

# D20 – HISTORY CREATOR



*Eduardo J. Marrero Rodríguez.  
Claudio Roldán Ramos.*

# INTRODUCCIÓN

*En este documento se especifica tanto la descripción del proyecto, su finalidad, diferentes niveles de complejidad (a cumplir dependiendo del tiempo disponible), así como el modelo de datos que conllevará.*

## ÍNDICE

<b>DEFINICION.....</b>	<b>2</b>
<u>La mesa de juego:.....</u>	<u>3</u>
<u>Finalidad:.....</u>	<u>3</u>
<b>CONTENIDO.....</b>	<b>3</b>
<u>Nivel 1: Básico.....</u>	<u>3</u>
<u>Nivel 2: Medio.....</u>	<u>3</u>
<u>Nivel 3: Programa Completo.....</u>	<u>4</u>
<b>Aspectos técnicos de JavaFX.....</b>	<b>4</b>

### DEFINICION.

D20 – History Creator es una herramienta destinada a la organización de datos pertenecientes o relativos a todo el proceso de creación de una historia que incluye aventuras (quests), y personajes que viven inmersos en su interior.

Basado en el sistema de juego D20 System, History Creator permitirá al denominado Game Master (Director de Juego) disponer de un programa que guarde y organice todo el contenido de sus historias que posteriormente presentar ante **la mesa de juego**.

### La mesa de juego:

Es aquel lugar dónde se reúnen tanto el Director de Juego como los Jugadores de la partida. Antaño, toda la gestión de las partidas se realizaba a lápiz y papel y es ahí dónde entra History Creator, el programa pasa a ser la mesa de juego y por tanto,

debe comportarse como tal, siendo capaz de gestionar toda la información que ocurra en la mesa de juego.

### Finalidad:

Uno de los problemas más importantes de los juegos de rol de mesa es la volatilidad de las campañas que se crean. Todos los datos; criaturas, personajes, objetos, etc., acaban por perderse dado el formato analógico (papel y lápiz), impidiendo el poder dar forma narrativa a tales datos.

Por tanto, es imprescindible que History Creator ayude no sólo a generar y guardar de forma intuitiva las campañas creadas por el Director de Juego, sino que además le sirva de herramienta para poder jugarlas y, sobretodo, que permita exportarlas en un formato propio que poder compartir.

## **CONTENIDO.**

El programa deberá estar lo suficientemente cerrado como para englobar todo el modelo de juego de D20 System, sin embargo, dado que esta tarea podría necesitar de más tiempo del disponible (menos de 2 meses), hemos decidido dividir el contenido del mismo en varios niveles:

### Nivel 1: Básico.

La versión básica de D20 – History Creator contendrá la siguiente especificación de requisitos:

- Capacidad para introducir Campañas
- Notas de campaña (Trasfondo)
- Aventuras
- Calculadora d20
- Deberá tener persistencia (Se baraja un fichero XML)

### Nivel 2: Medio

Además del nivel Básico, también deberá tener:

- Lista de Personajes (Jugadores).
- Capacidad para introducir fichas de personajes.
- Capacidad para introducir un compendio de Criaturas
- Capacidad para introducir un compendio de Tesoros

### Nivel 3: Programa Completo

Además del nivel Medio, también deberá tener:

- Capacidad para gestionar encuentros (combates), mediante una lista que contemple al menos, los puntos de vida de los personajes, su defensa (CA) y

sus posibles tiradas. Esta lista deberá poder ordenarse de mayor a menor dependiendo de las tiradas de iniciativa respectivas de sus integrantes, dando así la sensación de disponer de un combate por turnos.

- Capacidad para vincular tableros de combate sobre los que se pueda pintar (a modo de pizarra)
- Capacidad de extraer de los datos introducidos en el programa, los títulos de aquellos eventos más importantes que transcurran durante la partida para pasarlos a un formato de texto editable, ordenados y que en el futuro ayuden al creador de la campaña a desarrollar narrativa sobre ésta.

## **Aspectos técnicos de JavaFX.**

El programa contendrá los siguientes aspectos técnicos:

- Gestión de la Información:
  - ListView para gestionar Aventuras, Encuentros, Personajes, Notas.
  - ImageView para gestionar todas las imágenes de la Campaña.
  - Tareas en segundo plano para abrir nuevos Stages que contengan información en su interior, evitando así el colapso del programa en caso de que la carga de datos no pudiera realizarse.