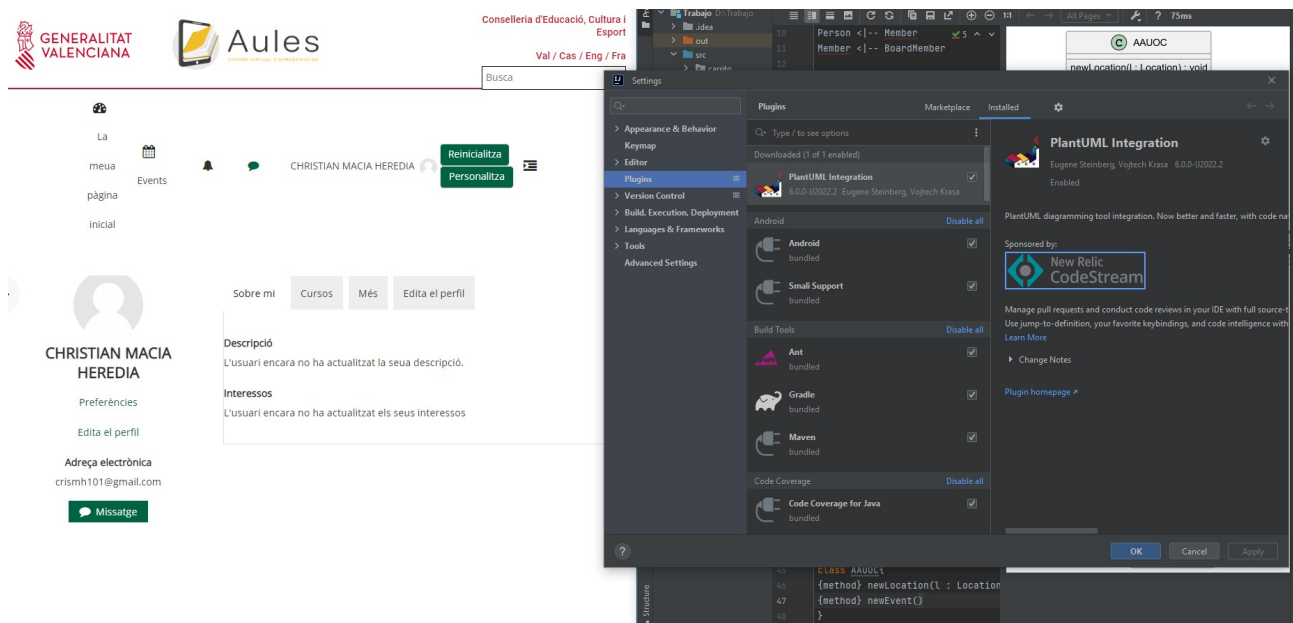


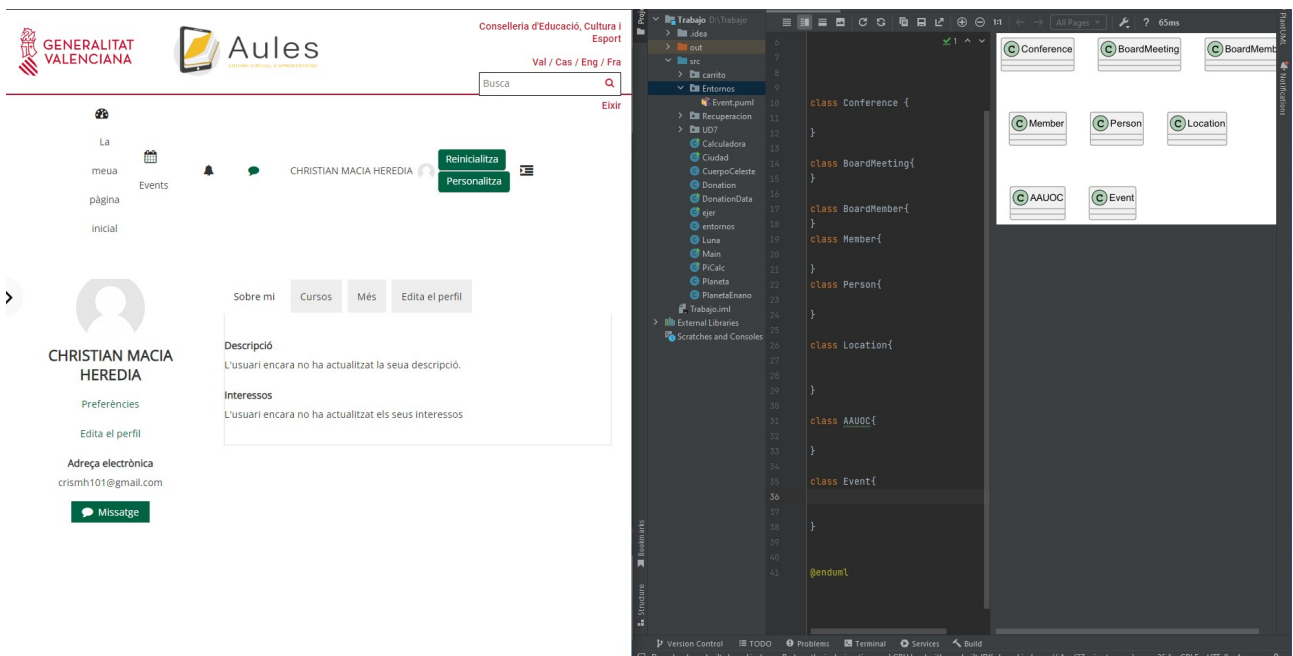
Diagramas UML

Instalamos PlantUML en IntelliJ, para ello iremos File, Settings y en el apartado de plugins buscamos PlantUML Integration y lo descargamos. Después nos pedirá que reiniciemos IntelliJ.



Creamos un nuevo PlantUML File dándole click derecho a la carpeta que queramos realizar el proyecto y en el apartado donde pone PlantUML File le hacemos click.

Dentro del nuevo archivo UML creamos las clases que vamos a necesitar.



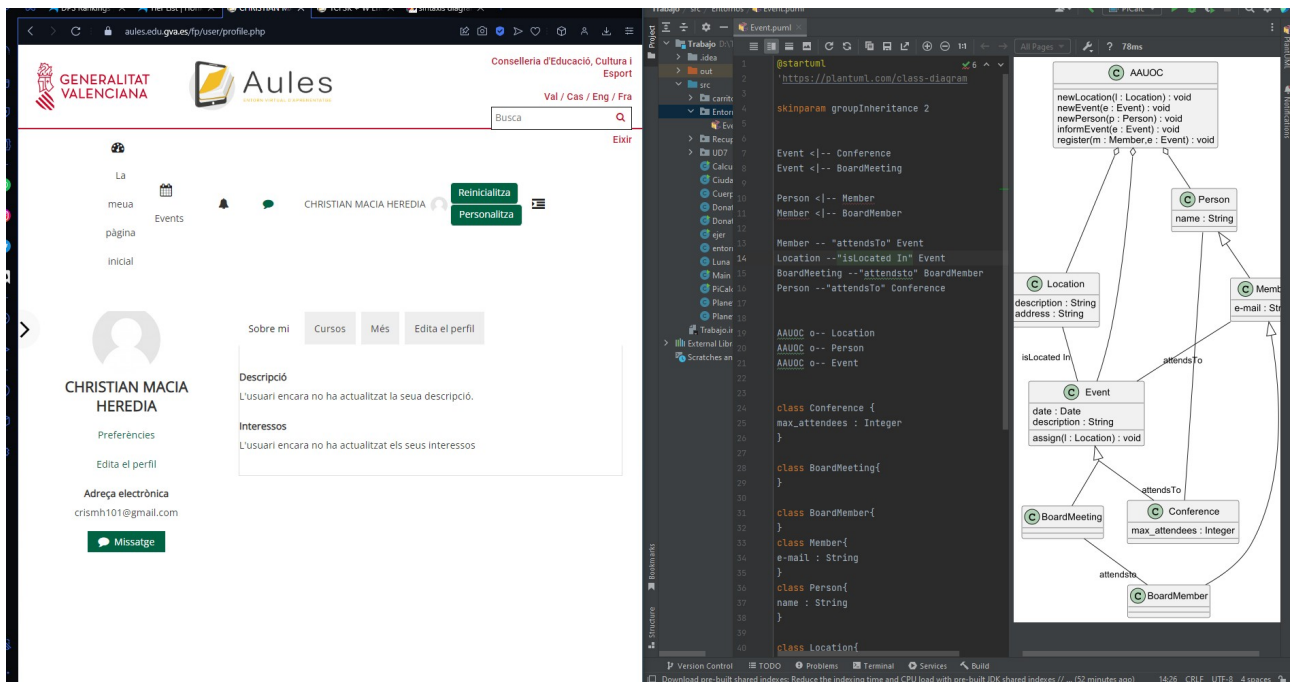
Ahora introduciremos las relaciones entre estas clases, pero si tener en cuenta las cardinalidades.

The screenshot displays the Aules web application on the left and an IDE on the right. The web application shows a user profile for Christian Macia Heredia with options to reset or personalize the profile. The IDE shows a class diagram with classes AAUDC, Location, Event, BoardMeeting, Person, Member, and BoardMember. The diagram illustrates relationships between these classes, including inheritance and associations. The code in the IDE shows the implementation of these classes, including methods like newLocation(), newEvent(), newPerson(), informEvent(), and register().

