



industriales
etsii

Escuela Técnica
Superior
de Ingeniería
Industrial

UNIVERSIDAD POLITÉCNICA DE CARTAGENA

Escuela Técnica Superior de Ingeniería
Industrial

“Realidad virtual en el entorno de una carretilla de manutención de cargas, aplicación en prevención de riesgos laborales”

TRABAJO FIN DE GRADO

GRADO EN TECNOLOGÍAS INDUSTRIALES

Autor: Eduardo Paños Córdoba
Director: Dolores Ojados Fernández
Codirector: Isidro J. Ibarra Berrocal

Cartagena, septiembre de 2016



Universidad
Politécnica
de Cartagena

Índice

1.	Introducción	7
1.1.	Realidad Virtual	7
1.2.	Seguridad Industrial y prevención de riesgos laborales.	8
1.2.1.	Carretillas elevadoras.	9
1.2.1.1.	Triángulo de estabilidad.	12
1.2.1.2.	Centro de gravedad combinado	12
1.3.	Normativa aplicable.	13
1.3.1.	Ley 31/1995, de 8 de noviembre, de Prevención de Riesgos Laborales. 14	
1.3.2.	Real Decreto 1215/1997. Disposiciones mínimas de seguridad y salud para la utilización de los equipos de trabajo.	14
2.	Objetivos.	15
3.	Material y métodos.	15
3.1.	Solidworks 2014	15
3.1.1.	Módulos de Solidworks.	16
3.1.1.1.	Módulo de pieza.	16
3.1.1.2.	Módulo de ensamblaje	16
3.1.1.3.	Módulo de plano o dibujo.	16
3.2.	Autodesk 3D Studio Max.	16
3.2.1.	Exportación de modelos.	17
3.3.	Unity.	18
3.4.	MonoDevelop.	22
3.4.1.	Scripts.	23
3.4.1.1.	Características de un script.	23
3.5.	Audacity.	24
3.5.1.	Características de Audacity.	25
3.6.	Carretilla elevadora.	25
3.7.	Lugar de trabajo.	27
3.7.1.	Nave Industrial.	27
3.7.2.	Estanterías y carga.	29
3.8.	Sala de realidad virtual.	31
3.8.1.	Plataforma	32
3.8.2.	Oculus Rift.	33
4.	Resultados.	35
4.1	Escena de realidad virtual.	35
4.1.1.	Carretilla elevadora.	36
4.1.1.1.	Motor.	37
4.1.1.2.	Movimiento de Horquilla.	42
4.1.1.3.	Movimiento del mástil.	44
4.1.1.4.	Inclinación de mástil y horquilla.	46
4.1.1.5.	Centro de gravedad.	47
4.1.1.6.	Oculus Rift.	49
4.1.1.7.	Plataforma.	50
4.1.2.	Lugar de trabajo.	50
4.1.2.1.	Nave industrial.	50
4.1.2.2.	Estanterías.	51

4.1.3.Edición de audio.....	51
4.1.3.1.Audio del motor de carretilla.....	51
4.1.3.2.Audio de marcha atrás de carretilla.....	53
4.1.3.3.Audio de horquilla y mástil de la carretilla elevadora.....	54
4.1.4.Escenario.....	55
4.1.4.1.Nave industrial.....	55
4.1.4.2.Carretera.....	56
4.1.4.3.Muro.....	56
4.1.4.4.Cielo.....	57
4.1.4.5.Iluminación.....	57
4.1.4.6.Cámara.....	58
4.2. Ensayos mecánicos.....	59
4.2.1.Vuelco frontal.....	60
4.2.1.1.Carga de 2500 Kg.....	60
4.2.1.2.Carga de 5000 Kg.....	61
4.2.1.3.Carga de 8000 Kg.....	63
4.2.2.Vuelco lateral.....	64
4.2.2.1.Carga de 2500 Kg.....	65
4.2.2.2.Carga de 5000 Kg.....	66
4.2.2.3.Carga de 8000 Kg.....	67
5. Conclusiones.....	68
6. Bibliografía.....	70
7. Anexo I.....	71

Índice de figuras

Fig. 1 Carretilla elevadora. 1927	10
Fig. 2 Principales factores de riesgo al manipular una carretilla elevadora	10
Fig. 3. Evolución de mercado de las carretillas en España	12
Fig. 4. Triángulo de estabilidad	12
Fig. 5. Centro de gravedad combinado	13
Fig. 6. Exportador de fbx.....	17
Fig. 7. Interfaz principal de Unity	19
Fig. 8. Ventana de proyecto	19
Fig. 9. Vista de la escena.....	20
Fig. 10. Ventana de jerarquía.....	20
Fig. 11. Ventana de inspector	21
Fig. 12. Barra de herramientas	21
Fig. 13. Asset store	22
Fig. 14. Interfaz principal de MonoDevelop	22
Fig. 15. Creación de script	23
Fig. 16. Funciones principales de un script	24
Fig. 17. Interfaz principal de Audacity	25
Fig. 18. Caterpillar DP70.....	26
Fig. 19. Guías para la horquilla	26
Fig. 20. Modelo 3D carretilla elevadora texturizada	27
Fig. 21. Modelo 3D nave industrial texturizada.....	28
Fig. 22. Anchura de pasillos de la nave industrial en función a la normativa vigente...	28
Fig. 23. Zona de carga y descarga.....	29
Fig. 24. Estanterías metálicas texturizadas	29
Fig. 25. Carga de 2500Kg	30
Fig. 26. Carga de 5000kg	30
Fig. 27. Carga de 8000Kg.....	30
Fig. 28. Composición estanterías y carga en el interior de la nave industrial	31
Fig. 29. Sala de realidad virtual.....	32
Fig. 30. Plataforma móvil	32
Fig. 31. Motores de la plataforma móvil	33
Fig. 32. Oculus Rift	34
Fig. 33. Plano del suelo	35
Fig. 34. Colliders de la carretilla elevadora	36
Fig. 35. WheelColliders en neumáticos.....	36
Fig. 36. RigidBody carretilla elevadora	37
Fig. 37. GameObjects que componen la carretilla elevadora	37
Fig. 38. Declaración de variables en script Motor.....	38
Fig. 39. Giro ruedas traseras	39
Fig. 40. Aceleración y freno en script Motor	39
Fig. 41. Función para el cambio de marchas en script Motor	40
Fig. 42. Audio en script Motor	40
Fig. 43. GUIs en script Motor	41
Fig. 44. GUIs implementados en la escena.....	41
Fig. 45. Ventana de inspector del GameObject chasis y ruedas	42
Fig. 46. Colliders de la horquilla.....	43

