**Identificación de características, requerimientos y funcionalidades del framework “Unity ETH UQ” para el proyecto:**

**Desarrollo de un videojuego haciendo uso del framework “Unity ETH UQ”.**

**Dival Mauricio Hoyos Castro**

**Julián David Serna Echeverri**

2016

Facultad de Ingeniería – Programa de Ingeniería de Sistemas y Computación

Universidad del Quindío

**Contenido**

[**Entrevistas** 4](#_Toc460012695)

[**Einer Zapata, Desarrollador** 4](#_Toc460012696)

[**Andrés Herrera, Desarrollador** 4](#_Toc460012697)

[**Andrés Carvajal, Administrador de proyectos** 4](#_Toc460012698)

[**Características del framework** 5](#_Toc460012699)

[**Analytics** 5](#_Toc460012700)

[**EthAnalytics** 5](#_Toc460012701)

[**EthGameAnalytics** 7](#_Toc460012702)

[**EthGoogleAnalytics** 8](#_Toc460012703)

[**Audio** 9](#_Toc460012704)

[**EthAudio** 9](#_Toc460012705)

[**Display** 9](#_Toc460012706)

[**Components** 10](#_Toc460012707)

[**EthButton** 10](#_Toc460012708)

[**EthComboBox** 10](#_Toc460012709)

[**EthComponent** 11](#_Toc460012710)

[**EthComponentManager** 12](#_Toc460012711)

[**EthDialog** 14](#_Toc460012712)

[**EthLabel** 15](#_Toc460012713)

[**EthLine** 15](#_Toc460012714)

[**EthModalWindow** 16](#_Toc460012715)

[**EthProgressBar** 16](#_Toc460012716)

[**EthScroll** 16](#_Toc460012717)

[**EthSlider** 17](#_Toc460012718)

[**EthSprite** 17](#_Toc460012719)

[**EthTabSheet** 18](#_Toc460012720)

[**EthTextField** 18](#_Toc460012721)

[**EthTexture** 18](#_Toc460012722)

[**EthToggleButton** 19](#_Toc460012723)

[**EthToggleButtonGroup** 20](#_Toc460012724)

[**EthVideo** 20](#_Toc460012725)

[**EthComponentAnimator** 20](#_Toc460012726)

[**SqueezeAnimator** 21](#_Toc460012727)

[**StretchAnimator** 21](#_Toc460012728)

[**Components3D** 22](#_Toc460012729)

[**EthButton3D** 22](#_Toc460012730)

[**EthComponent3D** 22](#_Toc460012731)

[**EthScroll3D** 23](#_Toc460012732)

[**EthToggleButton3D** 23](#_Toc460012733)

[**EthToggleButtonGroup3D** 24](#_Toc460012734)

[**Util** 25](#_Toc460012735)

[**EthArguments** 25](#_Toc460012736)

[**EthFileExist** 25](#_Toc460012737)

[**EthInfoRepository** 25](#_Toc460012738)

[**EthLang** 26](#_Toc460012739)

[**EthTimer** 26](#_Toc460012740)

[**EthTimerGo** 26](#_Toc460012741)

[**EthUtil** 27](#_Toc460012742)

[**EthObjectFactory** 29](#_Toc460012743)

[**UrlLoader** 29](#_Toc460012744)

[**Components** 30](#_Toc460012745)

[**EthImage** 30](#_Toc460012746)

[**EthPanel** 30](#_Toc460012747)

[**EthText** 30](#_Toc460012748)

[**EthToggle** 31](#_Toc460012749)

[**EthUiButton** 31](#_Toc460012750)

[**EthUIProgressbar** 31](#_Toc460012751)

[**EthSceneHandler** 32](#_Toc460012752)

[**EthTextMeshOutline** 32](#_Toc460012753)

[**Util** 32](#_Toc460012754)

[**Conclusión funcionalidades** 33](#_Toc460012755)

[**Analytics** 33](#_Toc460012756)

[**Audio** 33](#_Toc460012757)

[**Util** 33](#_Toc460012758)

[**Components** 34](#_Toc460012759)

[**Requerimientos** 35](#_Toc460012760)

# **Entrevistas**

A continuación se plasmarán los aspectos principales que se hablaron con diferentes integrantes de Ethereal Game Factory acerca del framework.

## **Einer Zapata, Desarrollador**

Ethereal Game Factory inicialmente desarrollaba contenidos en flash, para lo cual se creó una librería que reducía el tiempo de desarrollo, pero carecía de documentación, por lo que para nuevos desarrolladores su uso se tornaba complicado, posteriormente la empresa empezó a realizar desarrollos para dispositivos móviles de forma nativa (IOS y Android) hasta que para el proyecto “Run postman run” se decidió utilizar Corona SDK, pero durante el desarrollo surgieron inconvenientes y además constantemente se le agregaban funcionalidades. Por lo que Corona SDK no fue suficiente y el proyecto se migró a Unity 3D, donde para entonces la creación de componentes como botones, scrolls, labels y similares era un proceso difícil, por lo cual teniendo en cuenta su experiencia con la librería para flash, la empresa inicio la creación de un framework para Unity 3D, el cual tenía como objetivos:

* Agilizar el desarrollo de funcionalidades y componentes complicados.
* Garantizar la utilización de código de calidad.
* Permitir el uso de analíticas de uso y marketing.
* Sobrecargar métodos propios de Unity 3D para optimizar su función.

## **Andrés Herrera, Desarrollador**

El framework surgió como solución a un problema específico en el desarrollo de un proyecto, el cual era que la creación de botones y componentes gráficos de GUI era un proceso complicado, especialmente en 2D, pero posteriormente se vuelve de gran utilidad en otros proyectos, no necesariamente videojuegos, por esto la empresa se dedicó a identificar las funcionalidades ya existentes que fueran de gran utilidad para los proyectos, los cuales fueron mejorados y además se agregaron nuevas funcionalidades, hasta que el framework llegó a convertirse en lo que es en la actualidad. Andrés Herrera resalta que en su experiencia con el framework, este le fue de gran utilidad en los siguientes aspectos:

* Agiliza el proceso de aprendizaje de Unity 3D.
* Mejoramiento del tiempo en el cual se cumplen las tareas.
* Las funciones del framework facilitan el desarrollo.

