

## Practica UD 11. Diagramas UML, Clases.

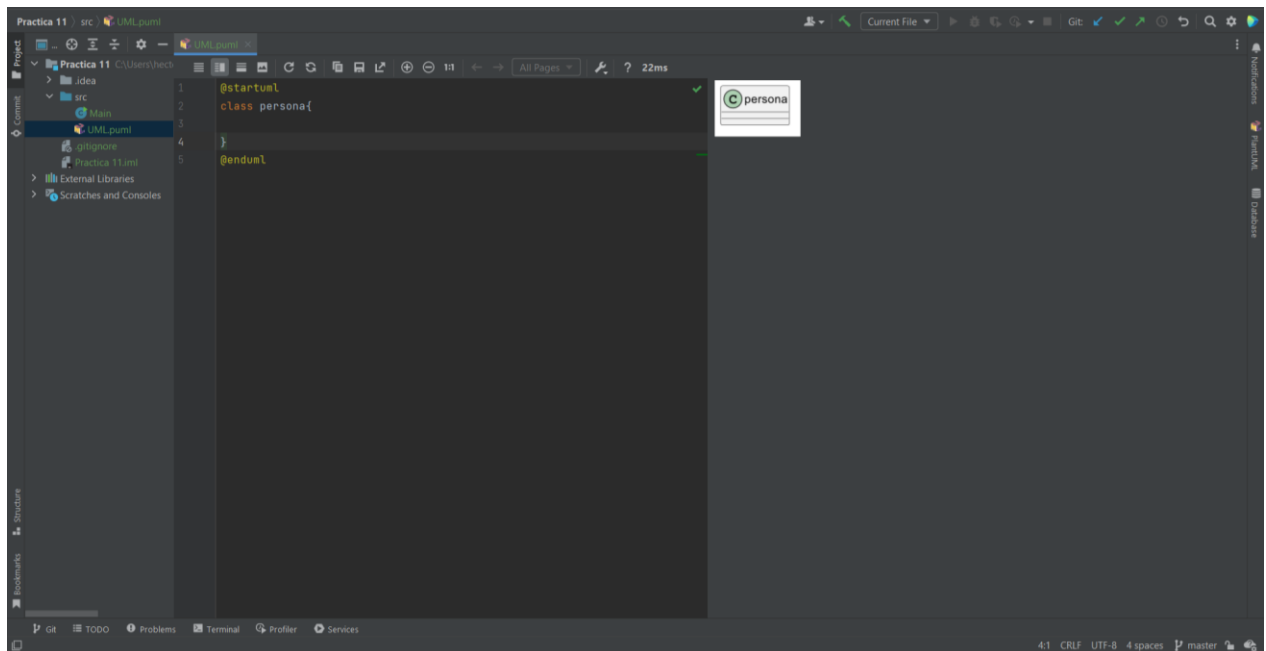
<https://github.com/Hectorprog21/UMLClases-Hectormanuelgelardosabater.git>

### 1) Identificación de las clases

- Miembro (o miembro numerario) (Member)
- Miembro de la junta directiva (BoardMember)
- Evento (Event)
- Conferencia (Conference)
- Reunión de la junta directiva (BoardMeeting)
- Localización (Location)