Fig. 47. Elevación y descenso en script de horquilla	43
Fig. 48. Audio en script de horquilla	44
Fig. 49. Ventana de inspector GameObject horquilla	44
Fig. 50. Colliders en mástil.....	45
Fig. 51. Elevación y descenso en script de mástil	45
Fig. 52. Ventana de inspector de mástil	46
Fig. 53. Inclinación conjunto mástil y horquilla hacia atrás	47
Fig. 54. Centro de gravedad de la carretilla elevadora en ausencia de manipulación de carga	48
Fig. 55. Colliders aplicados a las cargas	48
Fig. 56. Declaración de Rigidbodies en script de centro de gravedad	49
Fig. 57. Ventana de inspector del GameObject asociado a Oculus Rift	49
Fig. 58. Ventana de inspector del GameObject asociado a la carretilla incluyendo el script de la plataforma.....	50
Fig. 59. Colliders en la nave industrial	50
Fig. 60. Colliders en las estanterías metálicas	51
Fig. 61. Audio original del motor	52
Fig. 62. Audio modificado del motor.....	52
Fig. 63. Audio original de marcha atrás.....	53
Fig. 64. Audio modificado de marcha atrás	53
Fig. 65. Audio original horquilla y mástil.....	54
Fig. 66. Audio modificado horquilla y mástil	54
Fig. 67. Aspecto final de la nave industrial	55
Fig. 68. Composición de naves industriales	55
Fig. 69. Carretera.....	56
Fig. 70. Muro que delimita la escena	56
Fig. 71. Miniatura de textura de ladrillo visto	56
Fig. 72. Contenido de Cloud System en ventana de proyecto.....	57
Fig. 73. Cielo insertado en la escena de realidad virtual	57
Fig. 74. Ventana de inspector de la fuente de iluminación	58
Fig. 75. Emplazamiento de la fuente de iluminación en la escena	58
Fig. 76. Emplazamiento de la cámara ligada a Oculus Rift	59
Fig. 77. Aspecto final de la escena de realidad virtual inmersiva	59
Fig. 78. Emplazamiento de la horquilla para los ensayos a realizar	60
Fig. 79. Carga de 2500 Kg situada a 4 m del suelo.....	60
Fig. 80. Gráfica asociada al ensayo de 2500 Kg en referencia al vuelco frontal.....	61
Fig. 81. Carga de 5000 Kg situada a 3 m del suelo.....	62
Fig. 82. Gráfica asociada al ensayo de 5000 Kg en referencia al vuelco frontal.....	62
Fig. 83. Carga de 8000 Kg situada a 2 m del suelo.....	63
Fig. 84. Gráfica asociada al ensayo de 8000 Kg en referencia al vuelco frontal.....	64
Fig. 85. Inicio del vuelco frontal.....	64
Fig. 86. Carga de 2500 Kg situada a 1 m del suelo.....	65
Fig. 87. Gráfica asociada al ensayo de 2500 Kg en referencia al vuelco lateral.....	66
Fig. 88. Carga de 5000 Kg situada a 4 m del suelo.....	66
Fig. 89. Gráfica asociada al ensayo de 5000 Kg en referencia al vuelco lateral.....	67
Fig. 90. Situación final ante el vuelco.....	68

Índice de tablas

Tabla 1. Comparativa accidentes laborales año 2014-2015.....	11
Tabla 2. Resultados del ensayo de 2500 Kg en referencia al vuelco frontal.....	61
Tabla 3. Resultados del ensayo de 5000 Kg en referencia al vuelco frontal.....	62
Tabla 4. Resultados del ensayo de 8000 Kg en referencia al vuelco frontal.....	63
Tabla 5. Resultados del ensayo de 2500 Kg en referencia al vuelco lateral.....	65
Tabla 6. Resultados del ensayo de 5000 Kg en referencia al vuelco lateral.....	67

1. Introducción

El diseño y creación de escenas de realidad virtual inmersiva ha crecido exponencialmente en los últimos años, convirtiéndose en una herramienta muy útil en la simulación de sistemas mecánicos. Como consecuencia de su utilización se consigue una mayor interacción en tiempo real entre el sistema mecánico en estudio, el entorno y el usuario que está inmerso en la escena. Durante la simulación de la escena se pueden detectar errores de funcionamiento de la máquina y posibles actuaciones erróneas del operario, corrigiéndolas y evitando así que estos problemas aparezcan posteriormente en la vida real, con lo que se pueden evitar accidentes y conseguir un importante ahorro económico.

Una de las principales aplicaciones de la simulación de sistemas mecánicos en el entorno de una escena de realidad virtual inmersiva es la educación y formación. Este tipo de entornos facilitan enormemente la visualización, comprensión e interacción entre los diferentes sistemas mecánicos y su funcionamiento, fomentando el aprendizaje y dando lugar a un entorno más seguro y sin riesgo para el usuario.

En este proyecto se ha realizado el diseño y creación de una escena de realidad virtual inmersiva en la que se pueden experimentar, en primera persona, los riesgos más importantes que conlleva el manejo de carretillas elevadoras en la industria. Para su desarrollo, se han utilizado programas de diseño como Solidworks para modelado y ensamblaje, 3D Studio Max para poder exportar los modelos, y Unity, para elaborar la escena de realidad virtual inmersiva, establecer la interacción entre los diferentes sistemas que la componen, texturizar y realizar la simulación.

1.1. Realidad Virtual

La realidad virtual consiste en crear una serie de entornos o escenas mediante el uso de softwares de diseño, generando en el usuario la sensación de estar inmerso en el entorno, todo ello en tiempo real.

Esta tecnología comenzó a desarrollarse en la década de los años 60, pero sólo hasta tiempos recientes se han propuesto dispositivos asequibles para el público en general, lo que ha producido que una nueva industria relacionada a la realidad virtual se esté potenciando a gran velocidad.

La forma en que el ser humano construye la noción de “lo real” se lleva a cabo a través de la percepción, es decir, a través de los sentidos. Por tanto, es necesario el uso de accesorios periféricos para orientar los sentidos a lo que sucede en el mundo virtual que se experimenta, lo cual produce sensación de realidad.

La realidad virtual puede ser de dos tipos: inmersiva y no inmersiva. Los métodos inmersivos de realidad virtual se ligan a un ambiente tridimensional creado por un ordenador, el cual se manipula a través de hardwares adicionales tales como cascos, guantes u otros dispositivos que capturan la posición y rotación de diferentes partes del cuerpo humano.

La realidad virtual no inmersiva también utiliza el ordenador y se vale de medios como el que actualmente ofrece Internet, en el cual se puede interactuar en tiempo real con diferentes personas en espacios y ambientes que en realidad no existen sin la necesidad de utilizar hardware adicional.

La realidad virtual se puede llegar a convertir en una herramienta de enorme utilidad en el campo de la seguridad. Con ayuda de las nuevas tecnologías se logra el desarrollo de herramientas basados en gráficos 3D interactivos y técnicas de realidad virtual que permiten crear simuladores de máquinas o procesos industriales de cualquier sector y tipología. Esto hace que se puedan proporcionar:

- Entrenamiento de operarios para nuevas líneas de producción que aún no están instaladas u operativas.
- Medición de curvas de aprendizaje.
- Entrenamiento en zonas con riesgo de accidente laboral.
- Corrección de errores tempranos en la fase de diseño.
- Incrementar la seguridad del operario en espacios de trabajo con riesgo de accidente laboral.
- Estudios ergonómicos.

El objetivo de los proyectos que se realizan alrededor de estas tecnologías es que su utilización, por parte de las empresas, proporciona una serie de ventajas competitivas, de carácter económico, ya que son mucho más baratas que las reales y permiten entre otras aplicaciones, poner a prueba productos o procesos de forma virtual antes de su construcción. Con esto se consigue ahorrar costes, evitar riesgos físicos en la formación de un operario al interactuar con una máquina compleja o simular sistemas de protección ante catástrofes permitiendo entrenar con un coste mucho menor que el despliegue de una simulación real. Al mismo tiempo, permiten trabajar en un entorno virtual colaborativo con información técnica compleja, como la de los procesos industriales, como si de un entorno real se tratase y comprobar su funcionamiento, simplificando la comprensión de grandes cantidades de información compleja.