Así mismo resalta que el mayor potencial del framework se encuentra en la creación y manejo de componentes de GUI y la realización de pequeñas animaciones, pero también enfatiza en que la falta de documentación y estándares de programación dificultaban su entendimiento, pero dichas falencias fueron en gran medida mitigadas mediante depuraciones y documentación al framework.

Posterior a la entrevista Andrés Herrera informó al equipo de trabajo que Ethereal GF desarrolló la segunda versión del módulo de analíticas, con lo cual el componente *analytics* quedaba obsoleto y le daba lugar al nuevo componente *appsystem*.

## **Andrés Carvajal, Administrador de proyectos**

Anteriormente la empresa realizaba proyectos sobre la plataforma flash y surgió la necesidad de realizar las cosas de una manera más rápida y cumplir con el principio de programar de manera genérica para reutilizar código. Es por esto que inicialmente se creó un framework para que la creación de botones y elementos similares fuera una tarea más simple para los diseñadores. Cuando la empresa inicio la creación de videojuegos con Unity 3D querían realizar una herramienta que cumpliera el mismo propósito que la anterior, pero está vez se le agregaron funcionalidades como la conexión con Facebook y las analíticas de uso y marketing. Los principales beneficios de utilizar el framework en un desarrollo según Andrés son:

* Realizar funcionalidades de manera más rápida.
* Fácil integración de componentes que pueden enriquecer el juego.
* Fácil solución de errores.
* Buena curva de aprendizaje.

Pero Andrés a la vez resalta que poco a poco el framework está quedando obsoleto debido a que Unity 3D en sus nuevas versiones está incluyendo nuevas funcionalidades en su código nativo, haciendo innecesaria la utilización del framework en dichos aspectos. Por lo cual, en el futuro el framework posiblemente sea sometido a reajustes para remover funcionalidades y potencializar las que aún sean útiles.

Otro aspecto relevante es que a pesar que Unity 3D soporta diversos lenguajes, se utilizó C# debido a que es un lenguaje bastante estructurado, por ende su mantenimiento presenta más facilidad con respecto a los otros lenguajes.

Posterior a la entrevista, Andrés Carvajal informó al equipo de trabajo que debido a que Unity 3D a partir de su versión 5.0 incluyó soporte para componentes UI, Ethereal GF modificó el framework, específicamente el módulo *display*, el cual fue reemplazado por un nuevo módulo llamado *components*.

# **Características del framework**

A continuación se listarán los módulos del framework a ser evaluados con sus respectivos paquetes, clases y métodos con el fin de identificar sus funcionalidades.

## **Analytics**

Este módulo administra las analíticas de la aplicación mediante *Google Analytics* y *Game Analytics.*

#### **EthAnalytics**

Esta clase administra las *Google Analytics*, *Game Analytics* y las analíticas propias.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthAnalytics | Constructor de la clase. |  |  |
| GetInstance | Obtener la instancia actual de la clase, si no existe una, se crea y se retorna. | [EthAnalytics](#EthAnalytics) | EthAnalytics |
| ConfigureGoogleAnalytics | Configurar las Google Analytics para su uso en la aplicación. | [getInstance](#getInstance_EthGoogleAnalytics)  [ConfigureAnalytics](#ConfigureAnalytics_EthGoogleAnalytics) |  |
| ConfigureEthSystem | Configurar e iniciar la instancia de la clase. | [GetInstance](#GetInstance_EthAnalythics) |  |
| ConfigureGameAnalytics | Configurar las Game Analytics para su uso en la aplicación. | [getInstance](#getInstance_EthGameAnalytics) |  |
| GetVariable | Obtener el valor de una variable del sistema. |  | string |
| LogScreen | Registrar una ventana en los sistemas de analíticas. | [LogScreen](#LogScreen_EthGoogleAnalytics)  [LogScreen](#LogScreen_EthGameAnalytics) |  |
| LogEvent | Registrar un evento en los sistemas de analíticas. | [LogEvent](#LogEvent_EthGoogleAnalytics)  [LogEvent](#LogEvent_EthGameAnalytics) |  |
| LogException | Registrar una excepción en los sistemas de analíticas. | [LogException](#LogException_EthGoogleAnalytics)  [LogException](#LogException_EthGameAnalytics) |  |
| LogTiming | Registrar tiempo de uso en los sistemas de analíticas. | [LogTiming](#LogTiming__EthGoogleAnalytics) |  |
| LogSocial | Registrar interacciones de los usuarios mediante la aplicación en redes sociales en los sistemas de analíticas. | [LogSocial](#LogSocial_EthGoogleAnalytics) |  |
| LogTransaction | Registrar transacciones de los usuarios dentro de la aplicación en Google Analytics. | [LogTransaction](#LogTransaction_EthGoogleAnalytics) |  |
| LogTransaction | Registrar transacciones de los usuarios dentro de la aplicación en Game Analytics y el sistema propio. | [LogTransaction](#LogTransaction_EthGameAnalytics) |  |
| LogItem | Registrar la compra de ítems dentro de la aplicación en Google Analytics. | [LogItem](#LogItem_EthGoogleAnalytics) |  |
| LogItem | Registrar la compra de ítems dentro de la aplicación en Game Analytics. | [LogItem](#LogItem_EthGameAnalytics) |  |
| Dispose | Finalizar el seguimiento. | [Dispose](#Dispose_EthGoogleAnalytics) |  |
| ChangeStateVariable | Cambiar el valor de la variable de estado del sistema que indica el nivel actual del jugador. |  |  |
| LogProgression | Registrar el progreso del jugador en la aplicación. | [LogProgression](#LogProgression_EthGameAnalytics) |  |
| ToString | Retornar el nombre de la clase. |  | string |