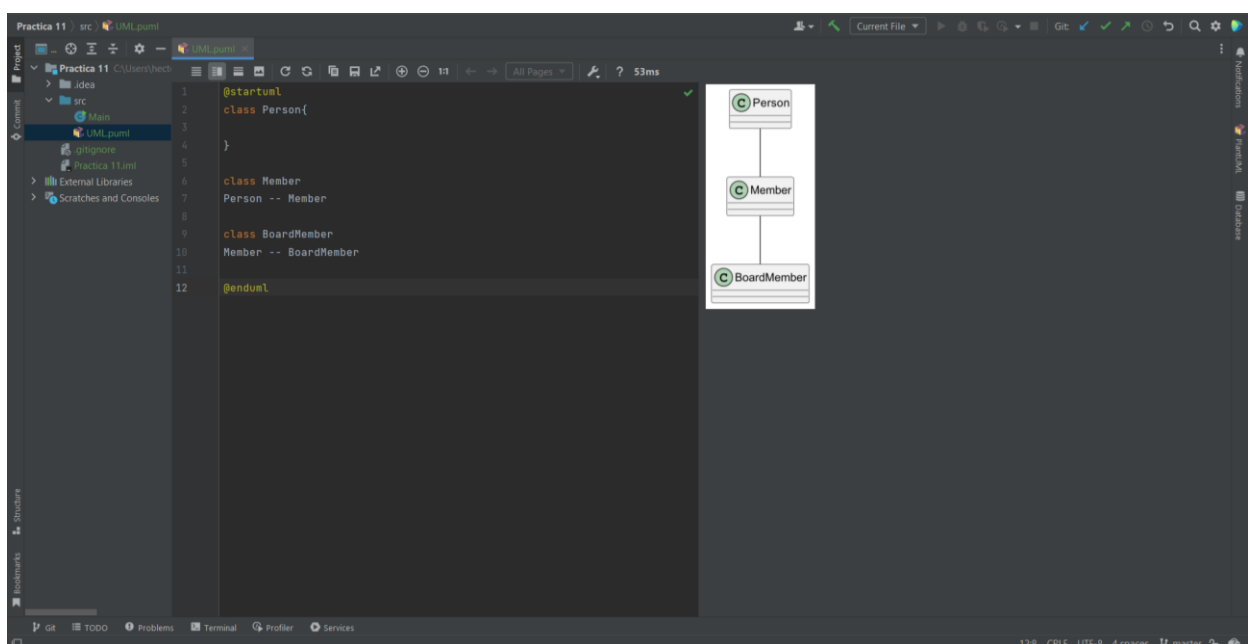
The image shows a composite of two screenshots. On the left is a web page from the 'GENERALITAT VALENCIANA' website, specifically the 'Aules' section. The page title is '2022 1CFSS+1CFSQ Ent ...'. It features a navigation bar with links like 'La meua pàgina inicial', 'Events', 'Aquest curs', and 'HECTOR MANUEL GELARDO SABATER'. Below the navigation bar, there's a breadcrumb trail: 'La meua pàgina inicial > Els meus cursos > ENTORNO DE DESARROLLO > UD 11 Documentación y Diagramas'. The main heading is 'Práctica Diagramas UML Clases.' followed by a paragraph of text in Spanish explaining the purpose of the practice and mentioning a PDF example and a video tutorial for installing PlantUML. On the right is a screenshot of an IDE (IntelliJ IDEA) showing a project named 'Practica 11'. The project structure includes folders for 'idea', 'src', 'Main', 'UML.puml', 'gitignore', 'Practica 11.uml', 'External Libraries', and 'Scratches and Consoles'. The 'UML.puml' file is open, displaying a sequence diagram with participants 'Alice', 'Bob', and 'autonumb'. The diagram shows a sequence of messages: '1 Authentication', '2 Authentication', '3 Another authentication', and '4 another authentication'. The IDE interface also shows a terminal, a TODO list, and a status bar at the bottom.

