


The screenshot displays a development environment with IntelliJ IDEA on the left and a web browser on the right. In the IDE, a sequence diagram is being edited, showing an actor and a start/end boundary. The browser window shows the rendered UML diagram, which includes text explaining actors and use cases. The text in the browser is as follows:

actores interactúan con él.

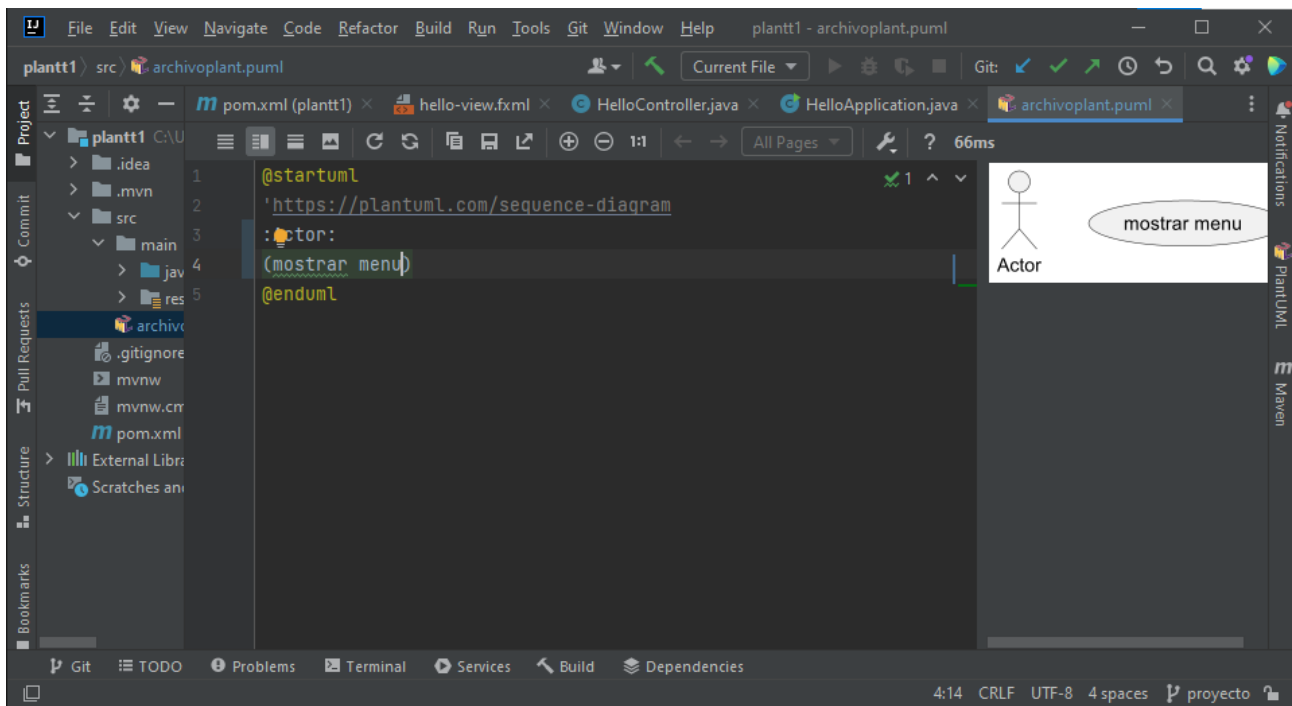
Elementos del diagrama

- **Actor:** Los actores son personas o procesos automáticos que necesitan interactuar con el sistema. Se deben identificar sus papeles en el sistema. En el diagrama, se representan del siguiente modo:

