 <p>Escuela Técnica Superior de Ingeniería Informática Departamento de Lenguajes y Sistemas Informáticos</p>	Introducción a la Ingeniería del Software y a los Sistemas de Información
	<b>Prueba de modelado conceptual</b> Septiembre 2018

### Enunciado:

La transcripción que aparece a continuación corresponde a una entrevista realizada al gerente de un teatro para determinar los objetivos y requisitos de una aplicación web que gestione la programación anual de espectáculos y venta de entradas a dichos espectáculos:

**Cuestión ©:** Para empezar, hábleme de los espectáculos. Me interesa conocer la información que debe gestionarse por parte de la aplicación web.

**Respuesta ®:** De acuerdo, los espectáculos que se programan en nuestro teatro pueden ser de distinto tipo (p.e. “Concierto”, “Opera”, .. ). Cada espectáculo es de un tipo.

**C:** ¿Qué información desea recoger de los espectáculos?

**R:** Básicamente, su denominación, tipo, duración y fecha y hora en las que se representan. Un espectáculo puede tener varias representaciones.

**C:** ¿Puede haber más de una representación el mismo día?

**R:** Sí, hay días con más de una representación pero nunca del mismo espectáculo.

**C:** ¿Algo más acerca de los espectáculos?

**R:** Se me olvidaba, también quiero guardar el precio. Cada tipo de espectáculo tiene un precio de la entrada.

**C:** ¿El precio varía según tipo de espectáculo?

**R:** Sí, pero también depende de la zona del teatro donde esté situada la localidad.

**C:** Acláreme esto, por favor.

**R:** El teatro está dividido en zonas. Las localidades más caras corresponden a la zona mejor situada, es decir la zona de “Patio”. Otras zonas son “1ª de Balcón”, “1ª de Terraza” etc. Cada zona tiene distinto precio.

**C:** En definitiva ¿el precio depende del tipo de espectáculo y zona del teatro?

**R:** Así es. También quiero conocer las localidades vendidas para cada representación.

**C:** ¿Hablamos de las entradas?

**R:** Una entrada es una localidad para una representación. Dentro de cada zona las localidades se identifican por fila y butaca (p.e. “fila 2, butaca 3 de Patio”, “fila 2, butaca 3 de 1ª de Balcón”,..).

**C:** ¿Quiere que se puedan adquirir las entradas únicamente a través de la web?

**R:** No, seguiremos vendiendo entradas en taquilla. Además algunas entradas se regalan como invitación. Necesito saber el medio o canal por el que se han adquirido las entradas ya sea web, taquilla o invitación.

**C:** ¿Qué información quiere almacenar de las entradas?

**R:** La fecha y hora de compra, el canal por el que se ha adquirido, representación, fila y butaca que le corresponde y el precio de compra al que se ha adquirido ya que puede variar con el paso del tiempo.

**C:** ¿Algo más?

**R:** Creo que en principio es todo. Como ya le comenté, queremos que tanto la planificación de espectáculos como la venta de entradas estén gestionados por la aplicación y se puedan consultar a través de Internet.

### Ejercicio:

Teniendo en cuenta el enunciado anterior, elabore un modelo conceptual usando tantos diagramas de clases UML como considere necesario. Si identifica algún problema en el enunciado, indíquelo junto con el modelo desarrollado.