#### **EthGameAnalytics**

Esta clase adminsitra las *Game Analytics*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthGameAnalytics | Constructor de la clase. |  |  |
| getInstance | Obtener la instancia actual de la clase, si no existe una, se crea y se retorna. | [EthGameAnalytics](#EthGameAnalytics) | EthGameAnalytics |
| LogScreen | Registrar una ventana en el sistema Game Analytics. |  |  |
| LogProgression | Registrar el progreso del jugador en el sistema Game Analytics. |  |  |
| LogEvent | Registrar un evento en el sistema Game Analytics. |  |  |
| LogException | Registrar una excepción en el sistema Game Analytics. |  |  |
| LogTransaction | Registrar una transacción en el sistema Game Analythics. |  |  |
| LogItem | Registrar la compra de ítems en el sistema Game Analythics. |  |  |
| ToString | Retornar el nombre de la clase. |  | string |

#### **EthGoogleAnalytics**

Esta clase adminsitra las *Google Analytics*.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthGoogleAnalytics | Constructor de la clase. |  |  |
| getInstance | Obtener la instancia actual de la clase, si no existe una, se crea y se retorna. | [EthGoogleAnalytics](#EthGoogleAnalytics) | EthGoogleAnalytics |
| ConfigureAnalytics | Configura las Google Analytics para su uso en la aplicación. |  |  |
| LogScreen | Registrar una ventana en el sistema Google Analytics. |  |  |
| LogEvent | Registrar un evento en el sistema Google Analytics. |  |  |
| LogException | Registrar una excepción en el sistema Google Analytics. |  |  |
| LogTiming | Registrar tiempo de uso en el sistema de Google Analytics. |  |  |
| LogSocial | Registrar interacciones de los usuarios mediante la aplicación en redes sociales en el sistema de Google Analytics. |  |  |
| LogTransaction | Registrar transacciones de los usuarios dentro de la aplicación en Google Analytics. |  |  |
| LogItem | Registrar la compra de ítems dentro de la aplicación en Google Analytics. |  |  |
| Dispose | Finalizar el seguimiento en Google Analytics. |  |  |
| ToString | Retornar el nombre de la clase. |  | string |

Este módulo permite llevar un seguimiento del uso y marketing de la aplicación, ya sea en el sistema de analíticas de google, game analytics o un sistema propio.

## **Audio**

Este módulo administra los sonidos en diferentes ambientes de la aplicación*.*

#### **EthAudio**

Esta clase se encarga de administrar los sonidos dentro de la aplicación.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| GetInstance | Obtener la instancia actual de la clase, si no existe una, se crea y se retorna. | [EthAudio](#EthAudio) | EthAudio |
| EthAudio | Constructor de la clase. |  |  |
| PlayMusic | Reproducir un audio ingresando su ruta de ubicación. |  |  |
| PlayEffect | Reproducir un efecto de sonido. | [EthTimer](#EthTimer) |  |
| PlayEffectRepeated | Reproducir un efecto de sonido repetitivamente. |  | AudioSource |
| RemoveEffect | Remover un efecto de la lista de reproducción. |  |  |
| SetMusicVolume | Modificar el volumen de los sonidos. |  |  |
| RefreshPos | Reiniciar el sistema de audio. |  |  |
| SetEffectsVolume | Modificar el volumen de los efectos de sonido. |  |  |
| SetAudioButtonDefault | Modificar el botón por defecto que administra el sonido. |  |  |

Este módulo permite administrar efectos de sonido y música de fondo.

## **Display**

Este módulo administra los componentes que se despliegan en pantalla y sus animaciones. Este módulo cuenta con varios directorios los cuales son: components, components3D e easing. Los cuales se desglosan a continuación:

#### **Components**

Este directorio contiene los componentes principales para elaborar un menú de juego y mostrar elementos en el HUD. A su vez contiene el directorio ComponentAnimator que contiene las animaciones para los componentes.

#### **EthButton**

Esta clase se encarga de administrar el componente botón.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthButton | Constructor de la clase. | [ReloadArguments](#ReloadArguments_EthButton) |  |
| SetOnDragFuction | Evento de arrastre del botón. |  | EthButton |
| SetTexture | Modificar la textura del botón. |  |  |
| ReloadArguments | Recargar los argumentos (EthArguments) del botón. |  |  |
| Draw | Dibujar el botón en pantalla. |  |  |
| regresarAOrigen | Ubicar el botón en su posición inicial. |  |  |
| guardarCoordenadas | Guardar las coordenadas del botón. |  |  |
| arrastrarBoton | Mover el botón hacia donde lo arrastre el cursor. |  |  |
| IsOver | Determinar si el cursor está sobre el botón. |  | bool |
| SetFontColor | Modificar el color de la fuente. |  |  |
| Click | Efectuar la función del botón cuando se le da clic. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al botón. |  | string |

#### **EthComboBox**

Esta clase se encarga de administrar el componente ComboBox.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthComboBox | Constructor de la clase. | [EthButton](#EthButton) |  |
| AddItem | Adicinar un ítem al combo box. |  |  |
| Deploy | Cambiar el estado del combo box entre desplegado o no desplegado. |  |  |
| selectItem | Seleccionar un ítem del combo box. |  |  |
| getSelectedIndex | Obtener el índice del botón seleccionado. |  | int |
| Draw | Mostrar el combo box en la pantalla. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al combo box. |  | string |

