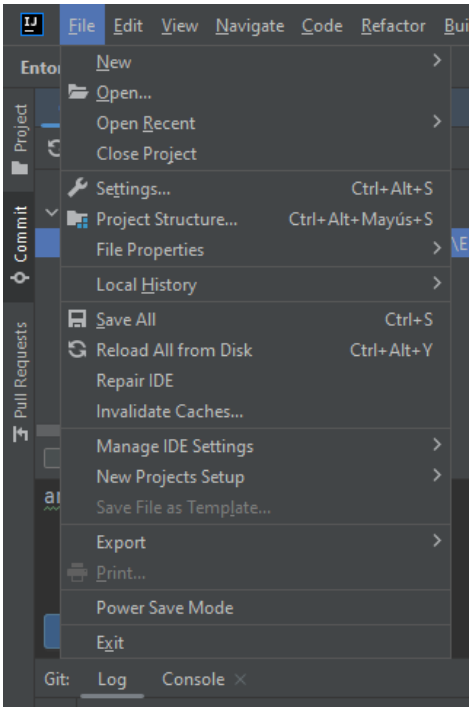
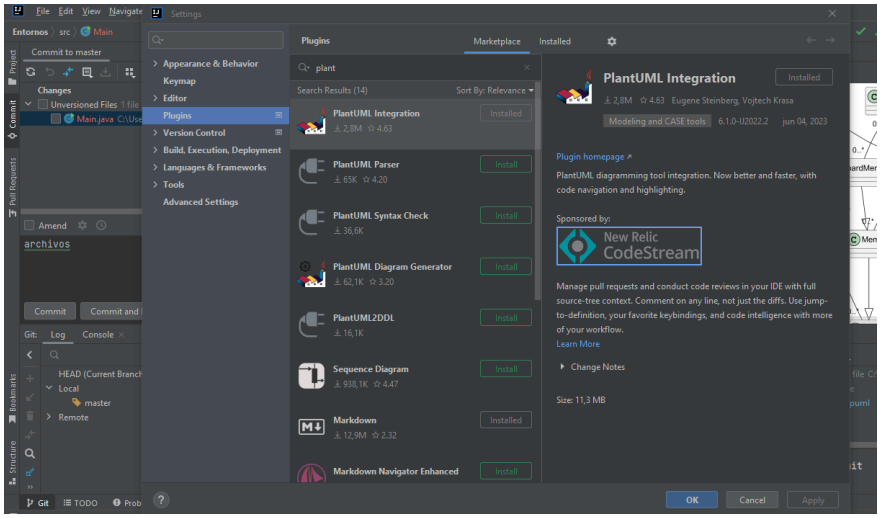
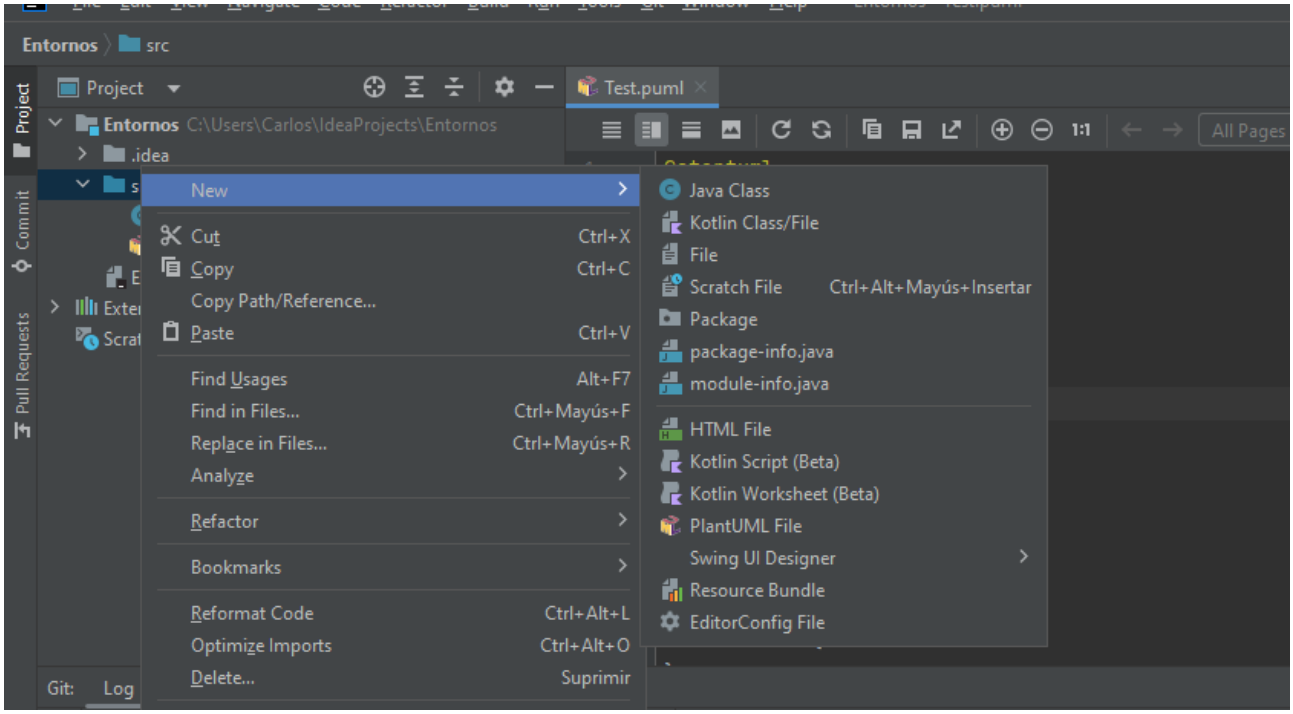
Práctica Diagramas UML Clases.