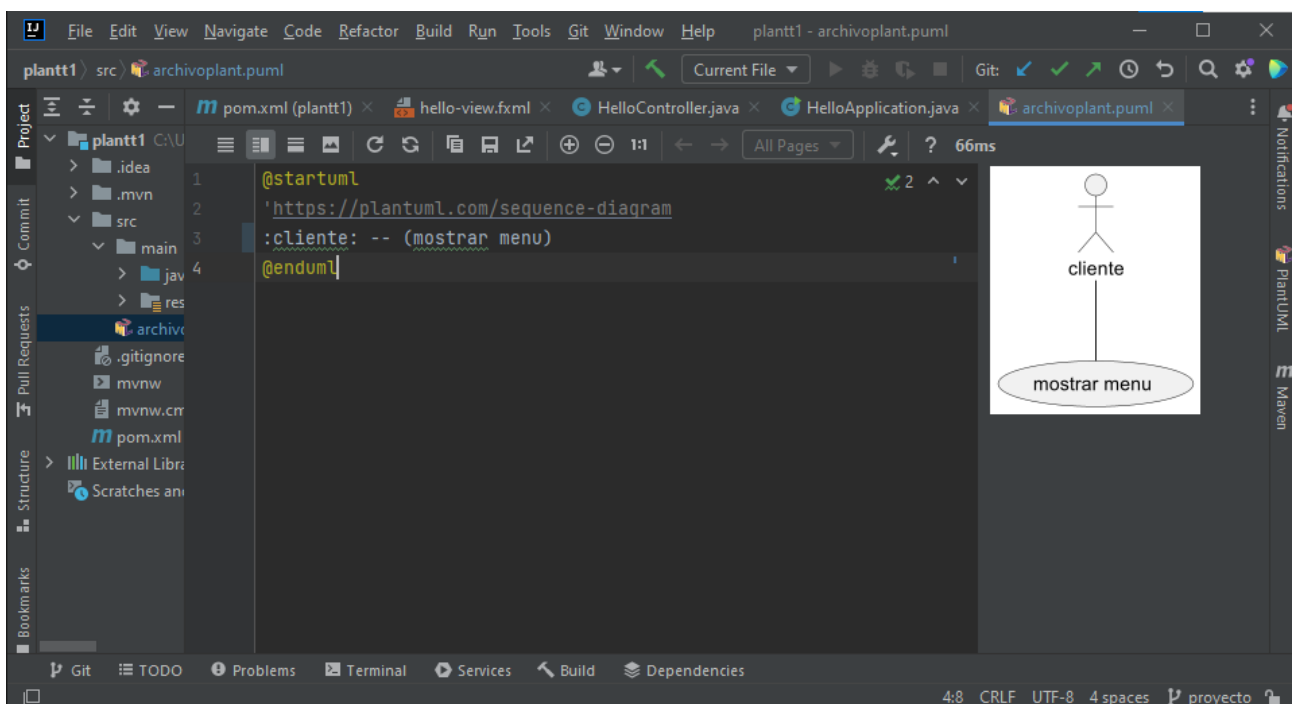


- **Caso de uso:** Los casos de uso se representan mediante elipses y corresponden a acciones generales del sistema. Una forma de reconocerlos es que suelen ser verbos en la descripción del caso de uso.

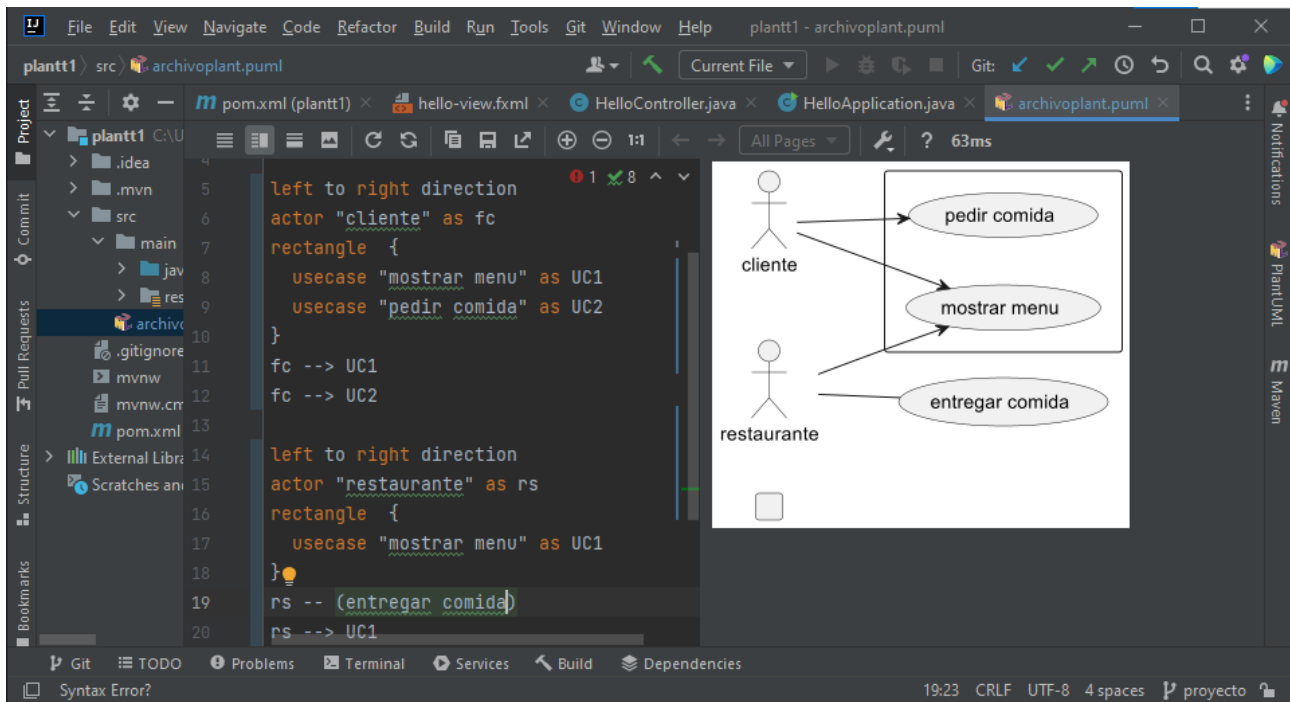
Caso de uso: Los casos de uso se representan mediante elipses y corresponden a acciones generales del sistema. Una forma de reconocerlos es que suelen ser verbos en la descripción del caso de uso.



