

# Informática Gráfica

# Índice

[Instrucciones y Características](#)

[Diseño e Implementación](#)

[Planificación](#)

[Descargas](#)

[Licencias y Agradecimientos](#)

# Instrucciones y Características

## Ejecución del programa

Está disponible en [este enlace](#) cada una de las diferentes versiones disponibles del juego (GNU/Linux, Windows, macOS). Para ejecutar el juego depende de la versión del sistema que estés usando:

- setup.cmd: Se deberá hacer un doble click
- setup.sh: Ejecutar desde el terminal el fichero

Este programa es portable, lo que significa que no modifica ni altera **ningún** registro y **no** requiere instalación/desinstalación puesto que los ficheros que se descargan son simplemente ejecutados.

**Nota:** Si en la ventana de comandos aparece alguna advertencia (TiXmlDocument load failed), se trata de un error al cargar los objetos y texturas puesto que el interprete está intentando buscar los modelos sin estos haberse descomprimido y almacenado en la memoria.

## Instrucciones del juego

Algunos de los comandos más importantes que realiza el juego

**Esc**: Salir del juego

**w**: Ir hacia adelante

**s**: Ir hacia atrás

**a**: Ir hacia la izquierda

**d**: Ir hacia la derecha

**Ratón**: Girar hacia cualquier dirección

**Click izquierdo**: El jugador se desplaza más rápido, pudiendo ascender por las escaleras

## Funcionamiento del juego

El juego muestra el avance del día y noche actual en el cual:

- Amanece a las 9 hasta las 10 de la mañana
- El sol se pone desde las 21 hasta las 22 de la noche

Además cada cierto tiempo emite sonidos diferentes dependiendo de si es de día o de noche.

# Diseño e Implementación

## ¿Qué hemos hecho?

El proyecto consta de dos partes bien diferenciadas, la primera de modelado en Blender y la segunda de creación de entorno y programación del motor de juego en Maratis. El juego consiste en un escenario de ambientación medieval/renacentista, donde el usuario controla al personaje en primera persona que puede caminar libremente. La escena es un pequeño poblado dentro de una muralla, y con un pequeño bosque.

## ¿Cómo se hizo?

El proyecto se ha dividido en dos partes Blender y Maratis.

### Blender

---

Al inicio del proyecto, la intención era crear nuestros propios modelos e introducirlos en el motor de Maratis, pero tras un periodo de aprendizaje de Blender, y darnos cuenta lo costoso que era, descubrimos que no era viable hacer la totalidad de los modelos desde el inicio. Los modelos 3D, fueron seleccionados de Internet, de páginas que suministraban modelos con licencias de uso libre. El uso de modelos externos no fue fácil debido a las exigencias de Maratis, y hubo que hacer una gran variedad de tareas:

1. Comprobar que el modelo esté adaptado y programado para Blender, en caso contrario importarlo adecuadamente.
2. Añadir texturas a aquellos modelos que no tuviesen y corregir los defectos tanto de modelado, como de texturas.
3. Corregir problemas de texturas derivados del uso en Maratis, algo que llevó mucho tiempo y esfuerzo debido a lo restrictivo que es Maratis en este sentido. Los materiales deben tener ajustados correctamente un conjunto de propiedades (de manera muy restringida) y las capas estar en el orden estricto que obliga Maratis.
4. Ajustar los tamaños de los modelos para que estén en la misma escala relativa.

El entorno (suelo, cielo, y limitaciones de áreas) fue hecho completamente desde el inicio en Blender, con sus correspondientes texturas.

### Maratis

---

El motor de juego Maratis y sus peculiaridades, nos dieron los problemas con las texturas mencionadas anteriormente, que hacían que los modelos se vieran completamente opacos, oscuros, y que reflejaran la luz. Algunas de las restricciones más importantes con las mallas creadas en Blender se encuentran disponibles [aquí](#).

Algunas de las tareas más relevantes fueron:

#### Exportación de los diferentes modelos a Maratis

Los modelos fueron importados a Maratis y adaptados, puesto que al tener todos ellos texturas debían de tener las rutas relativas adecuadas para la representación de los materiales. Para importar las mallas se debe de añadir a maps todas las texturas (se han organizado por carpetas) y a mesh todas las mallas.

