



# **MANUAL DE TECNICO DE MUSEUM SPACE**

APLICACIÓN MÓVIL PARA LA BÚSQUEDA DE IMÁGENES ESPACIALES EN UN  
MUSEO

**Presentado por:**

Cristian Andrés Parrado Barreto  
Candido Stiven Moreno Serrato

Universidad de los Llanos  
Facultad de ciencias básicas e ingeniería  
Villavicencio, Meta  
2019

## Contenido

<b>Introducción</b>	<b>3</b>
<b>Objetivo</b>	<b>3</b>
General	3
Específicos	3
<b>Requisitos del sistema</b>	<b>3</b>
<b>Requisitos mínimos para versión Android</b>	<b>3</b>
<b>Herramientas utilizadas para el desarrollo</b>	<b>4</b>
Unity 3D	4
Visual studio code	4
Gitlab	4
<b>Casos de uso</b>	<b>4</b>
<b>Requerimientos</b>	<b>5</b>
Funcionales	5
No funcionales	7
<b>Funcionalidades</b>	<b>8</b>
Texturizado del aplicativo móvil.	8
Animación de personaje.	9
Disparadores de imágenes	10

# Introducción

El presente manual describe los pasos para que una persona tenga las bases suficientes para dar soporte a la aplicación móvil y lograr desarrollar y mejorar la aplicación con fines de aprendizaje mediante la herramienta unity 3D.

## Objetivo

### General

Informar y especificar al usuario la estructura y conformación del sistema con el fin de que puedan hacer soporte y modificaciones o actualizaciones a la aplicación Museum Space.

### Específicos

- Dar a conocer la funcionalidad del sistema mediante casos de uso.
- Informar requerimientos funcionales y no funcionales.
- Definir animaciones para el personaje.
- Explicar el proceso adición de nuevas imágenes.
- Disparadores de imágenes
- Mostrar el proceso de texturizado.

## Requisitos del sistema

**Sistema operativo:** Windows 7 SP1+, 8, 10, 64-bit versions only; Mac OS X 10.12+; Ubuntu 16.04, 18.04, and CentOS 7.

**Unidad de procesamiento grafico (GPU):** Tarjeta de video con capacidad para DX10 (shader modelo 4.0).

## Requisitos mínimos para versión Android

- **OS:** Android 4.0 Ice Cream (mínimo).
  - **CPU:** Intel Quad Core.
  - **Memoria RAM:** 1GB.
  - **Almacenamiento:** 110MB.
- Dispositivos compatibles:  
Samsung, Huawei, Moto G5

# Herramientas utilizadas para el desarrollo

Lo más recomendado es ingresar a la página oficial de cada herramienta y descargar su última versión y gratuita.

## Unity 3D

Unity está disponible como plataforma de desarrollo para Microsoft Windows, Mac OS, Linux. La plataforma de desarrollo tiene soporte de compilación con diferentes tipos de plataformas, para el desarrollo de esta aplicación se usa la versión 2018.4.17f1 (64 bits).

## Visual studio code

Es un editor de código fuente desarrollado por Microsoft para Windows, Linux y macOS. Incluye soporte para la depuración, control integrado de Git, resaltado de sintaxis, finalización inteligente de código, fragmentos y refactorización de código.

## Gitlab

Es un servicio web de control de versiones y desarrollo de software colaborativo basado en Git. Además de gestor de repositorios, el servicio ofrece también alojamiento de wikis y un sistema de seguimiento de errores, todo ello publicado bajo una Licencia de código abierto.