1.2. Seguridad Industrial y prevención de riesgos laborales.

La seguridad industrial es un área multidisciplinaria que se encarga de minimizar los riesgos en la industria. Parte del supuesto de que toda actividad industrial tiene peligros inherentes que necesitan de una correcta gestión.

La seguridad industrial, por lo tanto, requiere de la protección de los trabajadores, la implementación de controles técnicos y la formación vinculada al control de riesgos. En concreto, podemos establecer que a la hora de hablar de la seguridad industrial se hace necesario especificar que la misma se desarrolla de manera específica para poder prevenir las posibles situaciones y riesgos que se den en ámbitos donde se trabaja con instalaciones frigoríficas, electricidad, manutención de maquinaria o equipos a presión. Cabe destacar que la seguridad industrial siempre es relativa, ya que siendo éste su objetivo último, es imposible garantizar que nunca se producirá

ningún tipo de accidente. Aún así, su misión principal es trabajar para prevenir los siniestros.

Por todo ello es importante establecer lo que se denomina como prevención de riesgos laborales. Se trata de un servicio y una serie de actuaciones que lo que intentan es dotar a los trabajadores de los conocimientos y habilidades que necesitan para poder acometer tareas que puedan no sólo evitar que sufran determinados peligros, accidentes y enfermedades en su puesto de trabajo sino también que estén capacitados para poder hacer frente a todos aquellos en el caso de que aparezcan.

La innovación tecnológica, el recambio de maquinarias, la capacitación de los trabajadores y los controles habituales son algunas de las actividades que provocan una mejora considerable en lo que a seguridad industrial se refiere. Una de las innovaciones tecnológicas que se están implementando para aumentar la seguridad industrial son los simuladores, es decir, escenarios de realidad virtual inmersivas.

Los simuladores de conducción, por ejemplo, permiten a los conductores practicar sus habilidades en las situaciones de riesgo más habituales al volante: distracciones, inclemencias climatológicas, obstáculos imprevistos, etc. La formación e información de los trabajadores son herramientas indispensables para controlar los riesgos laborales y tienen un efecto directo sobre la reducción de accidentes.

Con un simulador de prevención de riesgos laborales, el conductor adquiere confianza y seguridad al identificar los riesgos a los que se puede exponer durante su jornada laboral a manos del volante.

1.2.1. Carretillas elevadoras.

Se denominan carretillas automotoras de manutención o elevadoras, todas las máquinas que se desplazan por el suelo, de tracción motorizada, destinadas fundamentalmente a transportar, empujar, tirar o levantar cargas. Pueden ser eléctricas o con motor de combustión interna.

En 1851 Waterman ideó el primer montacargas de la historia. Se trataba de una plataforma unida a un cable. Fue en 1915 cuando surgieron las primeras carretillas capaces de desplazar la carga tanto en horizontal como en vertical. En la época de la Primera Guerra Mundial se diseñó una plataforma que podía subir y bajar las mercancías gracias a un mecanismo de elevación de potencia, pero, sin embargo, fue a Clark en 1917 al que se le ocurrió la idea de que el operario trabajara sentado en la propia carretilla elevadora. Sin embargo, no fue hasta comienzos de 1960 cuando empezó a preocupar la seguridad de los trabajadores y se empezaron a introducir jaulas o estructuras de protección, pues hasta entonces las carretillas eran abiertas.

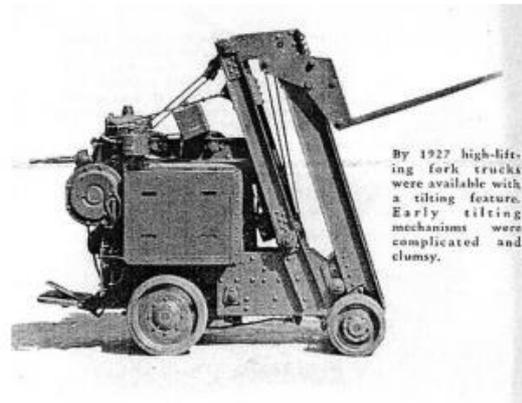


Fig. 1 Carretilla elevadora. 1927

Los principales factores de riesgo al manipular una carretilla elevadora son cinco:

- Carretilla elevadora.
- Carretillero.
- Lugar de trabajo.
- Organización del trabajo.
- Mercancía.



Fig. 2 Principales factores de riesgo al manipular una carretilla elevadora

Los riesgos laborales más importantes que se pueden ocasionar debido a estos cinco factores en el entorno de la manipulación de una carretilla elevadora se enuncian a continuación.

- Vuelco de la carretilla. Es el principal riesgo al manipular una carretilla elevadora, que puede ser ocasionado por circular a una velocidad excesiva o con una carga excesiva. Puede producir la caída del conductor en marcha si no existe o no se utiliza el cinturón de seguridad, pudiendo sufrir graves lesiones por atrapamientos por partes de la carretilla volcada.
- Caída de carga en manipulación, debido a cargas mal apiladas, rotura de palés, circular con la carga elevada o circular a velocidad excesiva.
- Choque contra objetos inmóviles, móviles o atropellos, debido a falta de

visibilidad, falta de iluminación, circular con carga elevada, distracciones o impericia en la conducción.

- Ruido, producido por un mantenimiento inadecuado de las partes que generan ruido tales como el motor o mal aislamiento en el puesto de conducción.
- Vibraciones, producidas por una superficie en mal estado, volante o asiento no ergonómicos.

Durante el año 2015, en el sector del Transporte y Almacenamiento, hubo 36.325 accidentes de trabajo, un 8,2% más que en el 2014 (33.497), según refleja el último avance de la Estadística de Accidentes de Trabajo publicado por el Ministerio de Empleo y Seguridad Social.

Por grupos de ocupación, 33.319 conductores y operadores de maquinaria móvil sufrieron accidentes en el 2015. Además, 19.847 accidentados conducían un medio de transporte o equipo de carga con motor durante su jornada.

Tabla 1. Comparativa accidentes laborales año 2014-2015

ACCIDENTES DE TRABAJO

ATR-A2. ACCIDENTES DE TRABAJO CON BAJA, EN JORNADA E IN ITINERE, SEGÚN GRAVEDAD, POR SECTOR, SECCIÓN Y DIVISIÓN DE ACTIVIDAD ECONÓMICA

Avance Enero - Diciembre 2015

	EN JORNADA				IN ITINERE			
	Total	Leves	Graves	Mortales	Total	Leves	Graves	Mortales
Transporte y almacenamiento	32.740	32.344	309	87	3.582	3.520	54	8
Transporte terrestre	17.790	17.483	227	80	1.554	1.514	33	7
Transporte marítimo	500	491	8	1	37	34	3	-
Transporte aéreo	1.382	1.376	5	1	255	253	2	-
Almacenamiento	9.384	9.330	49	5	1.226	1.212	13	1
Actividades postales	3.684	3.664	20	-	510	507	3	-

ACCIDENTES DE TRABAJO

ATR

ATR-A.1.2. ACCIDENTES DE TRABAJO CON BAJA, EN JORNADA E IN ITINERE, SEGÚN GRAVEDAD, POR SECTOR, SECCIÓN Y DIVISIÓN DE ACTIVIDAD ECONÓMICA

Año 2014

	EN JORNADA				IN ITINERE			
	Total	Leves	Graves	Mortales	Total	Leves	Graves	Mortales
Transporte y almacenamiento	30.249	29.856	309	84	3.257	3.185	57	15
Transporte terrestre	16.559	16.252	242	65	1.445	1.400	35	10
Transporte marítimo	545	535	9	1	36	36	-	-
Transporte aéreo	1.340	1.330	2	8	227	223	4	-
Almacenamiento	8.373	8.322	42	9	1.094	1.077	15	2
Actividades postales	3.432	3.417	14	1	455	449	3	3

El incremento de las ventas de carretillas elevadoras experimentado en los últimos años, sumado al aumento del empleo precario sin la formación y las competencias necesarias y adecuadas, el empeoramiento de las condiciones de trabajo propiciado por la crisis económica y la reforma laboral, son un factor de riesgo que ha incrementado la siniestralidad laboral.



