

Para gestionar el desarrollo del proyecto en grupo en IISSI-1 es necesaria una base de datos. A continuación se detallan los objetivos y requisitos (**RI**: Requisitos de Información, **RF**: Requisitos Funcionales y **RN**: Reglas de Negocio).

**Obj**

Como: Coordinador de IISSI-1

Necesito: Gestionar la composición de grupos de proyecto y sus calificaciones.

Para: cumplimentar las actas de junio, setiembre y diciembre

**Obj-1**

Como: Coordinador de IISSI-1

Necesito: Configurar la composición de grupos formados por alumnos

Para: Conocer la composición de grupos, necesaria para evaluar proyectos.

**Obj-2**

Como: Coordinador de IISSI-1

Necesito: Gestionar las calificaciones de defensas, por curso, en junio, setiembre y diciembre

Para: Rellenar actas de la asignatura, pues sólo se guardan aprobados hasta diciembre del curso académico vigente.

**RI-1.1**

Como: Profesor de IISSI-1

Necesito: Formar grupos en cualquiera de las convocatorias, compuestos por entre uno y cuatro alumnos. Un alumno podría, por razones diversas, integrarse en otro grupo en una nueva convocatoria. En cualquier caso se conocerá la fecha y hora de su incorporación al grupo y la de su fecha de baja en el mismo.

Para: Facilitar las entregas de documentación y realizar las defensas.

**RF-2.1**

Como: Profesor de IISSI-1

Necesito: Calificar las defensas en cada convocatoria, conociendo día y hora de su realización (fecha de examen). Por defecto, la calificación del alumno en esa convocatoria es la otorgada al proyecto, pero debe poder modificarse la nota individual del alumno asociada a ese proyecto y en esa convocatoria.

Para: facilitar las calificaciones a grupos e individuales en cada convocatoria.

**RN-01**

Como: Coordinador de IISSI-1

Necesito: Que un alumno no pueda defender más que un proyecto en cada convocatoria, teniendo en cuenta sus posibles pertenencias a distintos grupos en las convocatorias en que se guarda la nota.

Para: Evitar inconsistencias en las calificaciones.