#### **EthComponent**

Esta clase es la base de todos los componentes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthComponent | Constructor de la clase. |  |  |
| EthComponent | Constructor de la clase al cual se le indica la pantalla. |  |  |
| EthComponent | Constructor de la clase al cual se le indican la pantalla y los argumentos. |  |  |
| AddAnimator | Agregar una animación al componente. | [setComponent](#SetComponent_EthComponentAnimator)  [StartAnimation](#StartAnimation_EthComponentAnimator) |  |
| ReloadArguments | Recargar los argumentos del componente. |  |  |
| Remove | Eliminar el componente. |  |  |
| Draw | Dibujar el componente en pantalla. |  |  |
| ToBack | Mandar al fondo el componente. |  |  |
| ToFront | Traer el componente al frente. |  |  |
| GetTextAnchor | Obtener la alineación del texto. |  | TextAnchor |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  |  |

#### **EthComponentManager**

Esta clase administra las características de los componentes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthComponentManager | Constructor de la clase. |  |  |
| UseGroup | Definir el uso de un grupo. |  |  |
| CopyScreenValues | Copiar los valores de la pantalla de otro EthComponentManager |  |  |
| Remove | Elimina la instancia. |  |  |
| CenterScreen | Centrar la pantalla con respecto a un objeto. |  |  |
| AddWindow | Agregar una nueva ventana controladora. |  | EthComponentManager |
| AddTabSheet | Agregar un sub menú, lo cual puede ser sin parámetros, con parámetros de tipo string o parámetros de tipo EthArguments. |  | EthTabSheet |
| AddScroll | Agregar un scroll, lo cual puede ser sin parámetros, con parámetros de tipo string o parámetros de tipo EthArguments. | [EthScroll](#EthScroll) | EthScroll |
| EvaluateMinAndMax | Evaluar si hay un nuevo punto mínimo o máximo en pantalla. |  |  |
| ResetMinAndMax | Reiniciar los valores mínimo y máximo de la pantalla. |  |  |
| AddRawTexture | Agregar una textura cruda, puede ser con o sin EthArguments. | [EthTexture](#EthTexture) | EthTexture |
| AddRenderTexture | Agregar una textura renderizada, puede ser con o sin EthArguments. | [EthTexture](#EthTexture) | EthTexture |
| AddLine | Agregar una línea con parámetros de tipo string o tipo EthArguments. | [EthLine](#EthLine) | EthLine |
| AddTexture | Agregar una textura con parámetros de tipo string o EthArguments. | [EthTexture](#EthTexture) | EthTexture |
| AddVideo | Agregar un video con parámetros de tipo string o EthArguments. | [EthVideo](#EthVideo) | EthVideo |
| AddSprite | Agregar un sprite con parámetros de tipo string o EthArguments. | [EthSprite](#EthSprite) | EthSprite |
| AddModalWindow | Agregar una ventana modal. | [EthModalWindow](#EthModalWindow) | EthModalWindow |
| AddDialog | Agregar una ventana de dialogo con parámetros de tipo string o EthArguments. | [EthDialog](#EthDialog) | EthDialog |
| AddSlider | Agregar un slider con parámetros de tipo string o EthArguments. | [EthSlider](#EthSlider) | EthSlider |
| AddProgressBar | Agregar una barra de progreso con parámetros de tipo string o EthArguments. | [EthProgressBar](#EthProgressBar) | EthProgressBar |
| AddButton | Agregar un botón, lo cual puede ser sin parámetros, con parámetros de tipo string o parámetros de tipo EthArguments. | [EthButton](#EthButton) | EthButton |
| AddComboBox | Agregar un combobox con parámetros de tipo string o EthArguments. | [EthComboBox](#EthComboBox) | EthComboBox |
| AddLabel | Agregar un label con parámetros de tipo string o EthArguments. | [EthLabel](#EthLabel) | EthLabel |
| AddTextField | Agregar un campo de texto con parámetros de tipo string o EthArguments. | [EthTextField](#EthTextField) | EthTextField |
| AddToggleButton | Agregar un botón toggle, lo cual puede ser sin parámetros, con parámetros de tipo string o parámetros de tipo EthArguments. | [EthToggleButton](#EthToggleButton) | EthToggleButton |
| RemoveComponent | Eliminar un componente. |  |  |
| RemoveAllComponents | Eliminar todos los componentes. |  |  |
| ToBack | Mover un componente al final de la lista de componentes. |  |  |
| ToFront | Mover un componente al inicio de la lista de componentes |  |  |
| WorldToGUI | Desplazar la vista hasta un objeto en el escenario. |  | Vector2 |
| Draw | Dibujar en pantalla cada uno de los componentes. | [Draw](#Draw_EthComponent) |  |
| GetComponentByName | Obtener un componente por su nombre. |  | EthComponent |
| AddAnimation | Agregar una animación. |  |  |

#### **EthDialog**

Esta clase administra las ventanas de dialogo.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthDialog | Constructor de la clase. | [EthComponentManager](#EthComponentManager)  [AddModalWindow](#AddModalWindow_EthComponentManager)  [AddTexture](#AddTexture_EthComponentManager)  [EthToggleButtonGroup](#EthToggleButtonGroup)  [AddToggleButton](#AddToggleButton_EthComponentManager) |  |
| Draw | Dibujar en pantalla la ventana de dialogo y sus componentes. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |
| addDefaultDialog | Agregar un dialogo al diccionario de diálogos. |  |  |
| useDialog | Crear un EthDialog para mostrar en la ventana modal. | [AddDialog](#AddDialog_EthComponentManager) | EthDialog |

#### **EthLabel**

Esta clase administra las ventanas de dialogo.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthLabel | Constructor de la clase. |  |  |
| Draw | Dibujar en pantalla el label. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthLine**

Esta clase representa una línea divisoria.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthLine | Constructor de la clase. | [ReloadArguments](#ReloadArguments_EthLine) |  |
| ReloadArguments | Recargar los parámetros de la línea. |  |  |
| setOrigin | Modificar el vector de origen. |  | EthLine |
| setTermination | Modificar el vector que indica el final de la línea. |  | EthLine |
| Draw | Dibujar la línea en pantalla. | [DrawLine](#DrawLine_EthLine) |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |
| DrawLine | Dibujar la línea en pantalla. |  |  |

