

# Informe Fase 1

DSINToxicación del bit de ACarreo

Eduardo Giménez Domínguez

Germán Gil Planes

Javier Melgarejo Teruel

Grupo de trabajo: Subgrupo 1.3

24 de noviembre de 2024

## Índice

<b>1. Introducción</b>	<b>2</b>
<b>2. Diseño de la ontología</b>	<b>2</b>
<b>3. Clases de nuestro programa</b>	<b>2</b>
3.1. Personaje . . . . .	2
3.1.1. Dios . . . . .	3
3.1.2. Semidios . . . . .	3
3.1.3. Criatura . . . . .	3
3.1.4. Humano . . . . .	3
3.2. Obtenible . . . . .	3
<b>4. Relaciones</b>	<b>4</b>
<b>5. Árbol de jerarquía de clases</b>	<b>5</b>

## 1. Introducción

En este informe, incluimos el desarrollo de la ontología del dominio. Mostraremos las clases, así como la jerarquía entre ellas y sus atributos y relaciones.

## 2. Diseño de la ontología

Para diseñar la ontología de nuestro sistema, hemos usado la metodología de “desarrollo de ontologías 101”.

1. Hemos determinado el dominio y el alcance de la ontología, respondiendo a preguntas básicas como ¿cuál es el dominio que cubrirá la ontología? o ¿para qué la usaremos?
2. En segundo lugar, hemos aislado los términos importantes, como Personaje y sus subclases, Obtenible y sus subclases o las acciones y relaciones que hay entre ellas.
3. Posteriormente, hemos definido las clases y la jerarquía entre ellas, usando el enfoque descendente. Primero, distinguimos entre las clases Personaje, Acción y Obtenible, y a raíz de ahí empezamos a definir subclases. En el proceso, aparecieron también otras como PosesionCapacidad o TratoPersonaje.
4. Luego, hemos definido los slots que se requieren en cada clase, tanto las clases raíz como sus subclases, y también restricciones.
5. Por último, hemos creado instancias de las distintas clases.

Para ello, hemos usado la herramienta de desarrollo de ontologías Protégé.

## 3. Clases de nuestro programa

### 3.1. Personaje

Como su propio nombre indica, es una clase que engloba a todos los “seres” que aparecen en nuestra historia. Debido a esa generalización, es una clase abstracta de la que heredan “Dios”, “Semidios”, “Humano” y “Criatura”. Dentro de ella, tenemos los siguientes slots:

- Nombre: un String que indica cómo se llama el personaje.
- Estado: indica si el personaje está libre, preso o muerto. Se selecciona de un enumerado, pero inicialmente siempre es “LIBRE”.
- esMortal: booleano que indica si un personaje puede cambiar su estado a “MUERTO” o no.

- Ubicación: indica dónde se encuentra el personaje. Se selecciona de un enumerado.
- ataque: un número entero que indica la fuerza de ataque del personaje.

#### 3.1.1. Dios

Clase que define a los dioses de nuestra historia. Hereda de la clase personaje, con la particularidad de que el slot “esMortal” siempre es **false**, pues hemos supuesto que un dios nunca muere. De base, tendrán ataque 100, y Ubicacion será siempre “OLIMPO”.

#### 3.1.2. Semidios

Clase que define a los semidioses de nuestra historia. Hereda de la clase personaje, y el slot “esMortal” siempre es **true**, pues suponemos que un semidios siempre puede morir. Además, incorpora un nuevo slot:

- esHeroe: booleano que nos indica si el semidios es un héroe.

#### 3.1.3. Criatura

Clase que define a los distintos tipos de “monstruos o criaturas mitológicas”. Hereda de la clase personaje, y el slot “esMortal” siempre es **true**.

#### 3.1.4. Humano

Clase que define a todos los seres humanos de nuestra historia. Hereda de la clase personaje, y, al igual que en semidios, se incorpora el atributo “esHeroe”. Como los humanos siempre pueden morir, el atributo “esMortal” es siempre **true**.