**RN-02**

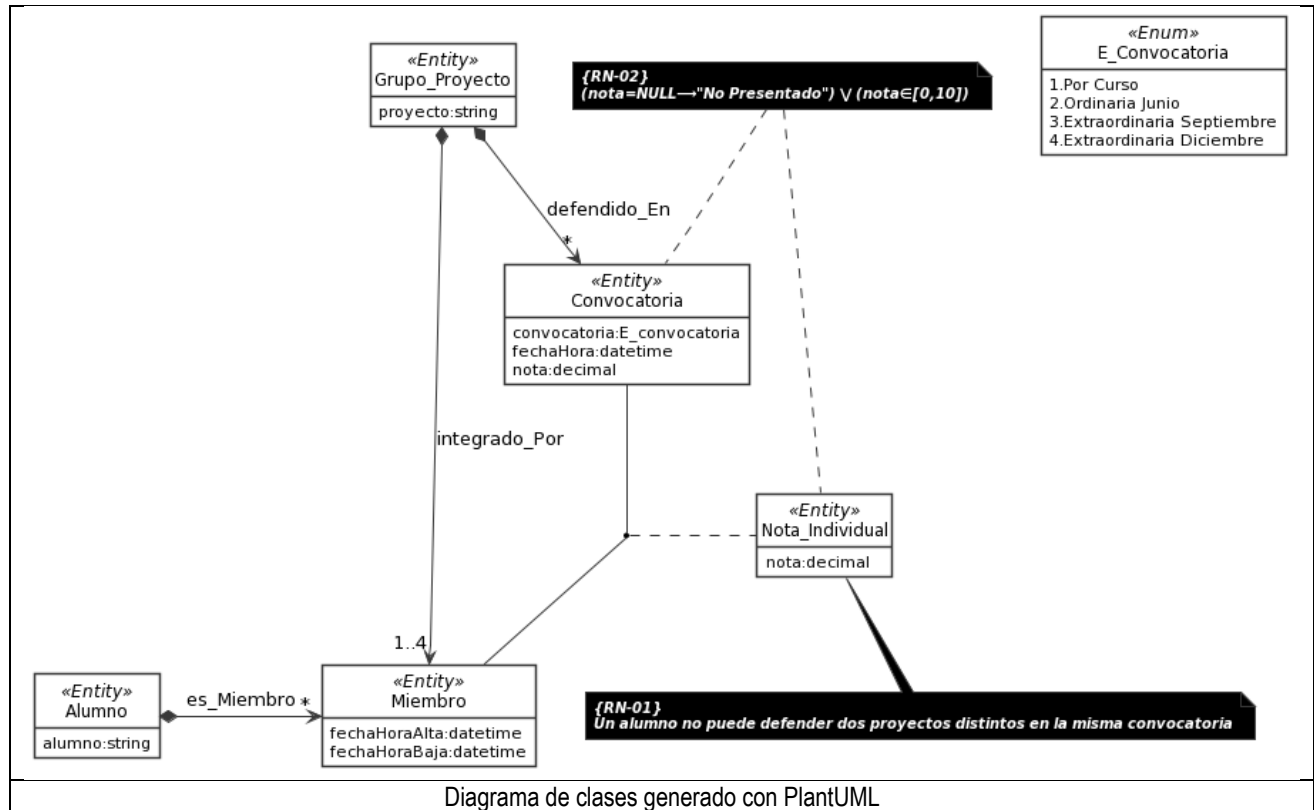
Como: Coordinador de IISSI-1

Necesito: La calificación de un grupo debe ser NULL (No presentado) o bien estar en el intervalo [0,10].

Para: Evitar inconsistencias en las calificaciones.

Elabore un modelo conceptual de diagramas de clases UML. Si identifica algún problema en el enunciado, indíquelo junto con el modelo desarrollado.

## Solución S1



```

@startuml
'Protectora de animales
'Opciones gráficas
hide methods
hide circle
skinparam monochrome true
skinparam shadowing false
skinparam ClassBackgroundColor white
skinparam Note {
    Backgroundcolor black
    FontColor white
    FontStyle Italic bold
    FontSize 10
}

'Enumerados
enum E_Convocatoria <<Enum>> {
    1. Por Curso
    2. Ordinaria Junio
    3. Extraordinaria Septiembre
    4. Extraordinaria Diciembre
}

'Entidades
class Grupo_Proyecto <<Entity>> {
    
```

```
proyecto:string
}
class Alumno <<Entity>> {
  alumno:string
}
class Convocatoria <<Entity>> {
  convocatoria:E_convocatoria
  fechaHora:datetime
  nota:decimal
}
class Nota_Individual <<Entity>> {
  nota:decimal
}
class Miembro <<Entity>> {
  fechaHoraAlta:datetime
  fechaHoraBaja:datetime
}

'Asociaciones
Grupo_Proyecto *-down-> "1..4" Miembro : "integrado_Por"
Alumno *-right-> "*" Miembro : "es_Miembro"
Grupo_Proyecto *-down-> "*" Convocatoria : "defendido_En"
(Convocatoria, Miembro) .. Nota_Individual

'Reglas de negocio
note bottom of Nota_Individual: {RN-01} \nUn alumno no puede defender dos proyectos distintos en la misma convocatoria

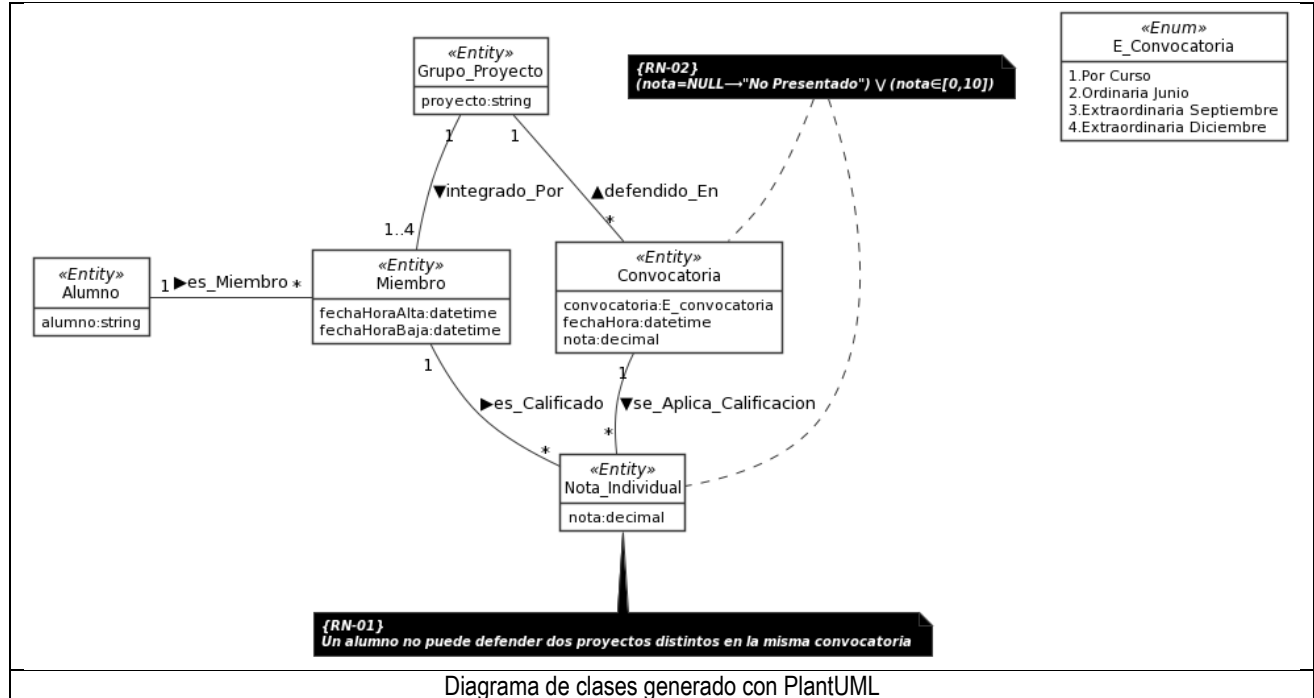
note as N2
{RN-02}
(nota=NULL→"No Presentado") V (nota∈[0,10])
end note

N2..Convocatoria
N2..Nota_Individual

@enduml
```