#### **EthModalWindow**

Esta clase administra las ventanas modales.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthModalWindow | Constructor de la clase. | [EthComponentManager](#EthComponentManager) |  |
| Draw | Dibujar la ventana modal en pantalla. |  |  |
| SetTextureBack | Cambiar la textura de fondo. |  |  |
| DoMyWindow | Dibujar la ventana modal sobre el component manager. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |
| Remove | Eliminar la ventana modal. | [RemoveComponent](#RemoveComponent_EthComponentManager) |  |

#### **EthProgressBar**

Esta clase administra las barras de progreso.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthProgressBar | Constructor de la clase. |  |  |
| Draw | Dibujar la barra de progreso en la pantalla. |  |  |
| acotarValor | Acotar los límites inferior y superior respecto al valor de dichas variables. |  |  |
| setValue | Modificar el valor actual del progreso. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthScroll**

Esta clase administra los scroll.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthScroll | Constructor de la clase. | [EthComponentManager](#EthComponentManager) |  |
| Draw | Dibujar el scroll en pantalla. |  |  |
| addScrollDrag | Agregar un vector para efectuar el arrastre. |  |  |
| finishScrollDrag | Finalizar el arrastre y dejar el scroll en la posición final. |  |  |
| reinit | Reiniciar el scroll a su posición inicial. | [RemoveAllComponents](#RemoveAllComponents_EthComponentManager) |  |
| setScrollPosition | Modificar la posición del scroll. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthSlider**

Esta clase administra los slider.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthSlider | Constructor de la clase. |  |  |
| Draw | Dibujar el slider en pantalla. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthSprite**

Esta clase administra los sprite.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthSprite | Constructor de la clase. | [ReloadArguments](#ReloadArguments_EthSprite) |  |
| ReloadArguments | Recargar los parámetros del sprite. |  |  |
| Draw | Dibuja el sprite en pantalla. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthTabSheet**

Esta clase administra los submenús.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthTabSheet | Constructor de la clase. | [EthToggleButtonGroup](#EthToggleButtonGroup) |  |
| AddTab | Agregar un ítem a la tabla. |  |  |
| RefreshPosition | Refrescar la posición de los ítems del menú. |  |  |
| ClickTab | Seleccionar un elemento. |  |  |
| Draw | Dibujar el submenú en pantalla. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthTextField**

Esta clase administra los submenús.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthTextField | Constructor de la clase. |  |  |
| Draw | Dibujar el campo de texto en la pantalla. |  |  |
| setText | Modificar el texto. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthTexture**

Esta clase administra las texturas.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthTexture | Constructor de la clase. | [ReloadArguments](#ReloadArguments_EthTexture) |  |
| ReloadArguments | Recargar los parámetros. |  |  |
| SetRawTexture | Configurar una textura cruda. |  |  |
| SetRenderTexture | Configurar una textura renderizada. |  |  |
| Draw | Dibujar la textura en pantalla. |  |  |
| setTexture | Configurar una textura normal. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  |  |
| IsOver | Definir si el cursor está sobre la textura. |  | bool |

#### **EthToggleButton**

Esta clase administra los botones de palanca.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthToggleButton | Constructor de la clase. | [CreateTextureBackUp](#CreateTextureBackUp_EthToggleButton) |  |
| SetTexture | Modificar la textura del botón. |  |  |
| CreateTextureBackUp | Crear una copia de seguridad de la textura. |  |  |
| Click | Administrar los estados del botón. |  |  |
| Draw | Dibujar el botón en pantalla. |  |  |
| isHover | Determinar si el puntero está sobre el botón. |  | bool |
| unselect | Terminar la animación de selección. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthToggleButtonGroup**

Esta clase administra los grupos de botones de palanca.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthToggleButtonGroup | Constructor de la clase. |  |  |
| AddToggleButton | Agregar un toggle al grupo. |  |  |
| reportClick | Reportar un click en un toggle para para guardar su valor. |  |  |
| GetSelectedButton | Obtener el toggle seleccionado. |  | EthToggleButton |
| GetButton | Obtener un toggle dado su índice. |  | EthToggleButton |
| Length | Obtener la cantidad de toggles en el grupo. |  | int |
| getChildren | Obtener todos los toggles del grupo. |  | List<EthToggleButton> |

#### **EthVideo**

Esta clase administra los videos.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthVideo | Constructor de la clase. | [ReloadArguments](#ReloadArguments_EthVideo) |  |
| ReloadArguments | Recargar los parámetros del video. |  |  |
| PlayVideo | Reproducir el video. |  | IEnumerator |
| Draw | Dibujar el componente en pantalla. |  |  |
| DoMyWindow | Crear la ventana modal sobre la que se reproduce el video. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthComponentAnimator**

Esta es una clase abstracta de la cual heredan las animaciones de los componentes.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthComponentAnimator | Constructor de la clase. |  |  |
| SetComponent | Modificar el componente asignado a la animación. |  |  |
| AnimateComponent | Animar el componente. |  |  |
| StartAnimation | Iniciar la animación |  |  |
| SetToOriginalSizeX | Mover el componente a la coordenada original en el eje X. |  |  |
| SetToOriginalSizeY | Mover el componente a la coordenada original en el eje Y. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **SqueezeAnimator**

Esta es clase administra la animación de apretón a un componente.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| SqueezeAnimator | Constructor de la clase. |  |  |
| SetComponent | Configurar el componente a animar. |  |  |
| StartAnimation | Iniciar la animación de apretón. |  |  |
| AnimateComponent | Animar el componente. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **StretchAnimator**

Esta es clase administra la animación de estirar un componente.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| StretchAnimator | Constructor de la clase. |  |  |
| StartAnimation | Iniciar la animación de estiramiento al componente. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | String. |

#### **Components3D**

Este directorio contiene algunos de los componentes del directorio Components, pero estos son en 3D.