Fig. 3. Evolución de mercado de las carretillas en España

1.2.1.1. Triángulo de estabilidad.

Una carretilla elevadora se comporta, sea cual sea el modelo, como si tuviera dos ruedas delanteras y solo una rueda trasera, lo que hace que aumente su inestabilidad. Esto se debe a que el eje de dirección de la carretilla se encuentra unido al chasis solo por su parte central, formando con las ruedas delanteras un triángulo imaginario al que se le denomina, triángulo de estabilidad.

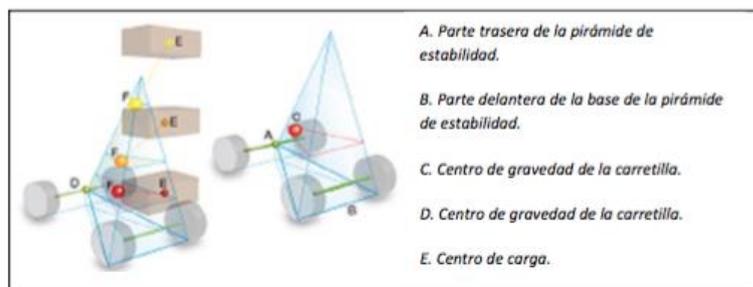


Fig. 4. Triángulo de estabilidad

Un accidente por vuelco puede darse en muchas situaciones, aunque en todos los casos la estabilidad de la carretilla falla por una mala conducción o por no entender los principios básicos de dicha estabilidad. El principal problema, según reconocen los organismos de Salud y Seguridad, es que no se emplea suficiente tiempo de formación para explicar y llegar a entender este concepto.

1.2.1.2. Centro de gravedad combinado

El punto que debe mantenerse dentro del triángulo de estabilidad es el centro de gravedad combinado. La carretilla tiene un centro de gravedad propio,

que normalmente está a 20 cm por debajo del asiento del conductor. La carga también tiene un centro de gravedad, el centro de carga, que suele estar a 500 mm por delante del respaldo de horquillas. Estas dos fuerzas son las que componen el centro de gravedad combinado: un nuevo centro de gravedad en movimiento que debe permanecer dentro de la pirámide de estabilidad. Si este punto se traslada más allá de la pirámide por la parte delantera, la carretilla volcará frontalmente. Si se sale de la pirámide por la izquierda o la derecha, la carretilla volcará lateralmente.

Cuando la carretilla está parada, el centro de gravedad combinado se encuentra en una línea que va desde el centro de gravedad de la carretilla al centro de carga. Si la carga se eleva, el centro de gravedad combinado asciende con ella hasta formar triángulos cada vez más pequeños.

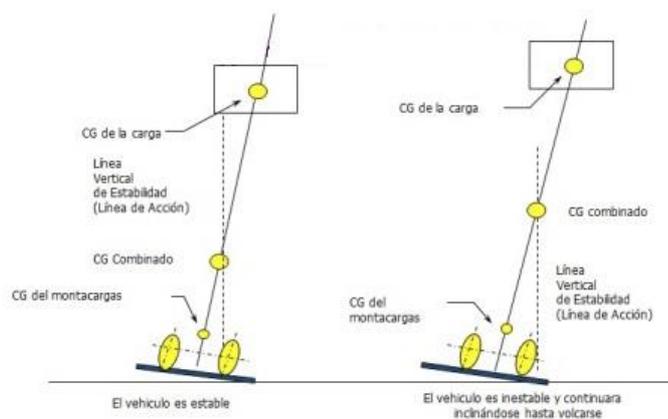


Fig. 5. Centro de gravedad combinado

Debido a que la pirámide se va estrechando hacia arriba, las cargas que son muy estables a baja altura se vuelven inestables al elevarlas, ya que el centro de gravedad combinado no debe salirse de un triángulo que es cada vez más pequeño. En grandes alturas, este triángulo puede ser realmente pequeño, lo que hace que se tenga un margen de error mínimo.

El centro de gravedad no es la única fuerza operante. Con la carretilla en movimiento, hay fuerzas adicionales causadas por la aceleración y desaceleración. Incluso una frenada bastante suave puede hacer que el centro de gravedad combinado se mueva hacia adelante fuera del triángulo y crear un riesgo de vuelco.

1.3. Normativa aplicable.

Existe un amplio número de leyes y normas que afectan a la manipulación de carretillas elevadoras y a los sistemas de almacenamiento. Se nombran a continuación aquellas que se consideran relevantes para el presente proyecto:

1.3.1. Ley 31/1995, de 8 de noviembre, de Prevención de Riesgos Laborales.

Artículo 18. El empresario debe adoptar las medidas adecuadas para que los trabajadores reciban todas las informaciones necesarias en relación con los riesgos para la seguridad y la salud de los trabajadores y en relación a las medidas y actividades de protección y prevención aplicables a los riesgos señalados en el apartado anterior.

Artículo 19. El empresario deberá garantizar que cada trabajador reciba una formación teórica y práctica suficiente y adecuada en materia preventiva, tanto en el momento de su contratación, cualquiera que sea la modalidad o duración de ésta, como cuando se produzcan cambios en las funciones que desempeñe, se introduzcan nuevas tecnologías o cambios en los equipos de trabajo.

1.3.2. Real Decreto 1215/1997. Disposiciones mínimas de seguridad y salud para la utilización de los equipos de trabajo.

Artículo 3. El empresario debe adoptar las medidas necesarias para que los equipos de trabajo que se pongan a disposición de los trabajadores sean adecuados al trabajo que deba realizarse, de forma que garanticen la seguridad y la salud de los trabajadores al utilizar dichos equipos de trabajo. Cuando a fin de evitar o controlar un riesgo, deba usarse un equipo de trabajo bajo unas condiciones que requieran un particular conocimiento, el empresario adoptará las medidas necesarias para que el uso del equipo quede reservado a los trabajadores designados para ello.

Artículo 4. El empresario debe someter a una comprobación inicial, tras su instalación y antes de la puesta en marcha por primera vez, y a una nueva comprobación después de cada montaje en un nuevo lugar o emplazamiento a todos los equipos de trabajo cuya seguridad dependa de sus condiciones de instalación con el fin de asegurar el buen funcionamiento de los equipos.