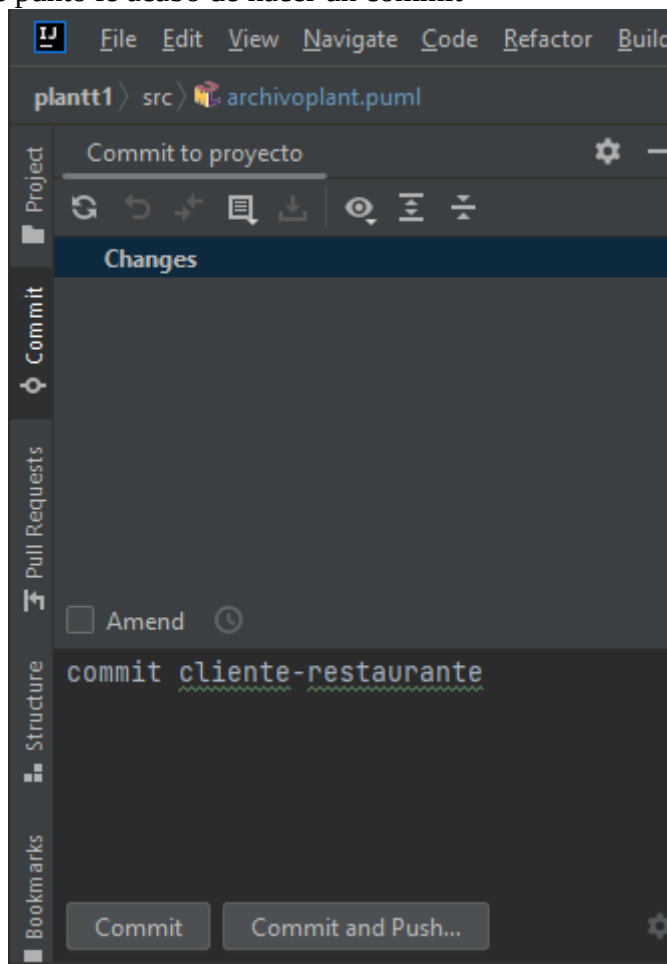
Asociación: La interacción entre los actores y los casos de uso del sistema se representan por una línea recta que une a ambos.