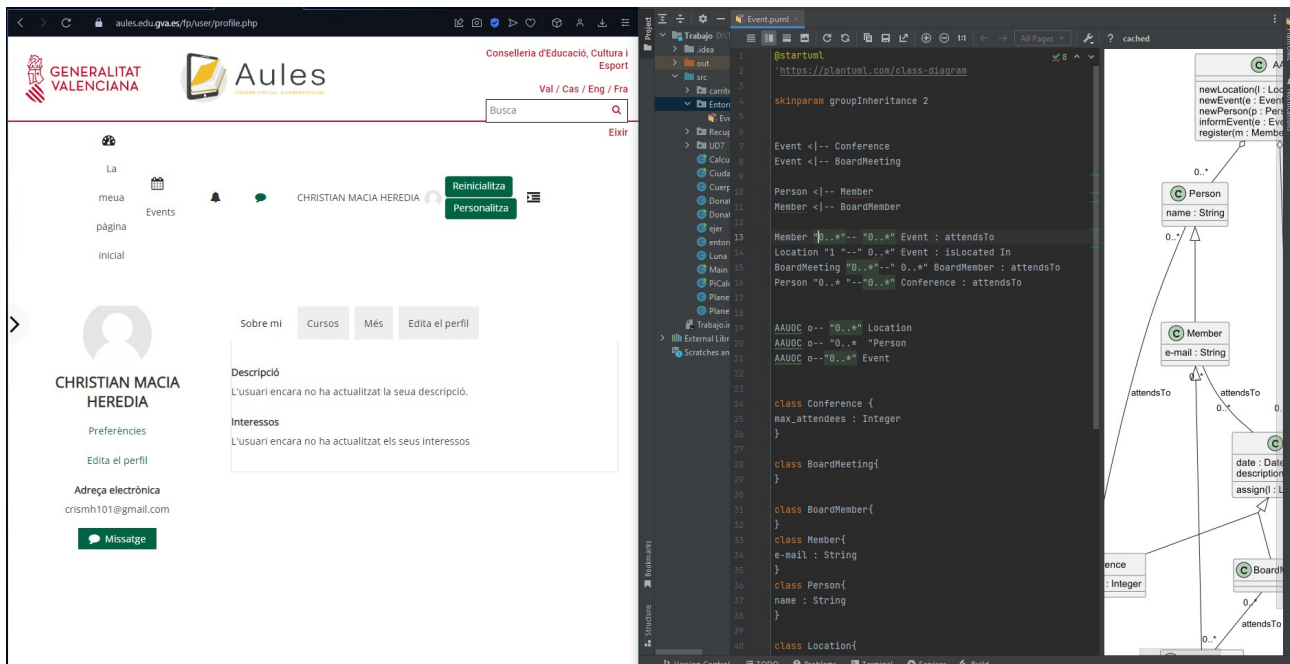
Introducimos los atributos y sus métodos en cada clase de esta forma. Utilizamos method para cambiar al espacio reservado para métodos en la clase.

This screenshot shows the Aules web application and an IDE with updated class diagrams and code. The web application remains the same. The IDE's class diagram now includes attributes like max_attendees for Event and BoardMeeting, and email for Person and Member. The code in the IDE shows the implementation of these attributes and methods, including newLocation(), newEvent(), newPerson(), informEvent(), and register(). The IDE also shows the implementation of the newLocation() method in the AAUDC class.

Nombramos las relaciones entre clases ,poniendo entre comillas el nombre en la relación.



Y por último introducimos las cardinalidades entre las clases.



Tiene que quedar así.

The image displays a web application interface on the left and its corresponding UML class diagram on the right.

Web Application Interface (Left):

- Header: Conselleria d'Educació, Cultura i Esport. Val / Cas / Eng / Fra. Busca. Eixir.
- User Profile: CHRISTIAN MACIA HEREDIA. Buttons: Reinicialitza, Personalitza.
- Navigation: La meua, Events, pàgina, Inicial.
- Profile Section: Sobre mi, Cursos, Més, Edita el perfil.
- Description: Descripció. L'usuari encara no ha actualitzat la seua descripció.
- Interests: Interessos. L'usuari encara no ha actualitzat els seus interessos.
- Contact: Adreça electrònica: crismh101@gmail.com. Missatge.

UML Class Diagram (Right):

- AAUOC** (Class): newLocation(l: Location): void, newEvent(e: Event): void, newPerson(p: Person): void, informEvent(e: Event): void, register(m: Member, e: Event): void.
- Person** (Class): name: String.
- Member** (Class): e-mail: String.
- Location** (Class): description: String, address: String.
- Event** (Class): date: Date, description: String, assign(l: Location): void.
- Conference** (Class): max_attendees: Integer.
- BoardMeeting** (Class).

Associations:

- AAUOC to Person (0..* to 0..*).
- AAUOC to Location (0..* to 0..*).
- AAUOC to Event (0..* to 0..*).
- Person to Member (0..* to 0..*).
- Member to Event (0..* to 0..*).
- Event to Conference (0..* to 0..*).
- Event to BoardMeeting (0..* to 0..*).
- Location to BoardMeeting (1 to 0..*).

Generalization:

- Member is a generalization of Person.
- Conference is a generalization of Event.
- BoardMeeting is a generalization of Event.

Code Snippets (Middle):

```
class C
max_att
}

class B
}

class B
}

class M
e-mail
}

class P
name :
}

class L
```