Si el equipo de trabajo es sometido a influencias que puedan ocasionar deterioros, el empresario debe realizar comprobaciones y pruebas de carácter periódico, con objeto de asegurar el cumplimiento de las disposiciones de seguridad y de salud y de remediar a tiempo dichos deterioros. Deberá realizarse también en caso de accidentes, fenómenos naturales o falta prolongada de uso.

2. Objetivos.

Este proyecto persigue dos objetivos principales.

1. Experimentar en primera persona algunos de los riesgos que se pueden encontrar al manipular carretillas elevadoras contrapesadas en el entorno de unas naves industriales destinadas al almacenamiento convencional paletizado.
2. Someter a la carretilla elevadora a situaciones límite de funcionamiento y estudiar bajo qué condiciones se produce vuelco.

Para conseguir tal fin lo primero que se hace es el modelado de una carretilla elevadora mediante las técnicas que ofrecen el Diseño Asistido por ordenador (CAD). Con estas mismas herramientas se realizan diferentes objetos tales como palés, carga, estanterías y naves industriales para lograr un escenario lo más representativo posible de las condiciones que se pueden encontrar en el entorno de trabajo.

Evaluando los riesgos asociados a la manipulación de carretillas elevadoras y al trabajo en lugares con estanterías metálicas para el almacenamiento de palés, y con ayuda de herramientas de simulación y realidad virtual, se realiza el texturizado de cada componente y el ensamblaje de los diferentes objetos para lograr el entorno de la escena.

Finalmente, con un programa de animación en 3D y realidad virtual, se elabora la composición de la escena de realidad virtual inmersiva, profundizando en el aspecto formativo del uso de carretillas elevadoras y mostrando las posibles consecuencias que genera su incorrecta utilización.

3. Material y métodos.

A nivel de empleo de programas de diseño, el presente proyecto se puede dividir en tres fases bien diferenciadas. La primera consiste en el modelado 3D y exportación de los objetos que componen la escena. La segunda consiste en el texturizado de los modelos 3D y la creación de la escena de realidad virtual inmersiva, y la tercera consiste en la edición de sonido, muy importante a la hora de crear una escena de realidad virtual inmersiva. Para cada una de las fases se utilizó un software diferente: Solidworks 2014 y Autodesk 3D Studio Max para el modelado y exportación, Unity para el texturizado y creación de la escena de realidad virtual inmersiva y Audacity para la edición de sonido.

3.1. Solidworks 2014

SolidWorks es un programa de diseño asistido por ordenador para modelado 3D, desarrollado por SolidWorks Corporation, una subsidiaria de Dassault Systèmes (Suresnes, Francia). Es un modelador de sólidos 3D paramétrico, que usa el kernel de modelado geométrico Parasolid. Fue introducido en el mercado en 1995 para competir con otros programas CAD como Pro/ENGINEER, NX, Solid Edge, CATIA, y Autodesk Mechanical Desktop.

El software permite modelar piezas y ensamblajes y extraer de ellos tanto planos como otro tipo de información necesaria para la producción. El proceso consiste en traspasar la idea del diseñador al sistema CAD, "construyendo virtualmente" la pieza o ensamblaje.

El software se caracteriza por su entorno intuitivo y por disponer de herramientas de diseño fáciles de utilizar, todo integrado en un único programa de diseño con más de 45 aplicaciones complementarias para facilitar el desarrollo de proyectos.

3.1.1. Módulos de Solidworks.

Solidworks contiene tres módulos principales: pieza, ensamblaje y plano o dibujo. La creación de un documento en cada uno de ellos genera un fichero con diferente extensión.

3.1.1.1. Módulo de pieza.

El Módulo de Pieza constituye un entorno de trabajo donde se puede diseñar modelos mediante el empleo de herramientas de diseño ágiles e intuitivas. El conjunto de funciones e iconos permite crear modelos tridimensionales partiendo de geometrías de croquis en 2D y obtener sólidos, superficies, etc.

3.1.1.2. Módulo de ensamblaje

Para realizar un ensamblaje se deben diseñar cada una de las piezas que lo conforman por separado y guardarlos como ficheros de pieza distintos. El módulo de ensamblaje permite insertar cada una de las piezas y asignar relaciones geométricas de posición para definir tridimensionalmente el ensamblaje.

La creación de ensamblajes permite analizar las posibles interferencias o choques entre los componentes insertados, así como simular el conjunto mediante motores lineales, rotativos y evaluar la correcta cinemática del conjunto.

3.1.1.3. Módulo de plano o dibujo.

Es el tercer módulo principal que contiene Solidworks. Permite crear planos con las vistas de los modelos o ensamblajes de forma automática.

3.2. Autodesk 3D Studio Max.

Autodesk 3D Studio Max es un software de renderización, simulación y modelado en 3D desarrollado por Autodesk. Dispone de una sólida capacidad de edición, una omnipresente arquitectura de plugins y una larga tradición en plataformas de Microsoft Windows.

3D Studio Max cuenta con un editor de materiales, en el cual se diseñan los distintos materiales, para una escena, los materiales de 3D Studio Max pueden llegar a ser

infinitos, ya que permite ir creando o subdividiendo en cada casilla y repite el proceso. Su editor de materiales muestra solamente 24 materiales distintos dentro de una escena, sin embargo se pueden crear un número indeterminado de materiales. Una de las grandes ventajas de este programa es que permite visualizar los materiales aplicados a una escena sin necesidad de renderizarla.

Una gran ventaja que ofrece es que permite la exportación de modelos y ensamblajes 3D en un amplio abanico de formatos y extensiones, entre ellos el formato .fbx, requerido para el programa de simulación y creación de escenas de realidad virtual Unity.

3.2.1. Exportación de modelos.

Para exportar los modelos elaborados en Solidworks a Unity es necesario transformar los archivos al formato FBX de Autodesk desde 3D Studio Max. Para ello es necesario realizar los siguientes pasos:

- Descargar el exportador de fbx desde la Página web de Autodesk.
- Exportar los modelos seleccionados desde 3D Studio Max (con File->Export) al formato .fbx.
- Copiar el archivo fbx resultante a la carpeta de proyecto de Unity.
- Cuando se inicie Unity, el archivo .fbx ha sido importado automáticamente.