Lo primero que tendremos que hacer es instalar el plugin de PlantUML en intellij, para ello iremos a file → Settings → plugins → buscamos PlantUML → instalamos y reinciamos Intellij

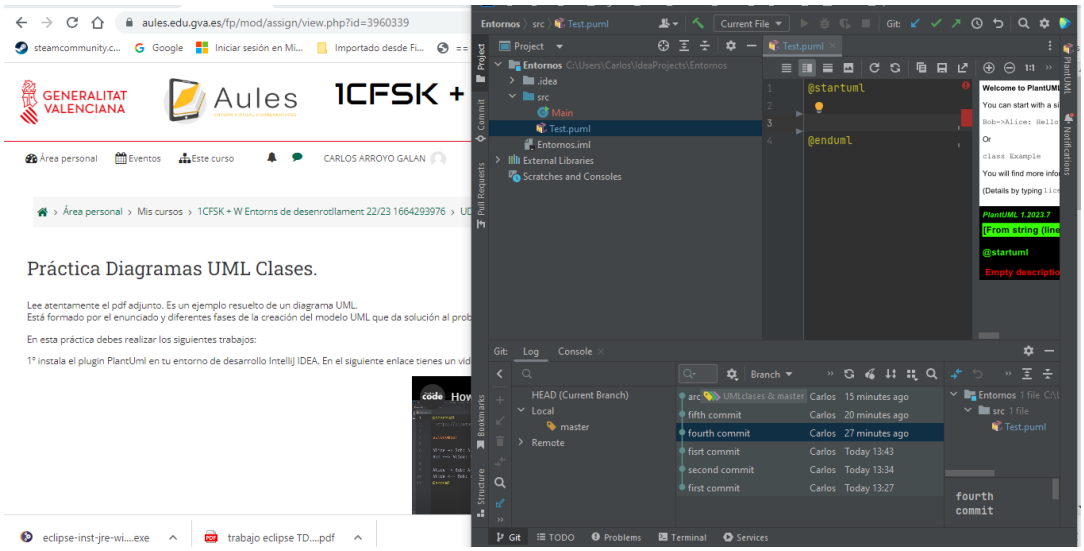




Una vez instalado PlantUML y reinciado Intellij crearemos nuestro UML File, para esto haremos click derecho encima de nuestro src → new → UML file