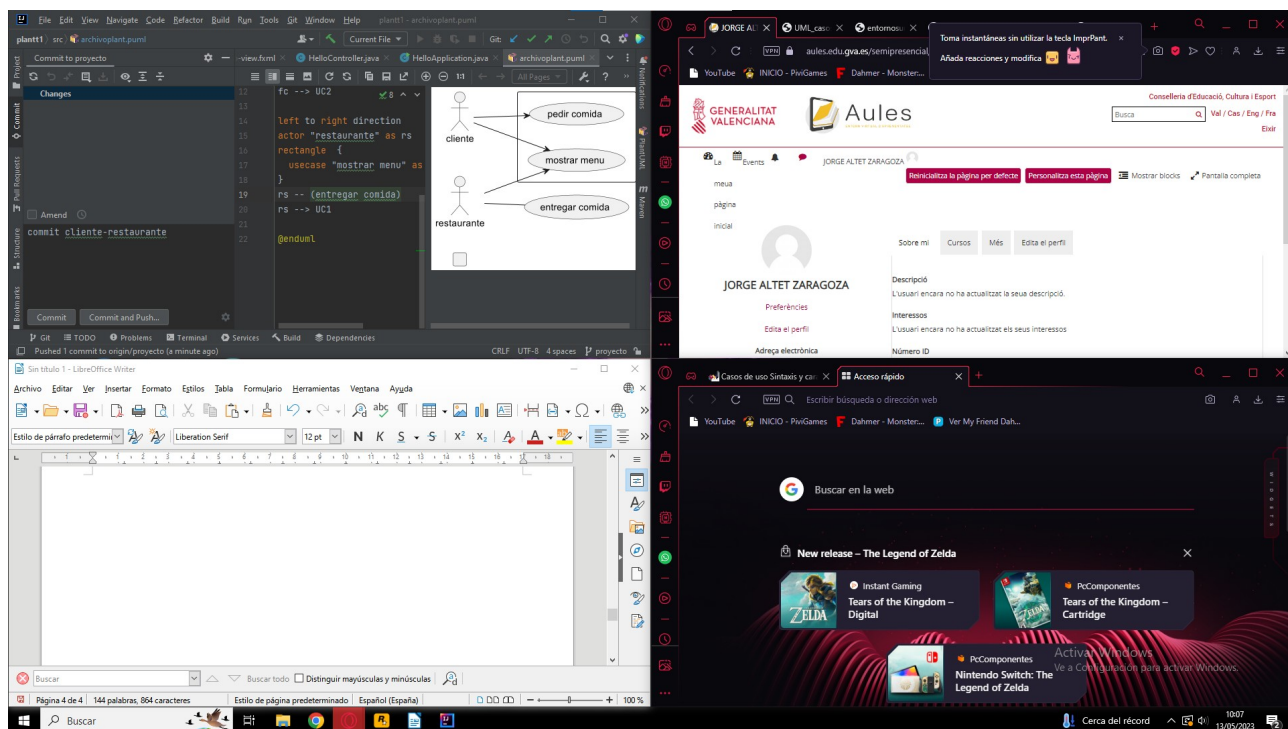


Sistema: El sistema es el software que vamos a desarrollar. Puede ser un pequeño componente cuyos actores son otros componentes, o puede ser una aplicación completa. Se representa como una caja rectangular. Dentro de ella se incluyen los casos de uso soportados por el sistema.