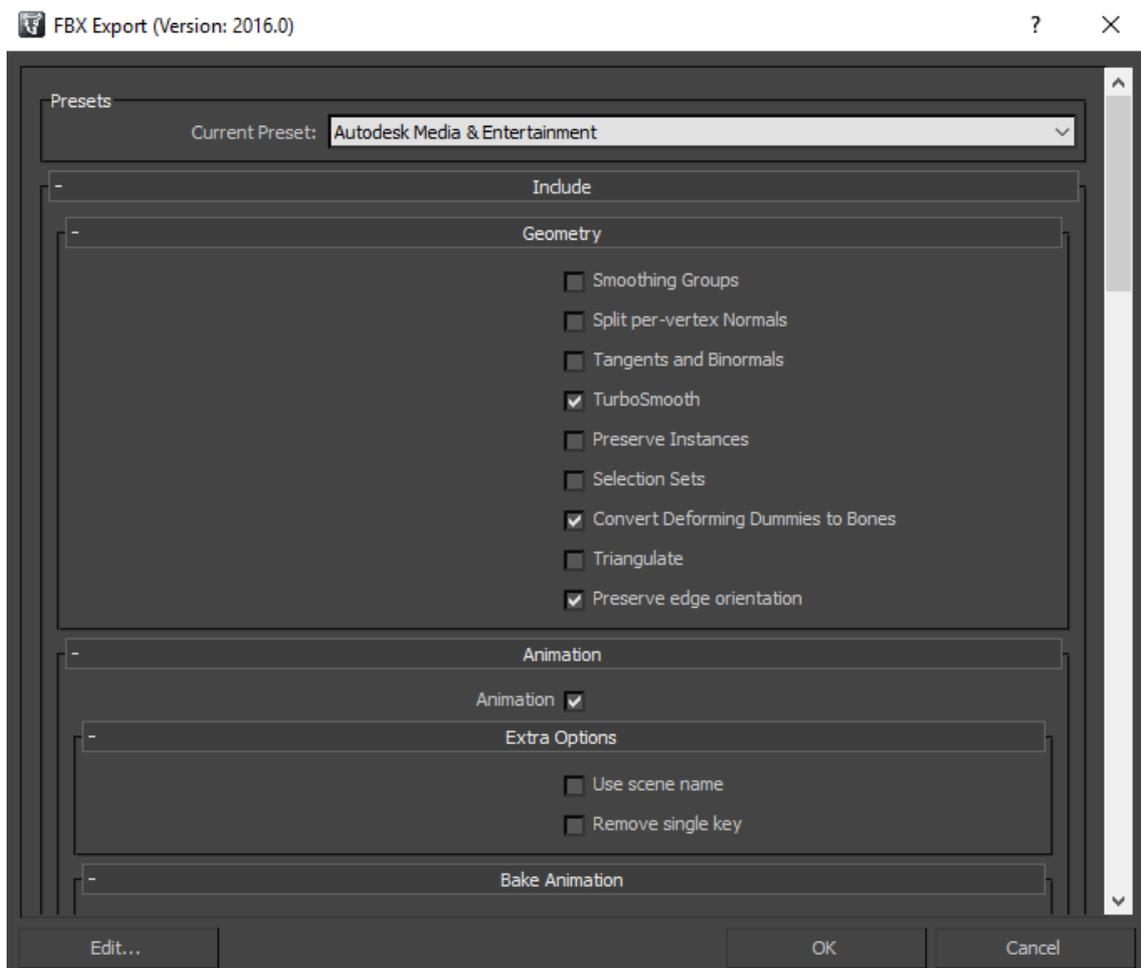


Fig. 6. Exportador de fbx

3.3. Unity.

Unity es un motor gráfico multiplataforma creado por Unity Technologies para Microsoft Windows, OS X y Linux. Es una herramienta que ayuda a desarrollar videojuegos para diversas plataformas, escenas de realidad virtual y simulaciones de todo tipo, mediante un editor y scripting para crear entornos con un acabado profesional.

La empresa Unity Technologies fue fundada en 2004 por David Helgason, Nicholas Francis y Joachim Ante en Copenhague, después de su primer juego, GooBall, que no obtuvo éxito. Los tres reconocieron el valor del motor y las herramientas de desarrollo y se dispusieron a crear un motor que cualquiera pudiera usar a un precio accesible.

Unity 3D provee de un editor visual muy útil y completo, donde de manera ágil e intuitiva se pueden importar modelos 3D, texturas, sonidos, etc. para después establecer interacción entre ellos. Incluye la herramienta de desarrollo MonoDevelop, con la que se pueden crear scripts en JavaScript y C#, para interactuar con los modelos 3D, texturas y sonidos importados.

Unity puede usarse junto con 3D Studio Max, Maya, Softimage, Blender, Modo, ZBrush, Cinema 4D, Cheetah3D, Adobe Photoshop, Adobe Fireworks y Allegorithmic Substance. Los cambios realizados a los objetos creados con estos productos se actualizan automáticamente en todas las instancias de ese objeto durante todo el proyecto sin necesidad de volver a importar manualmente.

El motor gráfico utiliza Direct3D en Windows, OpenGL en Mac y Linux y OpenGL ES en Android e iOS. Tiene soporte para mapeado de relieve, reflexión de mapeado, mapeado por paralaje, pantalla de espacio oclusión ambiental (SSAO), sombras dinámicas utilizando mapas de sombras, render a textura y efectos de post-procesamiento de pantalla completa.

Mecanim es la tecnología de animación de Unity, que ha estado en desarrollo durante años, primero por la empresa del mismo nombre, a continuación, por las oficinas de Unity en Canadá, después de su adquisición. La tecnología está diseñada para llevar el movimiento fluido y natural de los personajes con una interfaz eficiente. Mecanim incluye herramientas para la creación de máquinas de estados, árboles de mezcla, manipulación de los conocimientos nativos y retargeting automático de animaciones, desde el editor de Unity.

La interfaz principal de Unity se compone de ventanas con pestañas que pueden ser re-arregladas, agrupadas o desadjuntadas y minimizadas. Las ventanas más comunes y útiles se muestran en sus posiciones por defecto, en la figura 7.

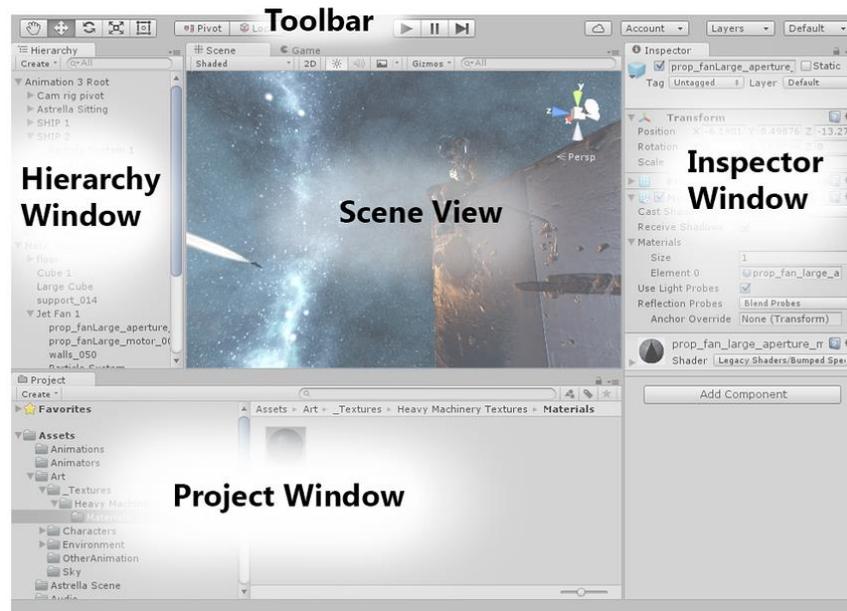


Fig. 7. Interfaz principal de Unity

- La ventana de proyecto que se muestra en la figura 8, muestra los assets de librería que están disponibles para ser usados. Cuando se importan los modelos procedentes de 3D Studio Max, estos aparecen aquí.

