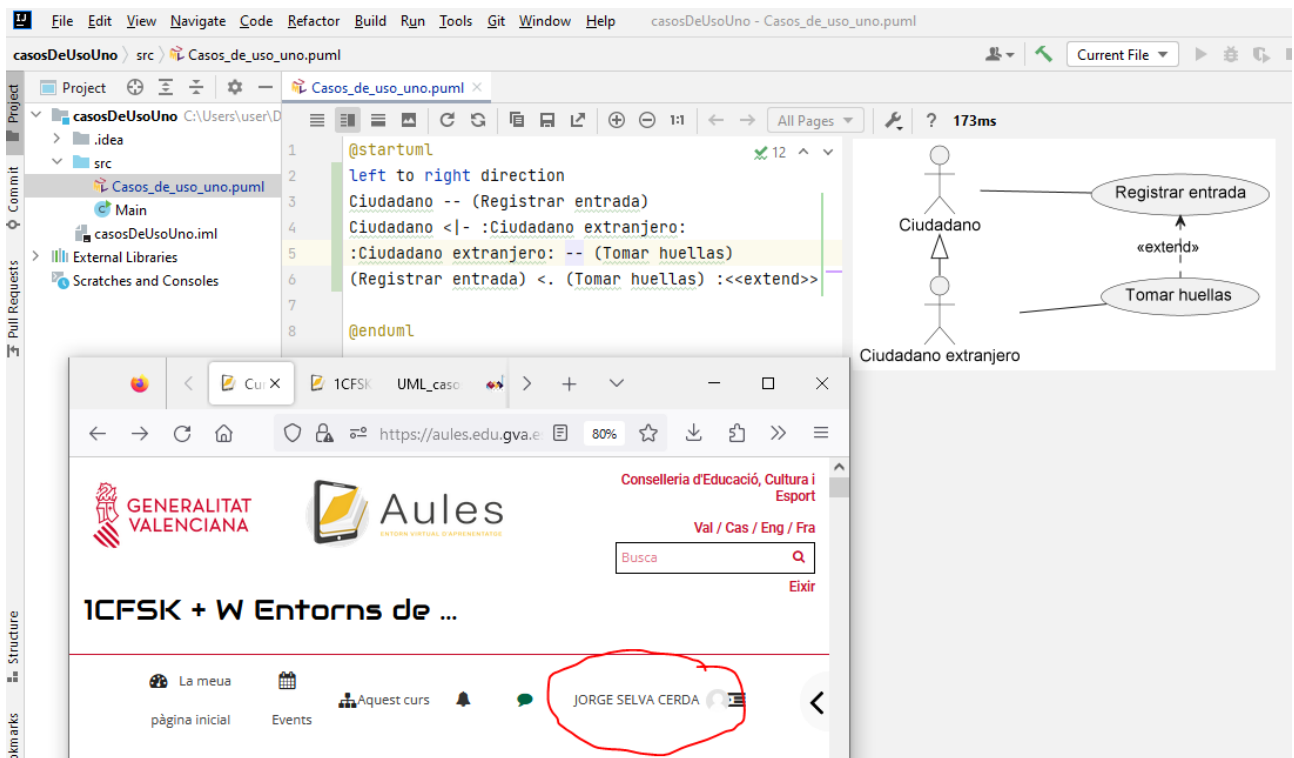


Ejemplo: Puesto fronterizo.

En la frontera de un país se registran todos los ciudadanos que entran. Además, en caso de que el ciudadano sea extranjero, se le toma la huella dactilar.

Nuevamente utilizamos la relación «extend» para indicar que la toma de huellas se realiza como parte del registro de entrada cuando se da la condición de que el ciudadano es extranjero, lo que expresamos añadiendo el actor “ciudadano extranjero” que interactúa con este caso de uso.

Como anteriormente, a continuación mostramos la imagen dónde se puede ver el diagrama de caso de usos de este supuesto en particular y el código empleado para conseguirlo.



### Puntos de extensión:

Los puntos de extensión se utilizan en las relaciones de extensión para indicar en qué punto del caso base se inserta el comportamiento del caso extendido, y bajo qué condición.

A continuación se muestra una imagen con el código y el diagram de usos que nos muestran esta características de las relaciones entre casos.

The image displays a UML modeling environment with a use case diagram and its corresponding UML code. The diagram shows a use case 'Sacar dinero' with the condition '+ solicitud hecha' extending the use case 'Retener tarjeta'. A note attached to the extension relationship specifies the condition: '{ historia sospechosa }' and the extension point: 'solicitud hecha'.

The UML code on the left defines two use cases, UC1 and UC2, with an extension relationship. The code is as follows:

```

62 Ciudadano <|- :Ciudadano extranjero:
63 :Ciudadano extranjero: -- (Tomar huellas)
64 (Registrar entrada) <. (Tomar huellas) : <<extend>>
65 @enduml
66 @startuml
67 left to right direction
68 usecase UC1 as "Sacar dinero
69 ----
70 + solicitud hecha
71 "
72
73 usecase UC2 as "Retener tarjeta
74
75 "
76
77 UC1 <.. UC2 : <<extend>>
78 note "Description\n\n condición:\n {
79 n1 -- UC2
80
81 @enduml

```

The browser window in the background shows the 'Aules' website, which is part of the 'GENERALITAT VALENCIANA' system. The website includes a search bar and a user profile for 'JORGE SELVA CERDA'.

En el caso base (Sacar dinero) hemos añadido un punto de extensión solicitud hecha, que es el momento en el que se inserta el comportamiento de Retener tarjeta, pero sólo cuando se dé la condición historia sospechosa.

En Modelio, los puntos de extensión están en el apartado Nodes →Extension Point. A la flecha le hemos añadido una nota (Common →Note) indicando la condición entre llaves y el punto de extensión.