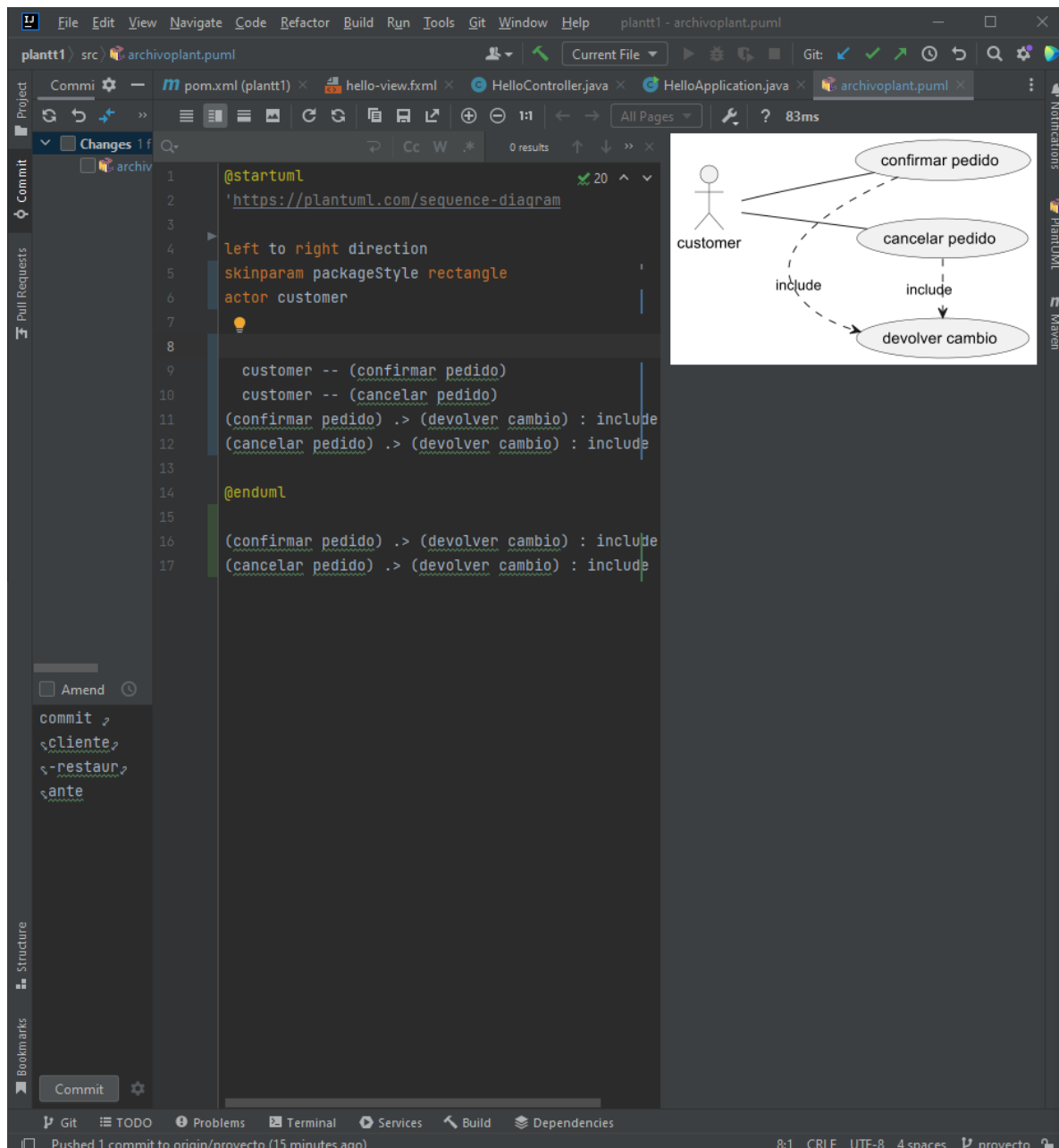


En este punto le acabo de hacer un commit

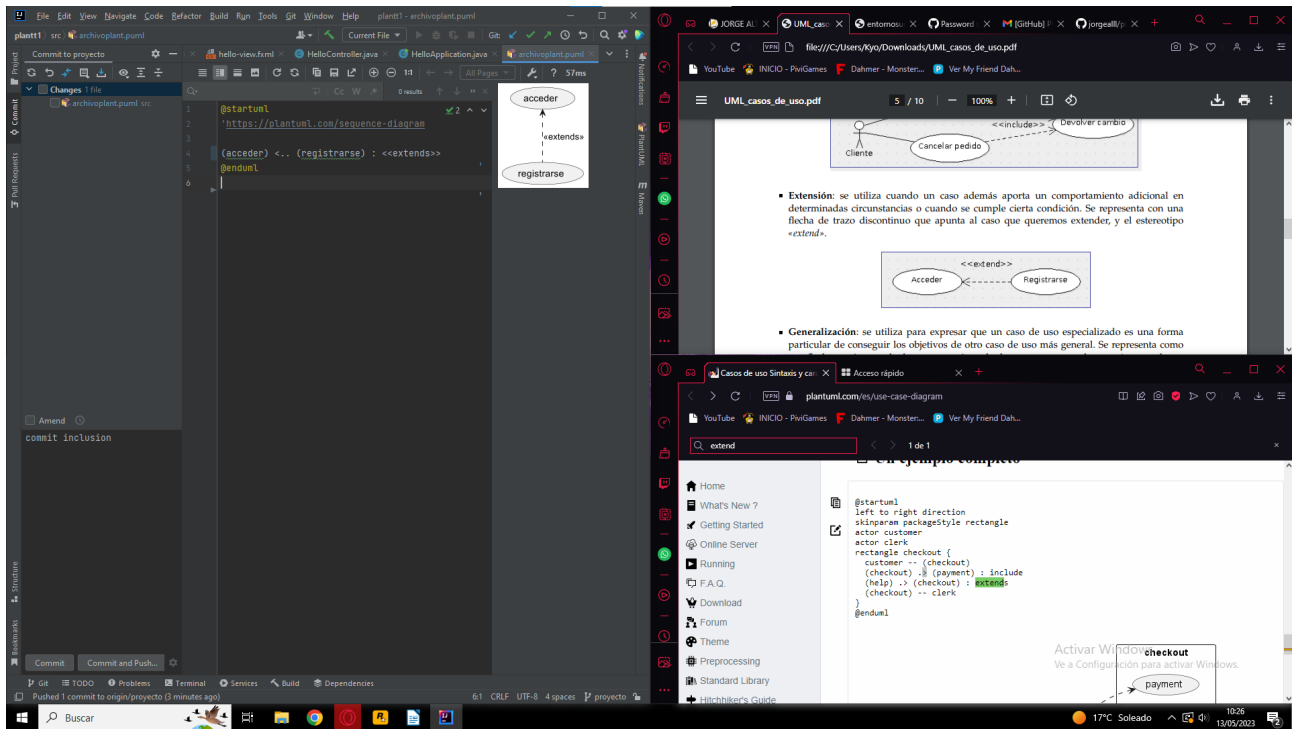




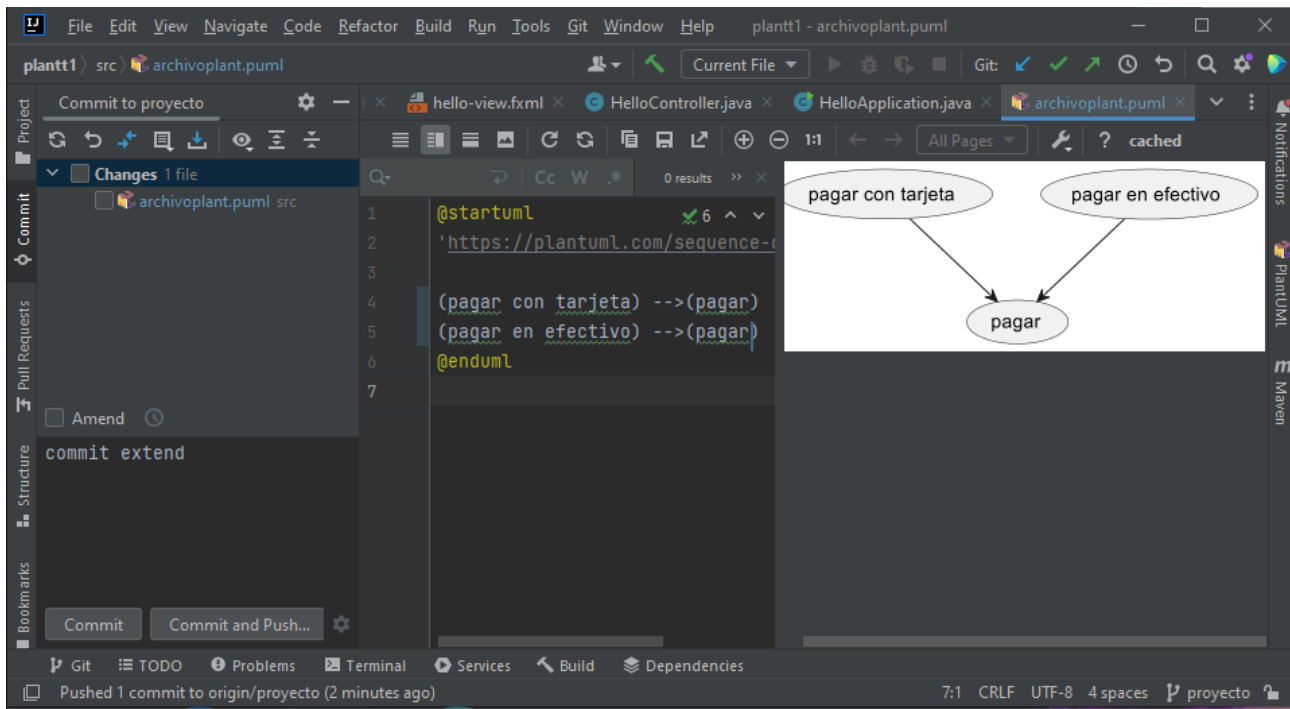
Inclusión: se utiliza cuando el comportamiento de un caso de uso se incluye dentro del comportamiento de otro. Se representa con una flecha de trazo discontinuo desde el caso que incluye hasta el caso incluido, con el estereotipo «include» o «use» Los casos de uso incluidos se pueden compartir, así evitamos repetirlos. También se pueden utilizar para estructurar el diagrama en varios niveles de detalle, pero no conviene abusar de ellos (ver recomendaciones más abajo)