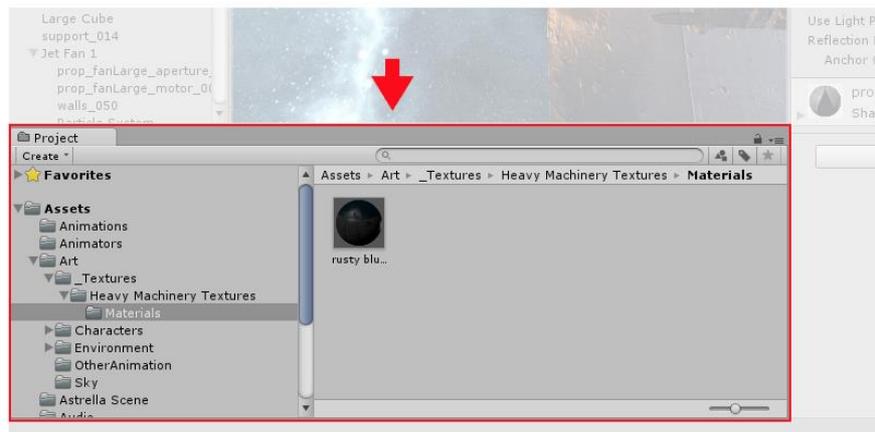


Fig. 8. Ventana de proyecto

- La vista de la escena (figura 9) permite una navegación visual y editar la escena. Puede mostrar una perspectiva 2D o 3D dependiendo del tipo de proyecto en el que se esté trabajando.

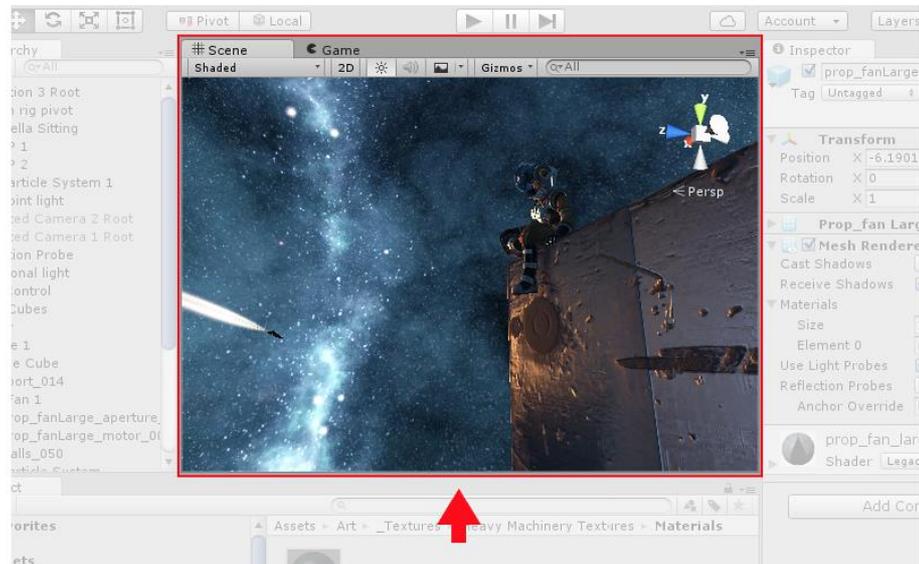


Fig. 9. Vista de la escena

- La ventana de jerarquía (figura 10) es una representación de texto jerárquico de cada objeto en la escena. Cada elemento en la escena tiene una entrada en la jerarquía, por lo que las dos ventanas están inherentemente vinculadas. La jerarquía revela la estructura de cómo los objetos están agrupados el uno al otro.

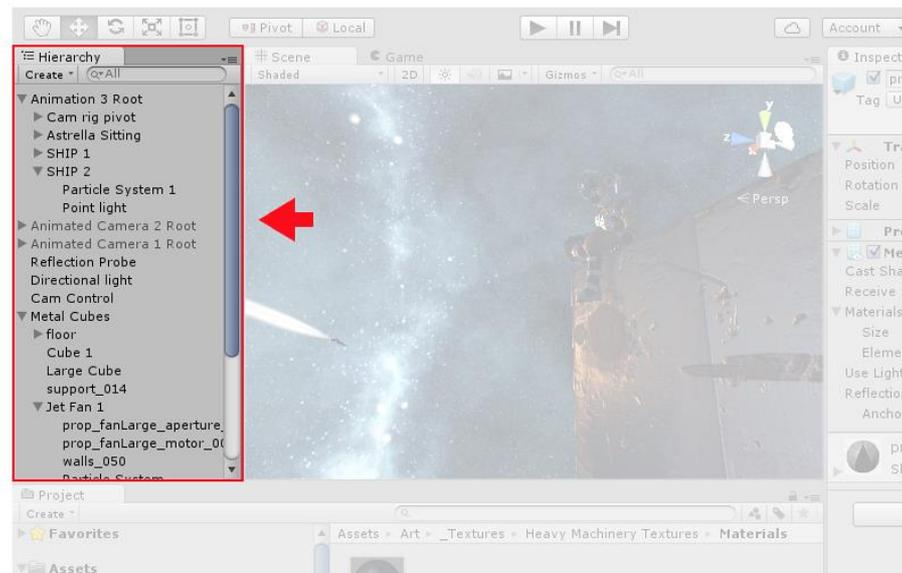


Fig. 10. Ventana de jerarquía

- La ventana del inspector (figura 11) permite visualizar y editar todas las propiedades del objeto actualmente seleccionado. Puesto que diferentes objetos tienen diferentes propiedades, el layout (diseño) y contenido de la ventana del inspector va a variar en función de la selección.

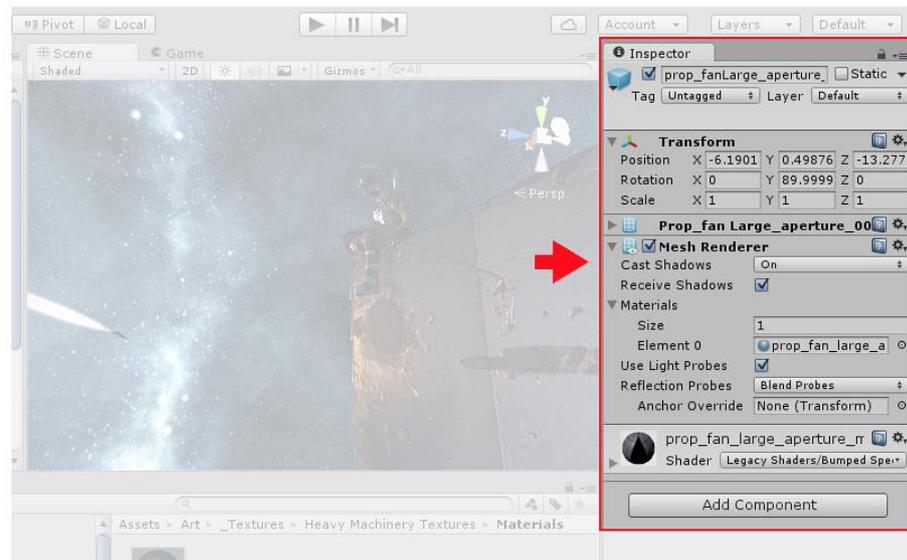


Fig. 11. Ventana de inspector

- La barra de herramientas proporciona un acceso a las características más esenciales para trabajar. En la izquierda se sitúan las herramientas básicas para manipular la “scene view” y los objetos dentro de ésta. En el centro están los controles de reproducción y pausa. Los botones a la derecha dan acceso a los servicios de Unity Cloud y cuenta de Unity, seguido por un menú de visibilidad de capas, y finalmente el menú del layout del editor, que proporciona algunos diseños alternativos para la ventana del editor, y permite guardar propios layouts personalizados. La barra de herramientas no es una ventana, y solamente es parte de la interfaz de Unity que no se puede reajustar.