#### **EthButton3D**

Esta clase se encarga de administrar el componente botón en 3D.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthButton3D | Constructor de la clase. |  |  |
| SetArgs | Configurar los argumentos. |  |  |
| SetTexture | Modificar la textura del botón. |  |  |
| mouseEvent | Registrar un evento del mouse sobre el botón. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthComponent3D**

Esta clase es heredada por todos los componentes 3D.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthComponent3D | Constructor de la clase. |  |  |
| AddChild | Agregar un componente 3D como hijo. |  |  |
| getEthParent | Obtener el padre de la clase. |  | EthComponent3D |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthScroll3D**

Esta clase se encarga de administrar el componente scroll en 3D.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthScroll3D | Constructor de la clase. |  |  |
| SetArgs | Configurar los argumentos del scroll. |  |  |
| ClickBot | Mover el botón del scroll cuando el puntero lo arrastra. |  |  |
| Update | Actualizar las coordenadas del scroll, esto se hace una vez por frame. |  |  |
| calculateLimits | Calcular los límites del scroll. |  |  |
| moveScrollY | Mover el scroll en el eje Y. |  |  |
| SetNewPositionY | Asignar una nueva posición en el eje Y al scroll. |  |  |
| ScrollTo | Mover a una posición relativa. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthToggleButton3D**

Esta clase se encarga de administrar los botones toggle en 3D.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthToggleButton3D | Constructor de la clase. |  |  |
| \_click | Modificar la textura cuando se da clic. |  |  |
| \_EthOnMouseUp | Modificar la textura y el estado del botón cuando el clic se suelta. |  |  |
| unselect | Cambiar la textura cuando se des-selecciona el botón. |  |  |
| setEthToggleButtonGroup | Modificar el grupo. |  |  |
| ToString | Retornar el nombre de la clase concatenado con el nombre asignado al componente. |  | string |

#### **EthToggleButtonGroup3D**

Esta clase se encarga de administrar los grupos de botones toggle 3D.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthToggleButtonGroup3D | Constructor de la clase. |  |  |
| AddToggleButton | Agregar un botón toggle al grupo. |  |  |
| ReportClick | Cambiar el estado de uno de los botones toggle. |  |  |
| SelectToggle | Seleccionar un botón dado su índice. |  |  |

Administra los componentes:

* Botón
* Botón de palanca
* Scroll
* Combo box
* Ventana de dialogo
* Etiqueta
* Línea
* Ventana modal
* Barra de progreso
* Slider
* Sub-menú
* Campo de texto
* Textura
* Video

De los cuales los tres primeros también tienen su versión 3D. Además cuenta con las animaciones de apretar o estirar un componente.

## **Util**

Este módulo brinda funcionalidades útiles para la administración de los scripts en el videojuego.

#### **EthArguments**

Esta clase interpreta una cadena de texto con argumentos para un componente y los aplica al mismo.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthArguments | Constructor de la clase. |  |  |
| EthArguments | Constructor de la clase que recibe la cadena que representa los argumentos. |  |  |
| HasArguments | Determina si la cadena si contiene parámetros válidos. |  | bool |
| GetArgumentsAsStr | Retornar los parámetros ordenadamente en una cadena. |  | string |
| ToString | Retorna todos los parámetros existentes en una cadena. |  |  |

#### **EthFileExist**

Esta clase brinda la funcionalidad de conocer si un archivo existe en una ruta determinada.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthFileExist | Consructor de la clase. |  |  |
| TextAssetExist | Verifica si un archivo existe en una URL específica. |  | bool |

#### **EthInfoRepository**

Esta clase administra datos de juego que no serán persistentes una vez el juego sea cerrado.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthInfoRepository | Constructor de la clase. |  |  |
| getInstance | Obtener la instancia de la clase. |  | EthInfoRepository |
| setReg | Modificar un objeto del diccionario de datos dado su nombre (llave). |  |  |
| getReg | Obtener un objeto del diccionario de datos dado su nombre (llave). |  | object |

#### **EthLang**

Esta clase administra los idiomas en la aplicación.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthLang | Constructor de la clase. |  |  |
| GetEntry | Buscar que palabra se retorna analizando el idioma actual. |  | string |
| ActiveLangs | Activar el manejo de idiomas desde un archivo XML previamente creado. |  |  |

#### **EthTimer**

Esta clase administra los temporizadores para ejecutar acciones.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthTimer | Constructor de la clase. |  |  |
| ContinueTimerCurrentScene | Continuar un timer en una nueva escena. |  |  |
| StartTimer | Iniciar el timer. |  |  |
| TimerExecuted | Define si el timer termina su ejecución o continúa. |  |  |
| CancelTimer | Cancelar el timer. |  |  |

#### **EthTimerGo**

Esta clase administra los timers que se siguen ejecutando en otra escena.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| InitTimer | Iniciar el timer. |  |  |
| DestroyTimer | Destruir el timer. |  |  |
| RepeatingMethod | Verificar si el timer se sigue ejecutando o no. |  |  |