Extensión: se utiliza cuando un caso además aporta un comportamiento adicional en determinadas circunstancias o cuando se cumple cierta condición. Se representa con una flecha de trazo discontinuo que apunta al caso que queremos extender, y el estereotipo «extend»



Generalización: se utiliza para expresar que un caso de uso especializado es una forma particular de conseguir los objetivos de otro caso de uso más general. Se representa como una flecha continua acabada en punta triangular hueca que apunta al caso más general.



¿Cómo distinguir las relaciones entre casos? A veces no está del todo claro cuándo debemos usar la extensión, generalización o inclusión. En caso de duda, podemos guiarnos por las siguientes reglas: La inclusión equivale a “copiar y pegar” un caso de uso dentro de otro. Si imaginamos los casos de uso como líneas de un programa, sería equivalente a incluir el código de un caso dentro del otro. La extensión se utiliza cuando un caso añade funcionalidad a otro dependiendo de alguna condición. Si imaginamos los casos de uso como líneas de un programa, la relación “B extiende a A” (flecha apuntando de B a A) equivaldría a introducir una condición if en algún punto de A, de forma que si se cumple la condición, se ejecuta B antes de proseguir con A. La generalización sirve para indicar que varios casos tienen el mismo comportamiento pero lo llevan a cabo de distinto modo. Equivale a la herencia en programación orientada a objetos, donde las clases hijas pueden rescribir parte del código de la clase padre. Recomendaciones Algunas recomendaciones a la hora de elaborar casos de uso: Javier Martín 2012/2013 6 1 DIAGRAMAS DE CASOS DE USO Tema 8. Diagramas de casos de uso Los casos de uso describen las interacciones más importantes con el sistema, no su funcionamiento interno. Conviene mantener los casos de uso lo más simples posibles. Todos los implicados en el proyecto deben ser capaces de entender el diagrama de casos de uso. Si se complica demasiado, pierde su utilidad. Hay que tener en cuenta que el diagrama de casos de uso se elabora durante la fase de especificación de requisitos y que éstos son cambiantes por naturaleza. Por ello no conviene ahondar demasiado en detalles ni descomponer cada caso en sub-casos, ya que si éstos cambian habremos malgastado el tiempo. En cualquier caso, es preferible acompañar al diagrama de una buena descripción narrativa del caso de uso. 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.

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.

The screenshot shows an IDE window with a UML sequence diagram for a border checkpoint scenario. The diagram is titled "plant1 - archivoplant.puml" and is located in the "src" directory. The diagram shows two actors, "ciudadano" and "ciudadano2", interacting with each other and performing actions. The sequence of events is as follows:

- Actor "ciudadano2" sends a message to actor "ciudadano".
- Actor "ciudadano" performs the action "(registrar entrada)".
- Actor "ciudadano2" performs the action "(tomar huellas)".
- Actor "ciudadano2" extends the action "(tomar huellas)" with the action "(registrar entrada) : extend".

The diagram is visualized in a window titled "75ms". The IDE interface includes a menu bar (File, Edit, View, Navigate, Code, Refactor, Build, Run, Tools, Git, Window, Help), a toolbar, and a sidebar with tabs for "pom.xml (plant1)", "hello-view.fxml", "HelloController.java", "HelloApplication.java", and "archivoplant.puml". The bottom status bar shows "Pushed 1 commit to origin/proyecto (4 minutes ago)", "12:3", "CRLF", "UTF-8", "4 spaces", and "proyecto".

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. Veámoslo con un ejemplo:

este ultimo no se como hacerlo