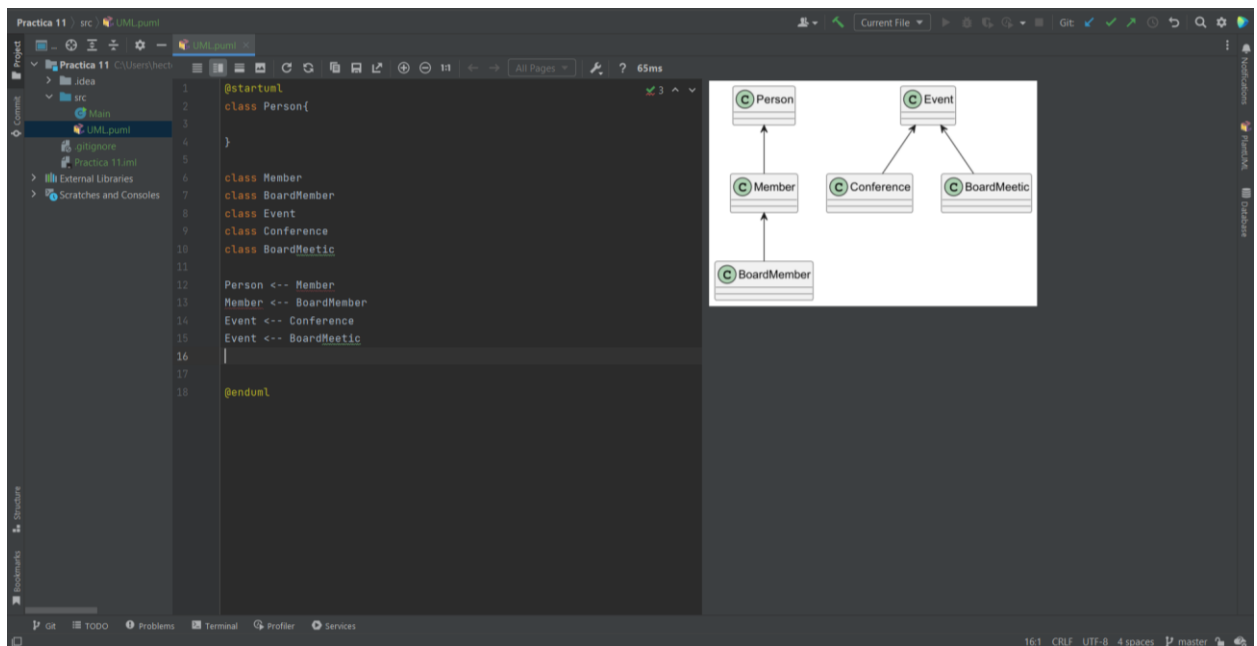
Adicionalmente, se ha añadido la clase Persona (Person) para poder identificar también a los conferenciantes, ya que podría darse el caso de que éstos no fueran miembros de la asociación.