## Casos de uso

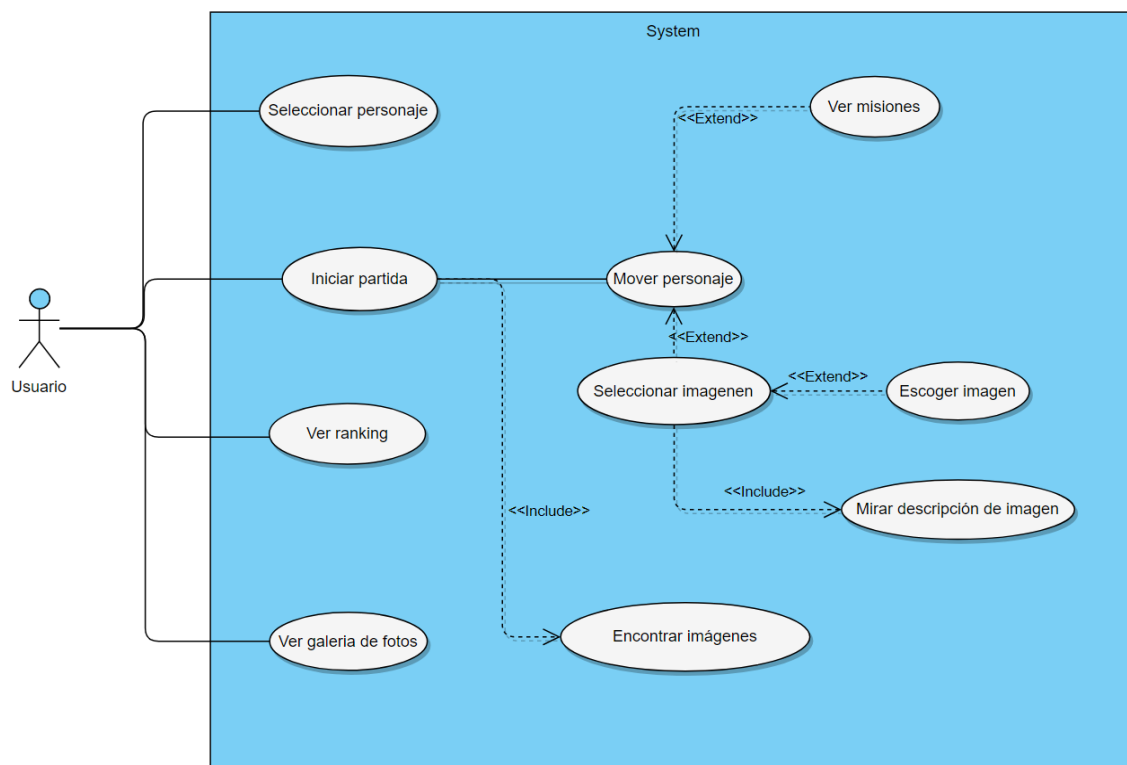


Figura 1. Caso de uso Museum Space.

# Requerimientos

## Funcionales

<b>Nombre:</b> Iniciar nueva partida
<b>Descripción:</b> El sistema debe permitir iniciar una nueva partida siempre que el usuario lo desee.
<b>Actores:</b> Usuario
<b>Precondición:</b> El usuario debe iniciar la aplicación
<b>Postcondición:</b> Se mostrará al usuario dentro del museo con su personaje.
<b>Flujo Principal:</b> <ol style="list-style-type: none"><li>1. El usuario solicita iniciar nueva partida</li><li>2. Se muestra el entorno del museo con su respectivo personaje</li><li>3. El caso de uso finaliza</li></ol>
<b>Flujo Alternativo:</b>

<b>Nombre:</b> Seleccionar personaje
<b>Descripción:</b> El sistema deberá permitir escoger el sexo de un personaje siempre que el usuario lo desee.
<b>Actores:</b> Usuario
<b>Precondición:</b> El usuario debe estar en el menú de inicio.
<b>Postcondición:</b> Se mostrará los personajes disponibles a usar.
<b>Flujo Principal:</b> <ol style="list-style-type: none"><li>1. El usuario solicita seleccionar personaje</li><li>2. Se muestran los personajes disponibles</li><li>3. El usuario elige el personaje que desea usar</li></ol>
<b>Flujo Alternativo:</b> <ol style="list-style-type: none"><li>3a. El usuario elige volver al menú principal</li></ol>

<b>Nombre:</b> Ver ranking
<b>Descripción:</b> El sistema permitirá ver los mejores tiempos en la búsqueda de las imágenes cuando el usuario lo desee.
<b>Actores:</b> Usuario
<b>Precondición:</b> El usuario debe estar en el menú de inicio.

Postcondición: Se mostrará los mejores tiempos.

Flujo Principal:

1. El usuario solicita ver ranking
2. Se muestra los tiempos récords en la búsqueda de imágenes
3. El caso de uso finaliza

Flujo Alternativo:

2a. El usuario elige volver al menú principal

Nombre: Ver galería

Descripción: El sistema le permitirá al usuario ver la galería de imágenes de la página ESO.org

Actores: Usuario

Precondición: El usuario debe estar en el menú de inicio.