#### **EthUtil**

Esta clase administra funciones útiles para la administración de los scripts de la aplicación.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EncodeBase64 | Codificar una cadena en caracteres base 64. |  | string |
| GetClassType | Obtener la clase de un Game Object y lo retorna como texto. |  | string |
| GetFullClassType | Obtener la herarquia completa de la clase de un Game Object. |  | string |
| GetVal | Obtener el primer valor no nulo de un arreglo. |  | object |
| ToArray | Separar una palabra cada que aparezca un separador definido y retornar un arreglo con las partes. |  | string[] |
| ToFloatArray | Separar una palabra cada que aparezca un separador definido y retornar dichas partes en un arreglo float. |  | float[] |
| ToString | Concatenar los elementos de un arreglo determinado utilizando un separador indicado. |  | string |
| ParseStringToArraylist | Separar una cadena por un índice indicado y almacenar cada parte en un array. |  | ArrayList |
| ArraylistToString | Concatenar los elementos de un Array determinado utilizando un separador indicado |  | string |
| PrintValues | Imprimir en consola los elementos de un objeto IEnumarable. |  |  |
| IsMobile | Definir si la aplicación es ejecutada en una plataforma móvil. |  | bool |
| ToInt | Convertir un objeto a entero, si es posible. |  | int |
| ToFloat | Convertir un objeto a flotante, si es posible. |  | float |
| Md5Sum | Encriptar a MD5 una cadena. |  | string |
| GetFormatedNumber | Convertir un entero a string en formato N0 (#,###). |  | string |
| GetNameOfMonth | Obtener el nombre del mes a partir del número que lo identifica. |  | string |
| Log | Mostrar en consola un mensaje. |  |  |
| ShuffleArray | Barajar aleatoriamente los elementos de una lista. |  | List<T> |
| CreateArray | Crear una lista con un número determinado de números consecutivos. |  | List<int> |
| GetInterfaces | Obtener las interfaces asignadas a un Game Object. |  | List<T> |
| IsMeetingProbability | Define mediante un número aleatorio entre 0 y 100 si la probabilidad asignada se cumple o no. |  | bool |
| SetVisibleGameObject | Modifica la visibilidad de un Game Object. |  |  |
| SetAlphaGameObject | Asignar un nivel de transparencia a determinado Game Object. |  |  |
| TintGameObject | Asignar un valor de tinte a determinado Game Object. |  |  |
| CountCharInString | Contar la cantidad de veces que un carácter se repite en una cadena. |  | int |
| GetChildByName | Obtener el hijo de un Game Object dado su nombre. |  | GameObject |

#### **EthObjectFactory**

Esta clase se encarga de la creación y obtención de Game Objects.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| CreateObject | Crear un Gme Object. |  | object |
| GetObject | Obtener un Game Object por su nombre. |  |  |

#### **UrlLoader**

Esta clase se encarga de administrar las URL en la aplicación.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| URLLoader | Constructor de la clase. |  |  |
| GET | Realiza una petición get a una URL ingresada. | [WaitForRequest](#URLLoader_WaitForRequest) | WWW |
| GET | Realiza una petición get a una URL ingresada más los valores de un diccionario de datos. | [WaitForRequest](#URLLoader_WaitForRequest) | WWW |
| POST | Realiza una petición post a una URL ingresada más los valores de un diccionario de datos. | [WaitForRequest](#URLLoader_WaitForRequest) | WWW |
| WaitForRequest | Esperar respuesta de una petición. |  | IEnumerator |
| OnRespEvent | Método delegado para definir el comportamiento al obtener una respuesta a la petición |  |  |

Este módulo brinda facilidades a la hora de administrar el código, brindando funciones como:

* Peticiones URL en la aplicación.
* Manejo de cadenas, listas y arreglos.
* Manejo de Game Objects.
* Manejo de timers.
* Soporte multilenguaje.

## **Components**

En la última versión del framework, comenta Andres Carvajal que los componentes del módulo *display* se manejan mediante un nuevo módulo llamado “components” el cual consta de:

#### **EthImage**

Esta clase se encarga de adminsitrar las propiedades de idioma de los elementos de tipo imagen.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| SetImageLang | Modificar el lenguaje de la imagen. |  |  |
| OnGUI | Mostrar en pantalla la imagen correspondiente al idioma actual. |  |  |

#### **EthPanel**

Esta clase se encarga de administrar las propiedades de los elementos de tipo panel.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| EthPanel | Constructor de la clase. |  |  |
| InitPanel | Iniciar el panel y sus componentes. | [SetTitle](#EthPanel_SetTile)  [SetButtons](#EthPanel_SetButtons)  [SetImage](#EthPanel_SetImage) |  |
| SetTitle | Configurar el título del panel. |  |  |
| SetButtons | Crear los botones ingresados por parámetro en el panel. |  |  |
| AddListener | Agregar el listener al botón del panel. |  |  |
| Destroy | Destruir el panel. |  |  |
| AddImage | Agregar una imagen al panel. |  |  |
| SetImage | Modificar la imagen del panel. | [AddImage](#EthPanel_AddImage) |  |
| ShowLoader | Mostrar un cargador en el panel. |  |  |
| HideLoader | Ocultar el cargador del panel. |  |  |

#### **EthText**

Esta clase administra el soporte de idiomas de los componentes de tipo texto.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| Start | Inicializa el componente y carga el teto en el idioma actual. |  |  |
| OnGUI | Muestra en pantalla el texto en el idioma actual. |  |  |

#### **EthToggle**

Esta clase administra los componentes tipo toggle para soporte de idiomas y audio.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| Start | Inicializar el componente y carga el texto en el idioma actual. | [AddListener](#EthToggle_AddListenes) |  |
| AddListener | Agregar el listener al toggle. |  |  |
| ClickOnButton | Reproducir el audio cuando se da clic en el toggle. |  |  |

#### **EthUiButton**

Esta clase administra los componentes de tipo botón.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| Start | Inicializar el componente y agregar el soporte de idiomas. | [AddListener](#EthUIButton_AddListener) |  |
| AddListener | Agregar el listener al botón. |  |  |
| ClickOnButton | Reproduce el audio cuando se da clic en el botón. |  |  |

#### **EthUIProgressbar**

Esta clase administra los componentes de tipo Scrollbar.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| Start | Inicializar las dos partes de la barra. |  |  |
| Update | Actualizar el estado de la barra. |  |  |
| SetHealth | Modificar el estado de la barra. |  |  |
| AuthorizeAnim | Configurar si se permite la animación. |  |  |

#### **EthSceneHandler**

Esta clase administra las escenas de la aplicación.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| RegisterIGui | Registrar una pantalla para poderla llamar si se hace uso de la tecla de retroceso. |  |  |
| LoadScene | Cargar una escena. |  |  |

#### **EthTextMeshOutline**

Esta clase administra los textos con textura.