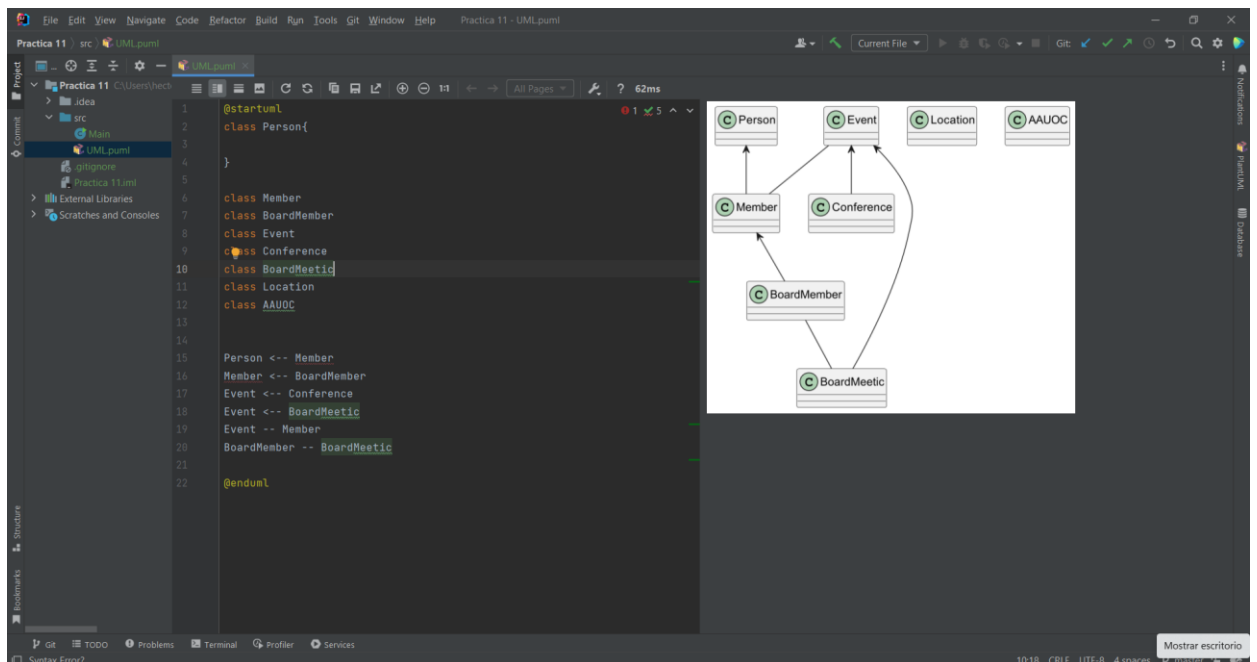
## 2) Creación del modelo de datos

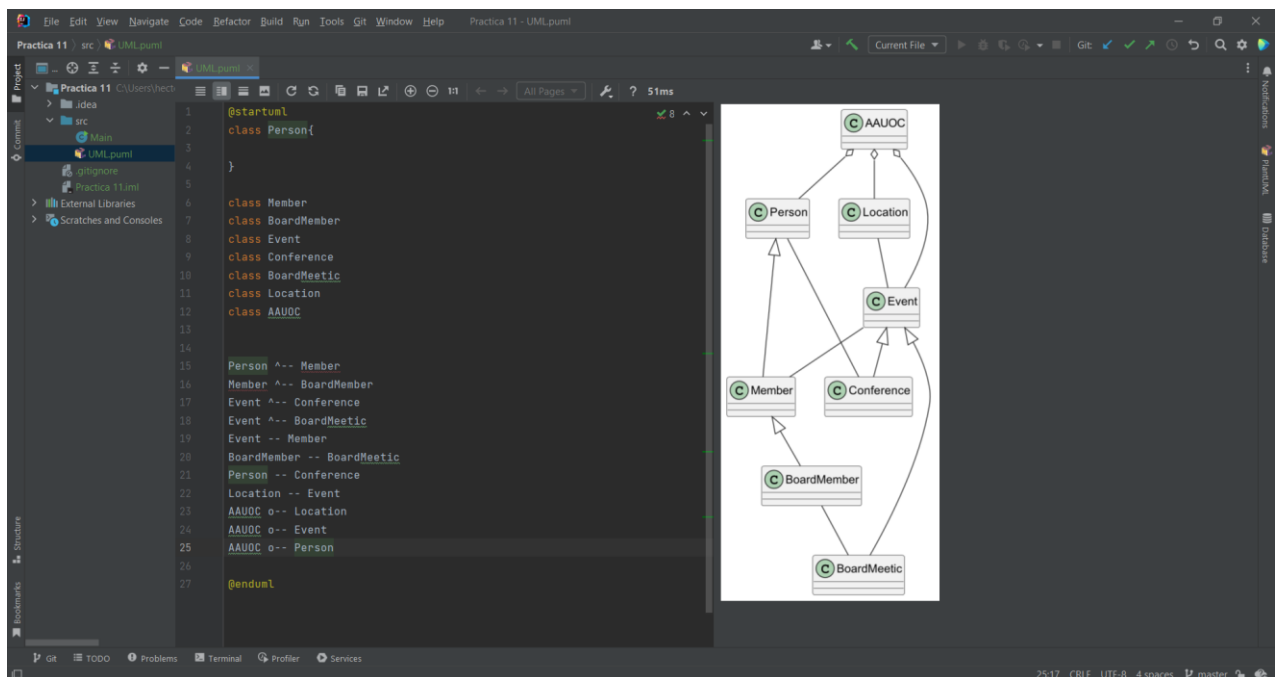
Para crear el modelo de datos, se ha detectado la existencia de una jerarquía de herencia cuya superclase es el evento y, según la descripción del problema, tiene únicamente dos subclases, que son las conferencias y las reuniones de la junta directiva. En este problema, se ha descartado la inclusión de una clase que represente a los eventos con restricciones en el número de asistentes. En caso de ampliarse la tipología de eventos, se debería considerar dicho punto.