Solución

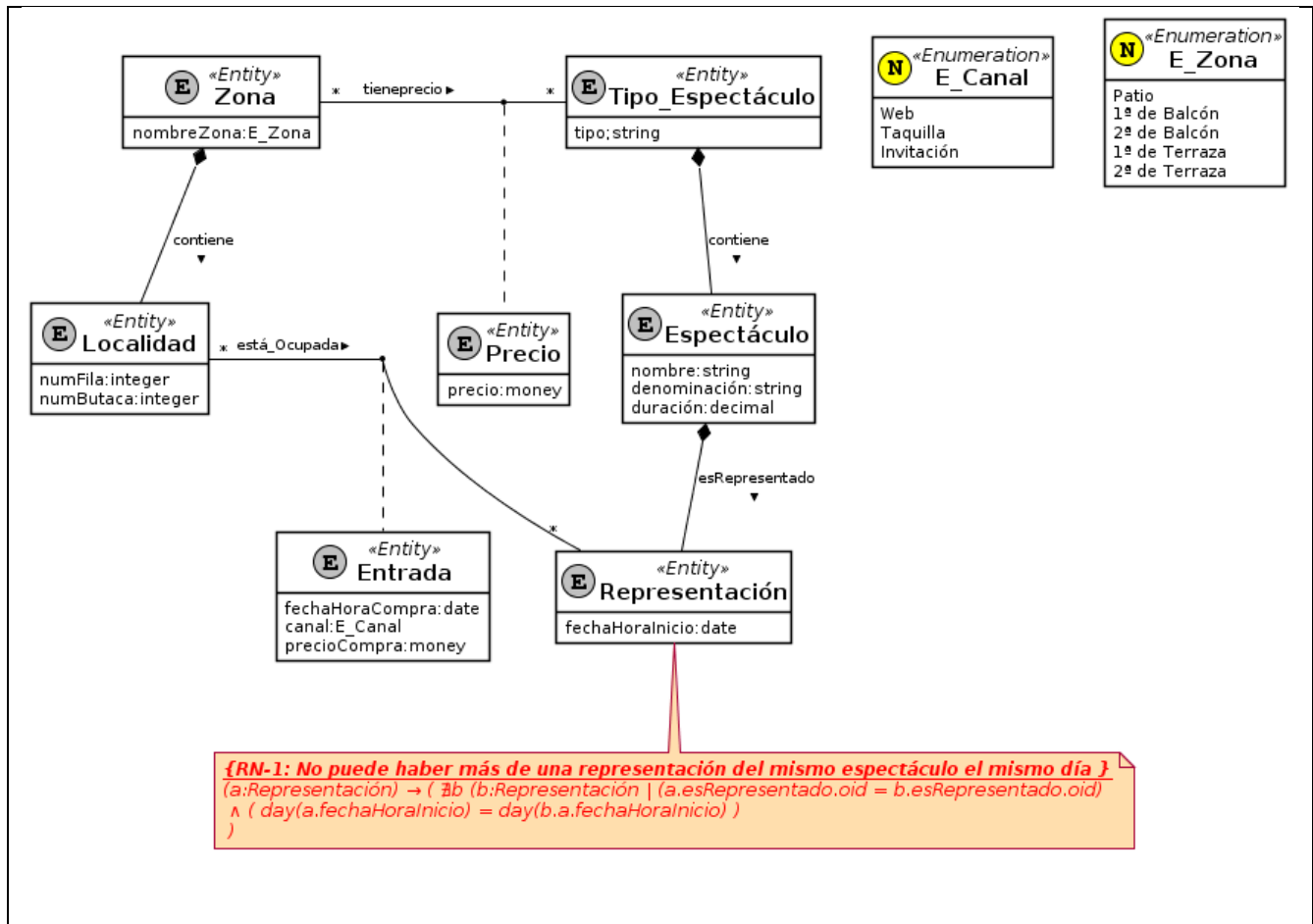



Diagrama de clases generado con PlantUML

```
@startuml
'Alquileres
'Opciones gráficas
skinparam class {
    FontSize 15
    FontStyle bold
    BackgroundColor transparent
    borderColor black
}
skinparam arrow {
    FontSize 10
    Color black
}
skinparam classAttributeIconSize 0
skinparam note {
    FontSize 12
    FontStyle italic
    FontColor Red
    BackgroundColor NavajoWhite
}
skinparam shadowing false
hide methods
hide circle

class Espectaculo << (E,silver) Entity >> {
    denominación:string
    duración:decimal
}
class Representación << (E,silver) Entity >> {
    fechaHoraInicio:date
}
class Zona << (E,silver) Entity >> {
    nombreZona:E_Zona
}
class Localidad << (E,silver) Entity >> {
    numFila:integer
    numButaca:integer
}
class Precio << (E,silver) Entity >> {
    precio:money
}
class Entrada << (E,silver) Entity >> {
    fechaHoraCompra:date
    canal:E_Canal
    precioCompra:money
}
```

 <p>UNIVERSIDAD DE SEVILLA Escuela Técnica Superior de Ingeniería Informática Departamento de Lenguajes y Sistemas Informáticos</p>	Introducción a la Ingeniería del Software y a los Sistemas de Información
	<b>Prueba de modelado conceptual</b> Septiembre 2018

<pre>' Entidades enum E_Canal &lt;&lt; (N,yellow) Enumeration &gt;&gt; { Web Taquilla Invitación } enum E_Zona &lt;&lt; (N,yellow) Enumeration &gt;&gt; { Patio 1ª de Balcón 2ª de Balcón 1ª de Terraza 2ª de Terraza } class Tipo_Espectáculo &lt;&lt; (E,silver) Entity &gt;&gt; { tipo:string } class Espectáculo &lt;&lt; (E,silver) Entity &gt;&gt; {</pre>	<pre>} 'Asociaciones Tipo_Espectáculo *-down- Espectáculo : contiene &gt; Espectáculo *-down- Representación : esRepresentado &gt; Zona *-down- Localidad : contiene &gt; Zona "*" - "*" Tipo_Espectáculo : tieneprecio &gt; Localidad "*" - "*" Representación : está_Ocupada &gt; (Representación , Localidad) .. Entrada (Tipo_Espectáculo , Zona) .. Precio ' Restricciones note bottom of Representación &lt;b&gt;&lt;u&gt;{RN-1: No puede haber más de una representación del mismo espectáculo el mismo día }&lt;/u&gt;&lt;/b&gt; (a:Representación) → ( ∄b (b:Representación   (a.esRepresentado.oid = b.esRepresentado.oid) ∧ ( day(a.fechaHoralInicio) = day(b.a.fechaHoralInicio) ) ) end note @enduml</pre>
Código PlantUML (probar en <a href="https://www.planttext.com/">https://www.planttext.com/</a> )	