#### Orden y compatibilidad de las capas

Las capas deben de estar en este orden para una correcta exportación de los modelos:

1. **Difuse (d)**: Es la capa encargada de detallar el color principal del objeto
2. **Specular**: Determina el brillo o la oscuridad del objeto en los diferentes puntos de la textura. (Una piedra y un cristal, poseen diferentes niveles de reflejo, no tiene que ver con la forma en sí, si no con el material usado)
3. **Normal**: Muestra toda la geometría del objeto, adaptándolo a la textura adecuada (Una textura de una ventana sobre un plano no tiene el mismo grosor en el marco que en el propio cristal).
4. **Emit/Occlusion (a)**: Es la luz que emite el objeto en todas sus áreas. (Si el objeto posee una esquina, la esquina no emite la luz con la misma intensidad en todo el objeto).
5. **Displacement (h)**: No usaremos esta capa.

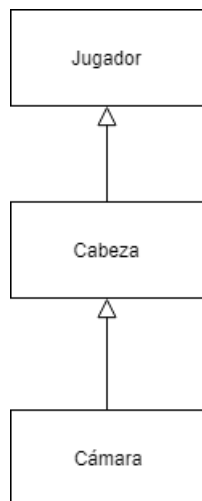
**LAS CAPAS DEBEN DE ESTAR EN EL MISMO ORDEN EXPUESTO PARA EVITAR INCOMPATIBILIDADES BLENDER/MARATIS**

## Jugador

Una vez solventados todos los problemas de modelado y texturización, se intentó realizar un personaje articulado que se desplazara por el entorno, se comprobó la dificultad, no solo de modelar los huesos, unirlos al propio modelo y que este realizara movimientos, si no que en conjunto se viera natural para el jugador y adaptado a su propio entorno (movimiento de ropa). Resultó imposible de realizar movimientos tan complejos y se desechó la idea de mostrar un personaje principal.

## Movimiento del jugador

Para realizar el movimiento en realidad lo que se desplaza es el objeto "Jugador", ese objeto, por herencia, desplaza a la cabeza y a su vez, por herencia, desplaza la cámara. Cuando se rota se gira en el eje de las Y (horizonte) se rota todo el cuerpo. Sin embargo cuando se rota sobre el eje de las X (tangente del horizonte) se desplaza la cabeza y con ella la cámara.



## Limitaciones en el mapa

Alrededor del mapa se ha bloqueado el acceso a los lugares que se consideran lejanos para mostrar cierta distancia con el fondo. Son bloques en los que si se entra en contacto con ellos el editor da un efecto de niebla y cuando se aleja desaparece. La niebla en realidad es un efecto de la cámara, que hace que se vea el entorno de diferente forma, según el tiempo en el que se esté en contacto con el bloque aumentará más la niebla hasta alcanzar un cierto límite (se sigue una proporción lineal)

## Ciclo día y noche

Se ha hecho un procedimiento para realizar todas las transiciones entre el día y la noche. En dicho procedimiento se pasa por parámetro el rgb actual, la suma que se quiere alcanzar de rgb y la hora desde la que se empieza. El procedimiento se encargará de devolver un array de valores que se utilizará para dar la luz deseada al foco.

## Sonidos

Los sonidos se ejecutan de cada minuto, de forma alternada. Cada sonido está almacenado en formato en la carpeta sounds. Dependiendo del ciclo de día o de noche se cambiará el sonido (se hace a la hora de detallar si es de día o de noche)

## Fuentes de información utilizadas

Aquí se describen los estándares a seguir en el proyecto. Además se incluyen algunos enlaces utilizados para realizar dicho proyecto (videotutoriales, wikis,...). Para guiarnos en esta tarea utilizaremos diferentes tutoriales/wikis:

### Wiki (Soporte oficial)

---

- [Wiki LUA-Maratis](#)
- [Wiki Maratis](#)
- [Documentación de Blender](#)
- [Compatibilidad Blender Vs Maratis](#)