Además, tenemos las clases Localización (Location) y Asociación (AAUOC), que se relacionan con el resto de clases del siguiente modo:

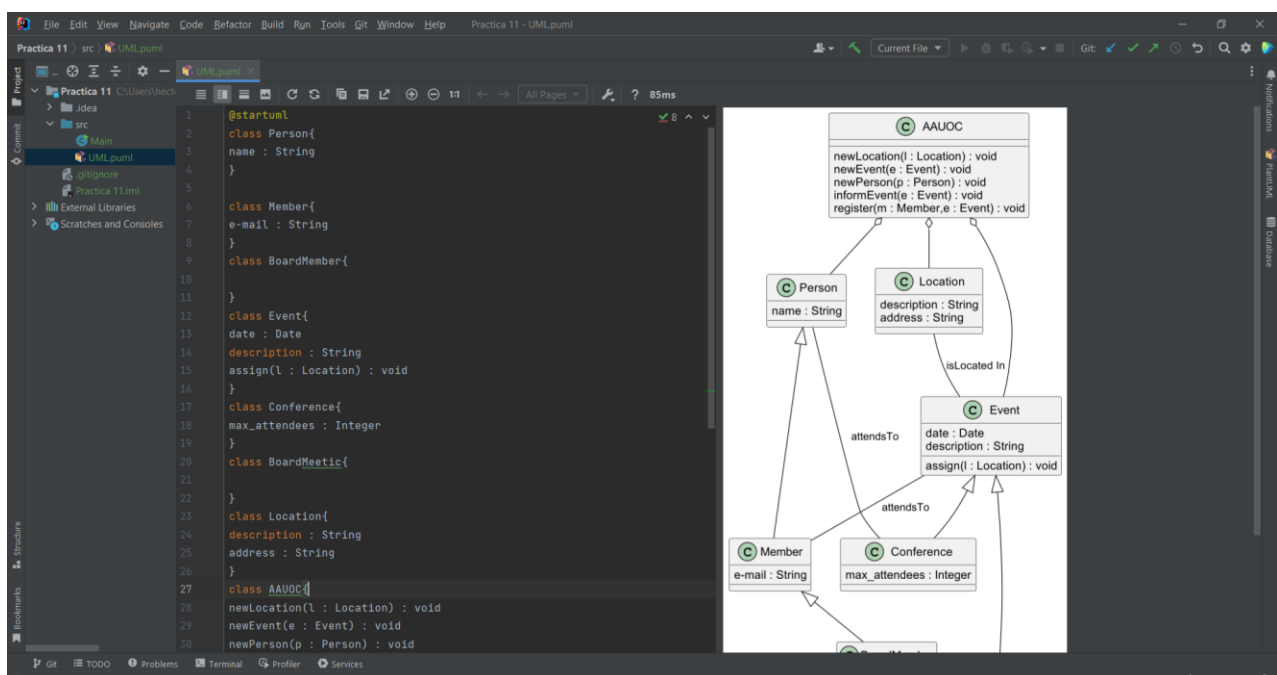




### 3) Inclusión de atributos y métodos:

Una vez creado el modelo de datos, completamos las clases con sus atributos y los métodos más relevantes (quedan fuera de este diagrama los métodos getters y setters, así como los métodos constructores de cada clase).

La asociación necesitará un conjunto de métodos para añadir nuevos eventos, personas y localizaciones al sistema (métodos newX de la clase AAUOC), así como también un método para informar a los miembros de la convocatoria de un evento (método informEvent). Al mismo tiempo, se dice que los usuarios necesitarán confirmar la asistencia a los eventos (método register, que deberá almacenar los asistentes por orden y controlar el número máximo de éstos si fuera necesario).



#### 4) Inclusión de la cardinalidad y navegabilidad de las relaciones:

En el siguiente diagrama, se han eliminado los métodos e incluido las navegabilidades (en este caso, todas son bidireccionales, debido a que no se nos ha comunicado ningún tipo de relación y/o acceso a la información de una clase a otra), y las cardinalidades de dichas relaciones.