Postcondición: Se mostrará las imágenes en mosaicos visibles.

Flujo Principal:

1. El usuario solicita ver galería
2. El sistema muestra las imágenes en mosaicos visibles.
3. El usuario puede seleccionar las imágenes.
4. El sistema mostrará la imagen en un tamaño más grande
3. El caso de uso finaliza

Flujo Alternativo:

2a. El usuario elige volver al menú principal

Nombre: Mover personaje

Descripción: El sistema permitirá que el usuario al usar el touch del celular pueda manipular su personaje

Actores: Usuario

Precondición: Haber elegido iniciar partida

Postcondición: El personaje podrá moverse libremente

Flujo Principal:

1. El usuario mueve personaje con el touch
2. El sistema permitirá movilidad en 3d
3. El caso de uso finaliza

Flujo Alternativo:

Nombre: Seleccionar imágenes
Descripción: El sistema permitirá que el usuario seleccione las imágenes durante su trayecto por el museo
Actores: Usuario
Precondición: Haber movido personaje a un rango cercano de los cuadros
Postcondición: Se abrirá una ventana emergente mostrando información de la imagen
Flujo Principal: 1. El usuario selecciona la imagen 2. Se muestra la imagen con una pequeña descripción sobre ella. 3. El caso de uso finaliza
Flujo Alternativo:

Nombre: Ver misiones
Descripción: El sistema permitirá que el usuario observe las imágenes que tiene como misión buscar.
Actores: Usuario
Precondición: El usuario debe haber iniciado la partida
Postcondición: Se abrirá una ventana emergente mostrando las misiones cumplidas y incompletas.
Flujo Principal: 1. El usuario selecciona el botón de misión. 2. Se muestra la ventana con las misiones
Flujo Alternativo: 2a. El usuario cierra la ventana.

## No funcionales

Título: Compatibilidad con dispositivos.
Descripción: El sistema deberá ser compatible en diferentes teléfonos móviles con sistema operativo Android superior a la 5
Prioridad: Alta

Título: Compatibilidad con resoluciones.
Descripción: La aplicación será compatible con diferentes resoluciones de pantalla.

Prioridad: Alta
-----------------

Título: Compatibilidad con formatos de audio.
---

Descripción: El sistema deberá reproducir distintos formatos de sonido como mp3, AAC, ogg, MIDI y WAV.
--

Prioridad: media
------------------

Título: Mostrar textos legibles.
----------------------------------

Descripción: El sistema se adaptará a la resolución de las pantallas y mostrará los textos legibles para el usuario.
--

Prioridad: Alta
-----------------

Título: Interfaz sencilla e intuitiva
---------------------------------------

Descripción: La aplicación deberá ser sencilla e intuitiva, para que los usuarios con bajos conocimientos en juegos puedan usarla sin que les suponga un esfuerzo adicional
---

Prioridad: Alta
-----------------

## Funcionalidades

### Texturizado del aplicativo móvil.

Teniendo los objetos 3d a utilizar en las escenas del juego, y de forma común con extensión .obj, mediante el uso de materiales con colores específicos, se asignan a los diferentes conjuntos de objetos, permitiendo tomar la textura deseada.