Código PlantUML (probar en <https://www.planttext.com/>)

Solución S2 (Sin composiciones ni asociaciones)



```

@startuml
'Protectora de animales
'Opciones gráficas
hide methods
hide circle
skinparam monochrome true
skinparam shadowing false
skinparam ClassBackgroundColor white
skinparam Note {
    Backgroundcolor black
    FontColor white
    FontStyle Italic bold
    FontSize 10
}

'Enumerados
enum E_Convocatoria <<Enum>> {
1.Por Curso
2. Ordinaria Junio
3. Extraordinaria Septiembre
4. Extraordinaria Diciembre
}

'Entidades
'-----
class Grupo_Proyecto <<Entity>> {
proyecto:string
}
class Alumno <<Entity>> {

```

```

alumno:string
}
class Convocatoria <<Entity>> {
convocatoria:E_convocatoria
fechaHora:datetime
nota:decimal
}
class Nota_Individual <<Entity>> {
nota:decimal
}
class Miembro <<Entity>> {
fechaHoraAlta:datetime
fechaHoraBaja:datetime
}

'Asociaciones
Grupo_Proyecto "1"-down- "1..4" Miembro : ▼ integrado_Por
Alumno "1"-right- "*" Miembro : ► es_Miembro
Grupo_Proyecto "1"-down- "*" Convocatoria : ▲ defendido_En
Convocatoria "1"-down- "*" Nota_Individual : ▼ se_Aplica_Calificacion
Miembro "1"-right- "*" Nota_Individual : ► es_Calificado

'Reglas de negocio
note bottom of Nota_Individual: {RN-01} \nUn alumno no puede defender dos proyectos distintos en la misma convocatoria

note as N2
{RN-02}
(nota=NULL→"No Presentado") V (nota∈[0,10])
end note

N2..Convocatoria
N2..Nota_Individual

@enduml

```

Código PlantUML (probar en <https://www.planttext.com/>)