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| Start | Inicializar las variables del texto. |  |  |
| LateUpdate | Actualizar el componente una vez por frame. |  |  |
| GetOffset | Obtener el vector de movimiento del objeto. |  |  |

#### **Util**

Esta clase

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| GetRand | Obtener un número entero aleatoriamente entre cero y un número dado por parámetro. |  | int |
| ToggleSound | Activar o desactivar los sonidos. |  | bool |
| GetSystemLanguage | Obtener el lenguaje del sistema. |  | string |
| GetChildByName | Obtener un objeto hijo dado su padre y el nombre. |  | GameObject |
| DestroyChildren | Destruir todos los hijos de un Game Object. |  |  |
| Angle | Hallar el ángulo formado por un vector. |  | float |
| PlayLoop | Reproducir un sonido en forma cíclica. |  |  |
| PlayEffect | Reproducir un efecto sonoro. |  |  |
| PlayLoopEffectTiedToObject | Reproducir un efecto atado a un objeto. |  |  |
| RefreshAudioPos | Reiniciar el controlador de audio. |  |  |

## **AppSystem**

Este módulo reemplaza el de analytics y consta de:

#### **EthAppSystem**

|  |  |  |  |
| --- | --- | --- | --- |
| **Método** | **Funcionalidad** | **Métodos llamados** | **Retorno** |
| GetInstance | Obtener la instancia de la clase, si no la hay, crea una. |  | EthAppsSystem |
| Init | Método para inicializar la conexión con el sistema de analíticas. | [CheckInitialInfo](#CheckInitialInfo) |  |
| GetIdDownload | Obtener el identificador de descarga generado por el sistema de analíticas para un dispositivo. | [RespGetIdDownload](#RespGatIdDownload) |  |
| Log | Registrar un evento en el sistema de analíticas. | [ReportLog](#ReportLog) |  |
| LogScreen | Registrar una pantalla en el sistema de analíticas. | [ReportScreen](#ReportScreen) |  |
| GetVariable | Obtener el valor de una variable desde el sistema de analíticas. |  |  |
| ChangeStateVariable | Cambiar el valor de una variable de estado en el sistema de analíticas. | [ReportStateVariable](#ReportStateVariable) |  |
| RespGetIdDownload | Respuesta que se genera cuando se obtiene el identificador de descarga. |  |  |
| SendReady | Establecer si el sistema de analíticas está listo. |  |  |
| GetIdSession | Obtener el identificador de sesión. |  |  |
| RefreshVariables | Refrescar el valor de las variables en el sistema de analíticas. |  |  |
| ReportLog | Registrar un evento en la base de datos del sistema de analíticas. |  |  |
| ReportScreen | Registrar una pantalla en la base de datos del sistema de analíticas. |  |  |
| RespScreen | Respuesta que se genera cuando se reporta una pantalla. |  |  |
| RespLog | Respuesta que se genera cuando se reporta un evento. |  |  |
| ReportStateVariable | Registrar el cambio del valor de una variable de estado en la base de datos del sistema de analíticas. | [RespState](#RespState) |  |
| RespState | Registrar el estado de un cambio en la base de datos del sistema de analíticas. |  |  |
| RespGetIdSesion | Respuesta que se genera cuando se obtiene el identificador de sesión. |  |  |
| RespGetVariables | Respuesta que se genera cuando se obtiene el valor de una variable. |  |  |
| CheckInitialInfo | Verificar si la sesión ya tiene un identificador de descarga. |  |  |
| LoadVariables | Cargar las variables del dispositivo en el sistema. |  |  |
| GetVariableFromDictionary | Obtener una variable registrada en el diccionario del sistema. |  | string |
| ToString | Retornar el nombre de la clase. |  | string |

# **Conclusión funcionalidades**

A continuación se listan las funcionalidades identificadas para cada módulo a partir del análisis realizado en la sección anterior:

## **Analytics**

* Inicializar el sistema de analíticas.
* Registrar un evento en el sistema de analíticas.
* Registrar una pantalla en el sistema de analíticas.
* Obtener y modificar variables registradas en el sistema de analíticas.

## **Audio**

* Reproducir un audio
* Reproducir un efecto
* Reproducir un efecto de forma cíclica
* Modificar el volumen de los sonidos
* Modificar el volumen de los efectos

## **Util**

* Obtener una palabra en el idioma actual desde el diccionario
* Manejo de timer
* Definir si la plataforma en que se ejecuta la aplicación es móvil
* Modificar la visibilidad de un Game Object
* Contar la cantidad de veces que se repite un carácter en una cadena
* Obtener un Game Object hijo a partir de su padre y su nombre

## **Components**

* El componente imagen se basa en la imagen por defecto de Unity 3D y agrega la funcionalidad de:
  + Soporte de idiomas
* El componente texto se basa en el texto por defecto de Unity 3D, y agrega la funcionalidad de:
  + Soporte de idiomas
* El componente toggle se basa en el toggle por defecto de Unity 3D y agrega las funcionalidades de:
  + Reproducción de sonido
* El componente botón se basa en el botón por defecto de Unity 3D y agrega las funcionalidades de:
  + Soporte de idiomas
  + Reproducción de sonido
* El componente barra de progreso tiene las siguientes funcionalidades:
  + Modificar el valor de progreso
  + Animar al actualizar el valor
* El manejador de escenas permite:
  + Cargar una escena
  + Al cargar una escena crear un panel sobre el cual se muestra un cargador
* El manejador de textos con texturas permite:
  + Rellenar de color un textmesh de unity
* Este componente tiene nuevas funcionalidades:
  + Obtener numero aleatorio
  + Obtener el lenguaje del sistema

# **Requerimientos**

La versión del framework que Ethereal GF brindó fue construida sobre la versión 5.1.1f1 de Unity 3D, por lo cual, para evitar posibles problemas de compatibilidad de algunas funciones al importar el framework, se recomienda que el desarrollo del videojuego sea sobre la misma versión. Los demás aspectos técnicos para la realización del proyecto quedan a disposición del equipo de trabajo y la compatibilidad de unity 3D.