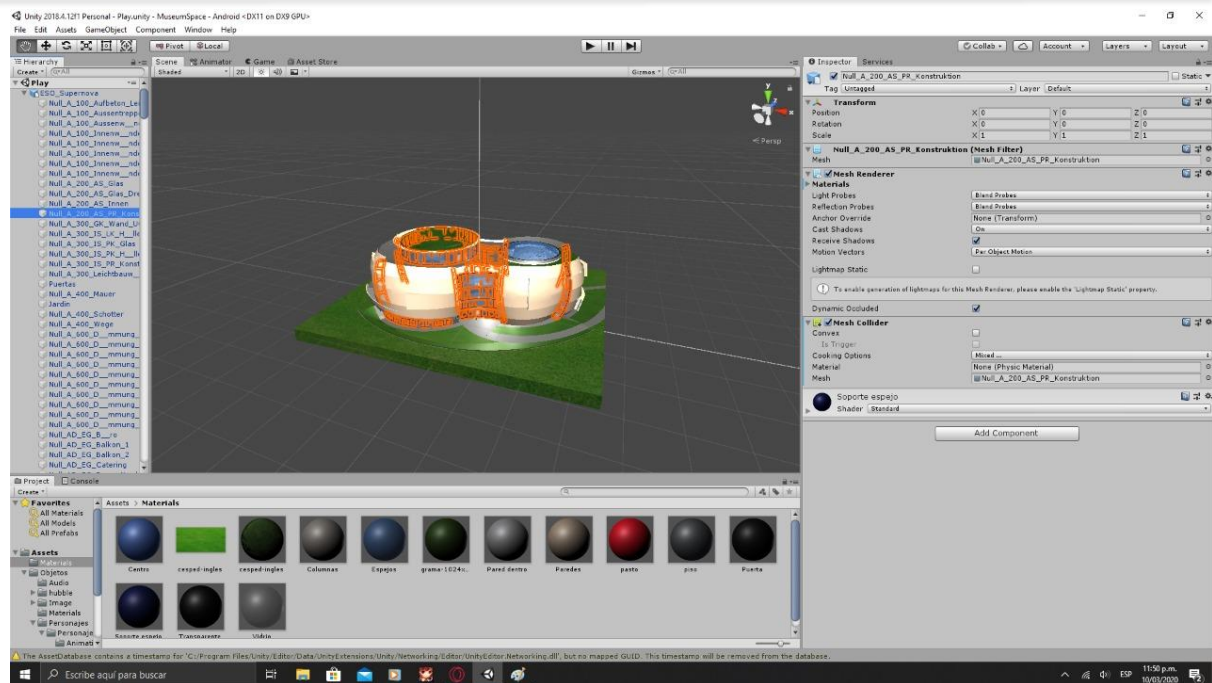


Figura 2. Texturizado del juego.

## Animación de personaje.

Inicialmente se deben descargar las animaciones, posteriormente se le asigna una animación usando un plano cartesiano que simula el movimiento del Joystick, cuando el jugador corre hacia adelante su estado es (0,1) cuando retrocede es (0,-1), si está corriendo y gira a la izquierda es (-1,1), cuando está quieto es (0,0), de este modo se asignan todas las animaciones como lo muestra la figura 3.