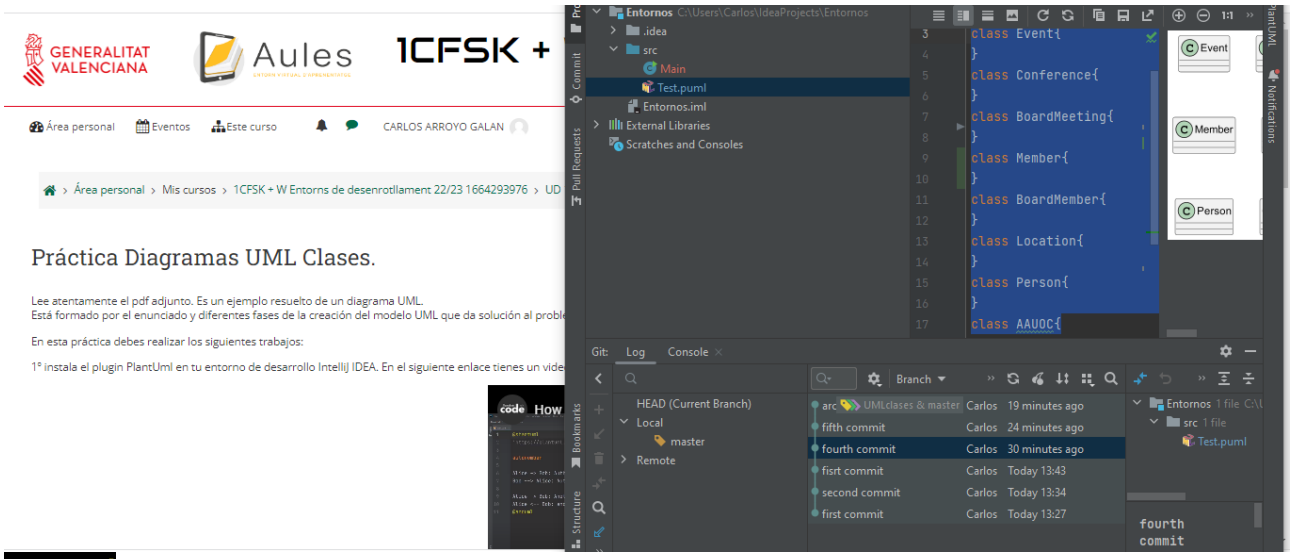
Una vez creado lo tendremos tal que así y podremos iniciar a hacer la practica



Iniciaremos añadiendo todas las clases que nos dicta el enunciado, siendo estas:

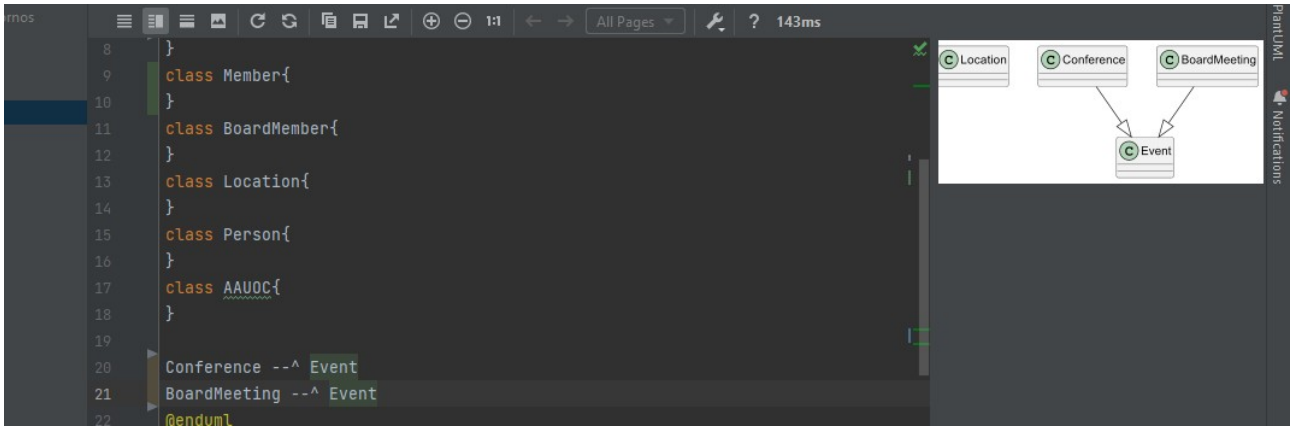
Event, Conference, BoardMeeting, Member, BoardMember, Location, Person, AAUOC

y se implementarían en UML de la siguiente forma:



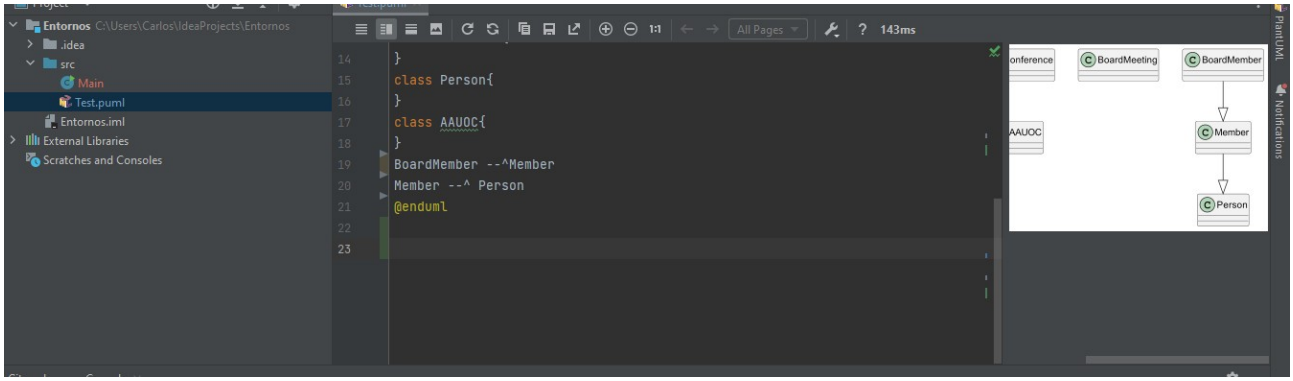

```
@startuml
class Event{
}
class Conference{
}
class BoardMeeting{
}
class Member{
}
class BoardMember{
}
class Location{
}
class Person{
}
class AAUOC{
}
@enduml
```