## Foros

---

- Foro oficial Maratis
- Blender
- Proyecto Fin de carrera UVA

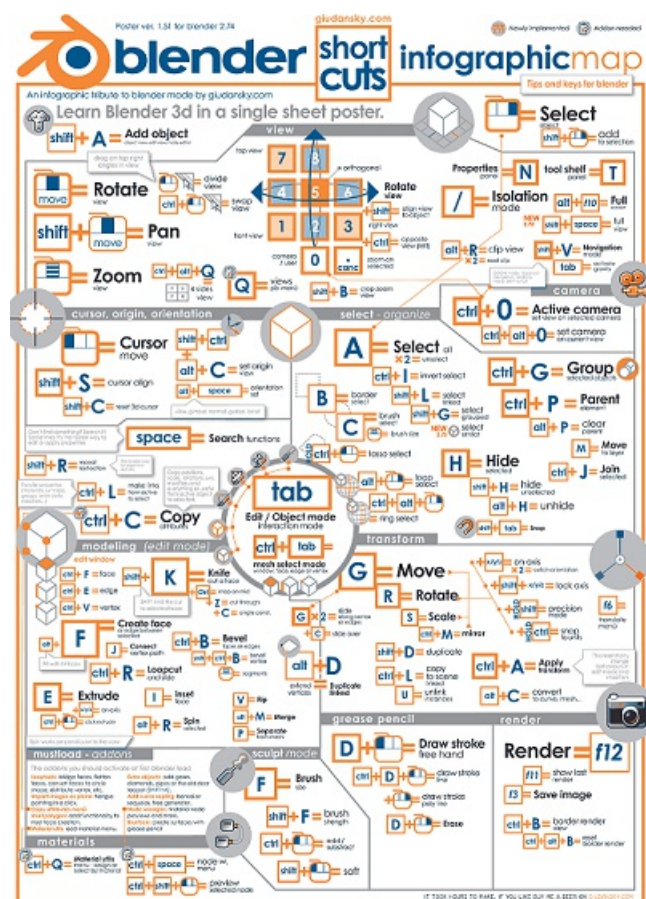
## Videos

- Tutorial de LUA
- El secreto del realismo son las texturas
- Textura de telas en Blender

## Libros

- Fundamentos de Síntesis de Imagen 3D: Un Enfoque práctico a Blender
- Blender y Yafaray: Aprende en 24h Blender y Yafaray. Diseño Gráfico 3D con Software Libre

## Lista de atajos Blender



## Programas Utilizados

- Manatis 3D: [http://www.maratis3d.org/?page\\_id=57](http://www.maratis3d.org/?page_id=57)
- Blender: <https://www.blender.org/download/>
- Add-ons para Blender: <https://goo.gl/zSFLR>
- GIMP: [https://portableapps.com/apps/graphics\\_pictures/gimp\\_portable](https://portableapps.com/apps/graphics_pictures/gimp_portable)

## Texturas

- [textures.com](https://www.textures.com)



# Planificación

A rasgos generales, describiremos el proceso que seguimos a lo largo del proyecto hasta su finalización con los tiempos estimados:

- Durante el primer mes, septiembre de 2017, valoramos sobre la temática y la herramienta con la que desarrollar el proyecto. Al final nos decantamos por Blender por la gran cantidad de tutoriales, y por Maratis, debido a que era completamente libre. La planificación que hicimos se corresponde con lo que encontramos [aquí](#)
- El mes de octubre nos dedicamos a aprender modelado de Blender.
- El mes de noviembre lo dedicamos a la búsqueda de modelos y aprendizaje de Maratis.
- Durante diciembre y principios de enero, nos dedicamos a exportar y juntar los modelos bajo el mismo escenario, realizamos toda la parte de programación en Lua y en definitiva se acabó el proyecto.

Contabilizadas, del proyecto hay 40 horas de trabajo colectivo. A estas horas habría que sumarle el trabajo individual realizado, pudiendo sobrepasar las 50 horas. Algunos de los motivos por los que el proyecto se extendió tanto fueron:

1. Los problemas con las texturas en Maratis, que costaron muchísimas horas solucionarlos.
2. Problemas con las texturas importadas, errores que había que solucionar.
3. El intento de añadir un personaje visto en 3º persona, con sus animaciones correspondientes para el movimiento. Primero debíamos hacer el esqueleto que había que asociar al modelo, crear la animación, y después añadirlo a Maratis.

Toda la información de tiempos en el proceso de implementación está accesible [aquí](#) (cada vez que se añadía modelos o se alteraba el proyecto).



# Descargas

Descargas disponibles:

- [GNU/Linux](#)
- [Windows](#)
- [macOS](#)

# Licencias y Agradecimientos

## Licencias

- Ángel: <https://graphics.stanford.edu/projects/mich/mgantry-in-lab/mgantry-in-lab.html> by University of Stanford
- Catedral: <https://www.blendswap.com/blends/view/75990> by Daniel74
- Casa1: <https://www.turbosquid.com/3d-models/medieval-house-x-free/606468> by LBdN
- Casa2: <https://www.turbosquid.com/3d-models/free-medieval-house-3d-model/727769> by Sayoend
- Casa3: <https://free3d.com/3d-model/old-farm-house-91130.html> by Animated Heaven (animatedheaven)
- Tienda: <https://free3d.com/3d-model/trade-tent-52279.html> by kaos3d
- Árbol1: <https://free3d.com/3d-model/tree02-35663.html> by rezashams (rezashams313)
- Árbol2: <https://free3d.com/3d-model/a-tree-88011.html> by Alan Sha Salim (alanthegamer)

## Especial agradecimiento

GiudanSky: <http://giudansky.com/>