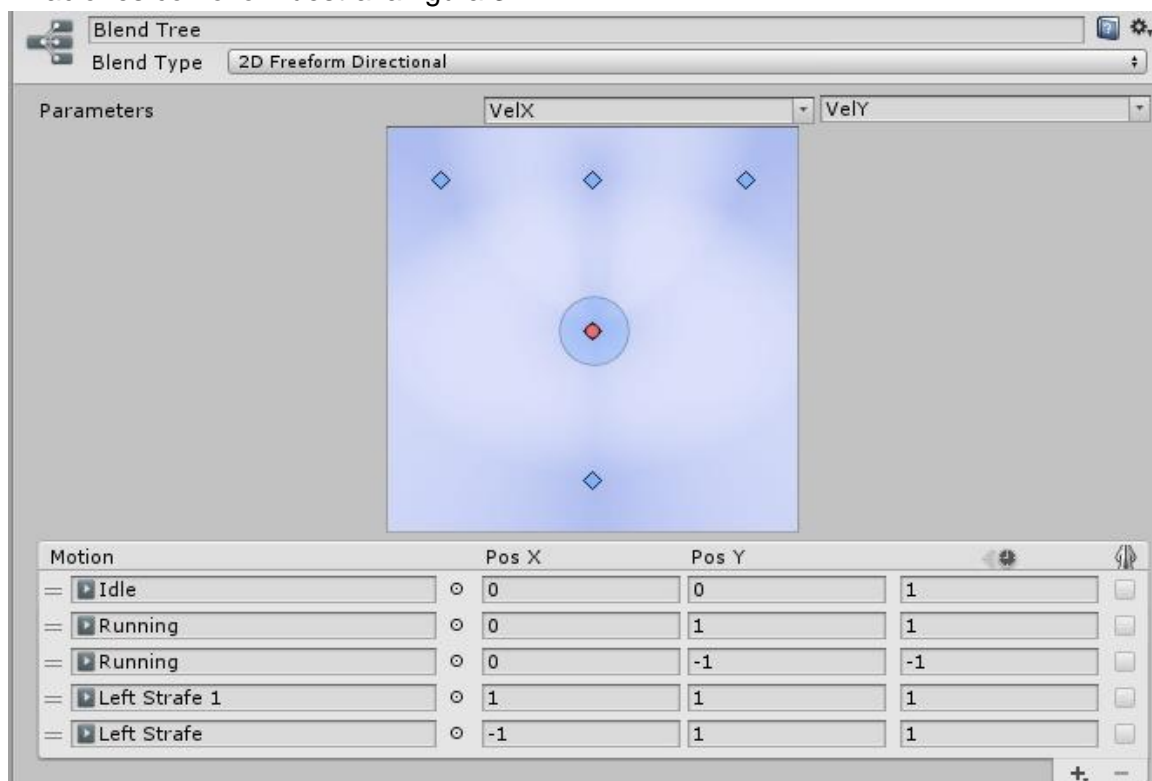


Figura 3. Animación de personaje.

## Disparadores de imágenes

Los cuadros con sus respectivas imágenes son distribuidas en el museo, cada cuadro tiene un “box collider” que cubre un cierto campo de activación que al detectar que un objeto con la etiqueta “player” ingresa en su rango, dispara un evento que permitir la visualización del botón con icono de lupa, que permite al ser presionado mostrar información más detallada sobre el cuadro actual en el que se encuentra el jugador, mediante su ID de cuadro permite encontrar la información del mismo.

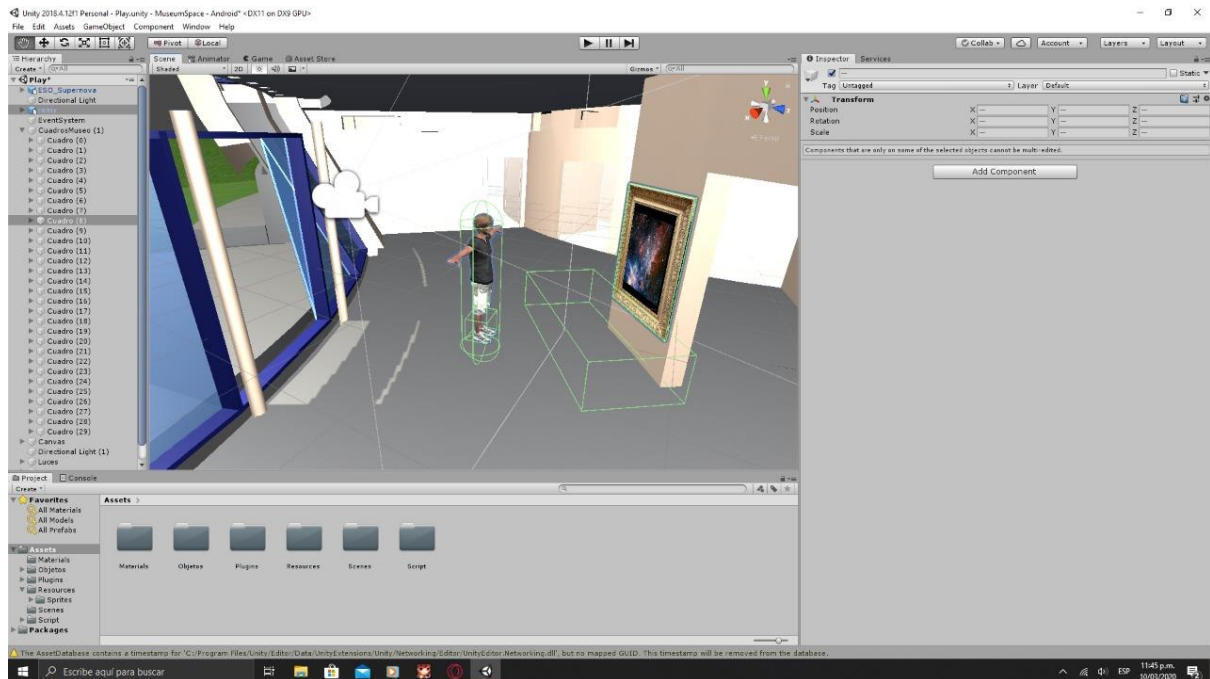


Figura 4. Disparadores de eventos.