Para hacer relaciones entra estas mostraré el siguiente ejemplo



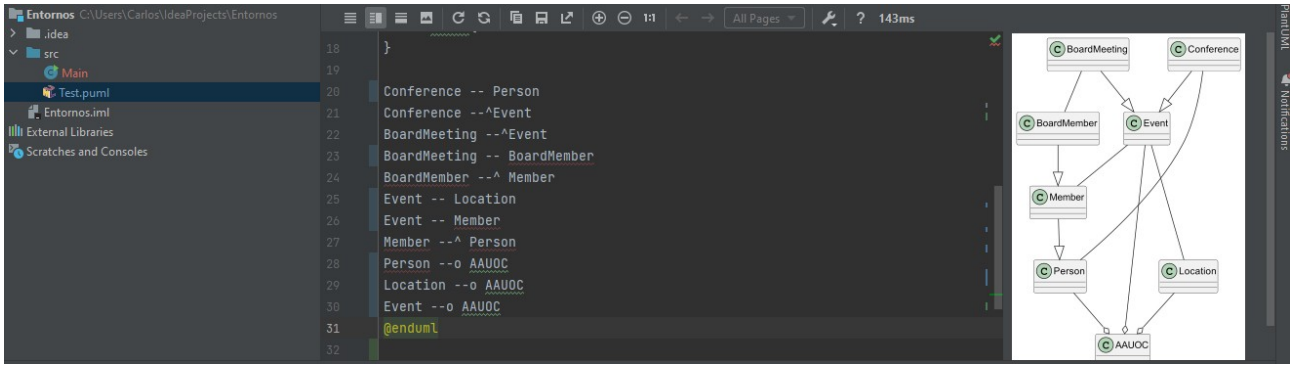
Con
Conference --^ Event
BoardMeeting --^ Event
Podemos apreciar que Conference y BoardMeeting son extensiones de Event, esto aclarado gracias a --^ simbolo en cual nos facilita poder distinguirlos

A continuación mostraré otros distintos ejemplos como por ejemplo el siguiente



En este vemos como
BoardMember --^Member
Member --^ Person
Son extensiones unas de otras hasta Person, seguimos usando el simbolo ^ para distinguirlos

Tercer Ejemplo

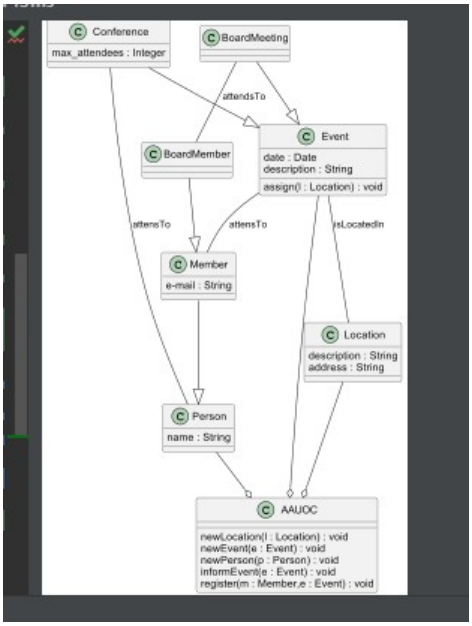



```
@startuml

class Event{
}
class Conference{
}
class BoardMeeting{
}
class Member{
}
class BoardMember{
}
class Location{
}
class Person{
}
class AAUOC{
}

Conference -- Person
Conference --^Event
BoardMeeting --^Event
BoardMeeting -- BoardMember
BoardMember --^ Member
Event -- Location
Event -- Member
Member --^ Person
Person --o AAUOC
Location --o AAUOC
Event --o AAUOC
@enduml
```

En este podemos ver diferentes tipos de relaciones como por ejemplo una relacion normal –y una relación de agreación (--o)