### 3.2. Obtenible

Clase abstracta para representar los distintos obtenibles que aparecen a lo largo de la historia. Hemos supuesto que un objeto tiene:

- Nombre: un String que indica cómo se llama el objeto.
- Personaje: instancia de la clase Personaje, que nos indica qué personaje es el poseedor del objeto. Solamente puede tener un poseedor.
- Ubicación: indica dónde se encuentra el personaje. Se selecciona de un enumerado.
- ataques: número entero que indica la fuerza de ataque del objeto.

De ella, heredan dos clases:

- Objeto: clase que define a los objetos físicos que se pueden obtener a lo largo de la historia.
- Animal: clase que define a los animales que se pueden obtener a lo largo de la historia. Incorporan los slots esMortal y estado.

## 4. Relaciones

Una vez definidas las clases, hemos definido una clase, Accion, que define las distintas acciones que se pueden realizar en nuestra historia. De ella, heredan las clases:

- Apresamiento: un Personaje puede apresar a otro.
- Asesinato: un Personaje puede asesinar a otro.
- Combate: dos Personajes pueden combatir entre sí, cada uno con un ataque.
- Intercambio: dos Personajes pueden intercambiar objetos.
- Liberación: un Personaje puede liberar a otro.
- Localización: un Personaje puede tener localizado a otro.
- Robo: un Personaje puede robar a otro un objeto.

Por otro lado, tenemos una clase “TratoPersonaje” que define las distintas relaciones que se pueden dar entre personajes. De ella, heredan las clases:

- Amistad: un Personaje puede ser amigo de otro.
- Enemistad: un Personaje puede ser enemigo de otro.
- Enamoramiento: un Personaje puede estar enamorado de otro.

También hemos definido una clase “PosesionCapacidad” que define las distintas relaciones que se pueden dar entre un personaje o un obtenible y la capacidad que poseen. De ella, heredan las clases:

- : PosesionObtenibleCapacidad: un Obtenible puede poseer una capacidad.
- PosesionPersonajeCapacidad: un Personaje puede poseer una capacidad.

Quedarán definidas reglas que asignarán las capacidades a los personajes y a los obtenibles, y que harán que si un personaje posee un objeto, obtendrá la capacidad que este objeto le otorga.

Por último, hemos definido la clase Objetivo, independiente del resto, para gestionar los diferentes objetivos que se pueden realizar.

## 5. Árbol de jerarquía de clases

Finalmente, hemos creado un árbol de jerarquía de clases, que se muestra en la figura 1.

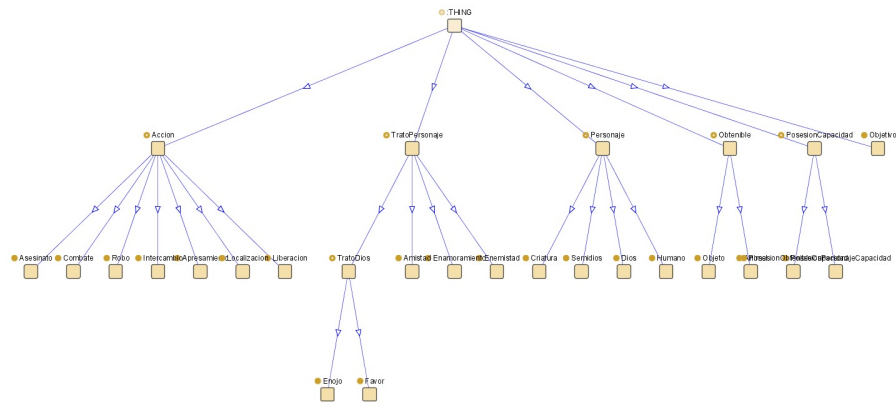


Figura 1: Árbol de jerarquía de clases

## Referencias

- [1] Universidad de Murcia (2024). *Apuntes de la asignatura de Desarrollo de Sistemas Inteligentes*. Disponible en: [www.aulavirtualum.es](http://www.aulavirtualum.es)