A continuación se mostrará un ejemplo de la implementación de atributos y metodos tanto en atributos como en relaciones

Cuarto Ejemplo




```
@startuml

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

Conference -- Person : attensTo
Conference --^Event
BoardMeeting --^Event
BoardMeeting -- BoardMember : attendsTo
BoardMember --^ Member
Event -- Location : isLocatedIn
Event -- Member : attensTo
Member --^ Person
Person --o AAUOC
```

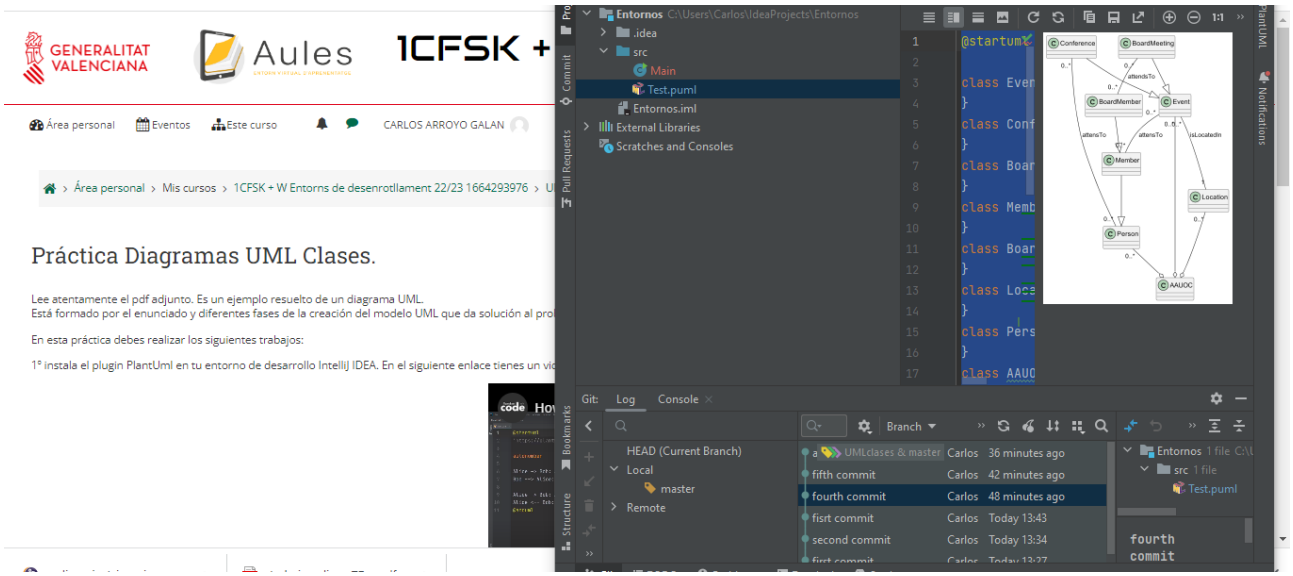
Location --o AAUOC
Event --o AAUOC
@enduml

En este observamos como se añaden atributos y metodos, cogiendo de ejemplo la case eventos
class Event{
 date : Date
 description : String
 assign(l : Location) : void
}
Observamos la implementación de tanto atributos como metodos dentro de una clase, en los atributos distinguiendo el nombre del tipo de atributo con : y en los metodos nombrando todo el método dentro de este.

Tambien podemos observar las relaciones dentro de clases cogiendo de ejemplo
BoardMeeting -- BoardMember : attendsTo
Hacemos que los atributos se integren en las relaciones separandolos con :

Por ultimo veremos un ejemplo de navegabilidad en las relaciones

Quinto Ejemplo



@startuml

class Event{
}
class Conference{
}
class BoardMeeting{
}
class Member{
}
class BoardMember{
}
class Location{
}
class Person{
}
class AAUOC{
}

```
Conference "0..*"--"0..*" Person : attensTo
Conference --^Event
BoardMeeting --^Event
BoardMeeting "0..*"-- "0..*"BoardMember : attendsTo
BoardMember --^ Member
Event "0..*"--"1" Location : isLocatedIn
Event "0..*"--"0..*" Member : attensTo
Member --^ Person
Person "0..*"--o AAUOC
Location "0..*" --o AAUOC
Event "0..*"--o AAUOC
@enduml
```

Observamos que ejemplos de navegabilidad se integran junto antes de definir que tipo de relacion será, por ejemplo en la siguiente

```
Conference "0..*"--"0..*" Person : attensTo
```

Vemos que las clases dos tendran relacion de cero a muchos

Carlos Arroyo Galán DAM1K