Fig. 12. Barra de herramientas

En noviembre de 2010 se lanzó el Unity Asset Store que es un recurso disponible en el editor de Unity. Más de 150.000 usuarios de Unity pueden acceder a la colección que supera los 4.400 paquetes de Assets en una amplia gama de categorías, incluyendo modelos 3D, texturas y materiales, sistemas de partículas, música y efectos de sonido, tutoriales y proyectos, paquetes de scripts, extensiones para el editor y servicios en línea.

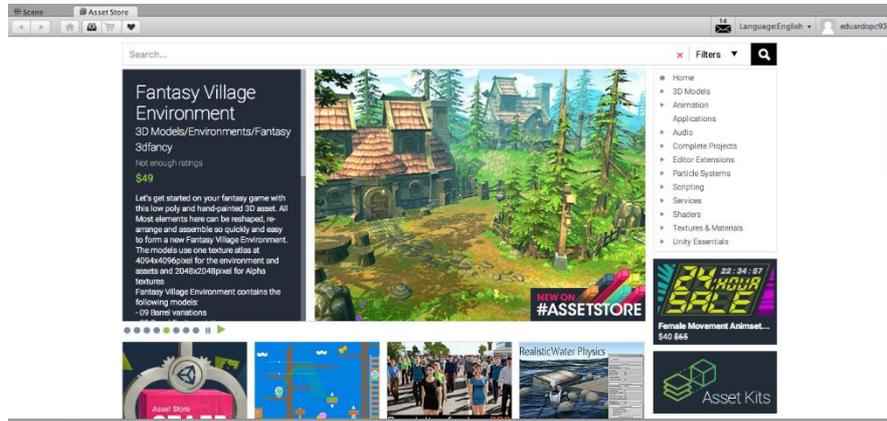


Fig. 13. Asset store

3.4. MonoDevelop.

MonoDevelop es el ambiente de desarrollo integrado (IDE) proporcionado con Unity. Un IDE combina la operación familiar de un editor de texto con características adicionales para depurar y gestionar otras tareas de proyecto.

A finales del 2003 un grupo de desarrolladores provenientes de la comunidad Mono comenzó a migrar SharpDevelop a Linux usando las librerías GTK.

MonoDevelop ha sido absorbido por el Proyecto Mono y está siendo activamente mantenido por Novell y la comunidad Mono. Mono es el nombre de un proyecto de código abierto iniciado por Ximian y actualmente impulsado por Novell para crear un grupo de herramientas libres, basadas en GNU/Linux y compatibles con .NET según lo especificado por el ECMA. En la figura 14 se muestra la interfaz principal de MonoDevelop.

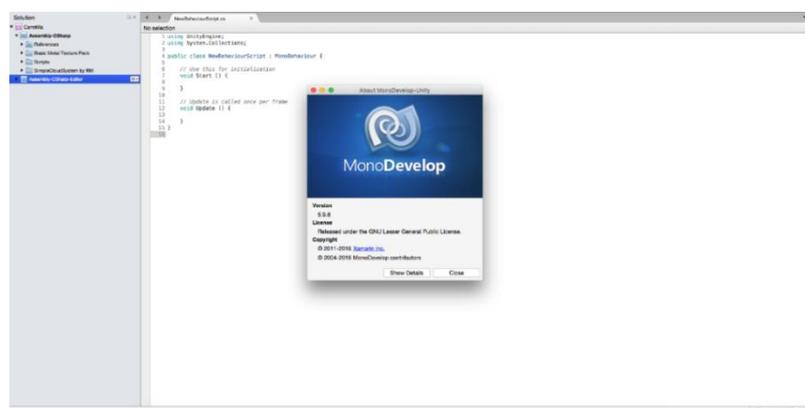


Fig. 14. Interfaz principal de MonoDevelop

3.4.1. Scripts.

El comportamiento de los GameObjects es controlado por los componentes que están adjuntos. Aunque los componentes integrados de Unity pueden ser muy versátiles, se necesita ir más allá de lo que éstos pueden proporcionar para implementar las características propias de la escena. Unity permite crear interacciones a GameObjects utilizando scripts.

A diferencia de la mayoría de los assets, los scripts son usualmente creados dentro de Unity directamente. Se puede crear un nuevo script desde el menú Create en la parte superior izquierda del panel del Proyecto o seleccionando Assets > Create > C# Script o JavaScript, desde el menú principal. El nuevo script será creado en la carpeta que se haya seleccionado en el Panel del Proyecto.

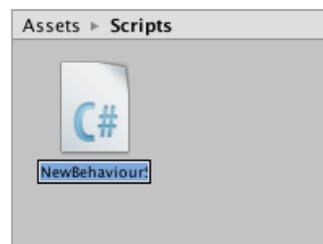


Fig. 15. Creación de script

3.4.1.1. Características de un script.

Cuando se hace doble click en un script en Unity, éste se abrirá en el editor de texto. Por defecto, Unity va a utilizar MonoDevelop.

Un script hace sus conexiones con el funcionamiento interno de Unity al implementar una clase que deriva desde la clase integrada llamada MonoBehaviour. Se puede pensar en una clase como un tipo de Component o interacción que puede ser adjuntado a GameObjects. Cada vez que se adjunte un componente script a un GameObject, éste crea una nueva instancia del objeto definido por el plano. El nombre de la clase es tomado desde el nombre que se proporciona cuando el archivo se crea. El nombre de la clase y el archivo se debió suministrar cuando el archivo fue creado. El nombre de la clase y el archivo debe ser el mismo para permitirle al componente script ser adjuntado al GameObject.

Lo primero a tener en cuenta son dos funciones definidas dentro de la clase. La función Update es el lugar para colocar el código que se encargará de la actualización por frame para el GameObject. Este puede incluir movimiento, acciones y responder al input del usuario, básicamente cualquier objeto que necesite ser manejado en el tiempo durante la puesta en marcha de la escena. Para que la función Update haga su trabajo, a veces es útil configurar variables, leer preferencias y hacer conexiones con otros GameObjects antes de que cualquier acción de la escena tome lugar. La función Start va a ser llamada por Unity antes de que la puesta en marcha de la escena comience y es un lugar ideal para hacer cualquier inicialización.

```
using UnityEngine;
using System.Collections;

public class MainPlayer : MonoBehaviour {

    // Use this for initialization
    void Start () {

    }

    // Update is called once per frame
    void Update () {

    }

}
```

Fig. 16. Funciones principales de un script

Nada del código será activado hasta que una instancia del script sea adjuntada al GameObject. Se puede adjuntar un script arrastrando el script asset a un GameObject en el panel de la jerarquía o al inspector del GameObject seleccionado en ese momento. También hay un sub-menú de Scripts en el menú del Componente que va a contener todos los scripts disponibles en el proyecto, incluyendo aquellos que se han creado. La instancia de script se ve mucho mejor como cualquier otro Componente en el Inspector.

3.5. Audacity.

Audacity es un software informático de multiplataforma libre, que se puede usar para grabación y edición de audio, distribuido bajo la licencia GPL.

Fue creado en otoño de 1999 por Dominic Mazzoni y Roger Dannenberg en la Universidad de Carnegie Mellon. Tras lo cual fue publicado en SourceForge.net

como software libre en mayo de 2000.

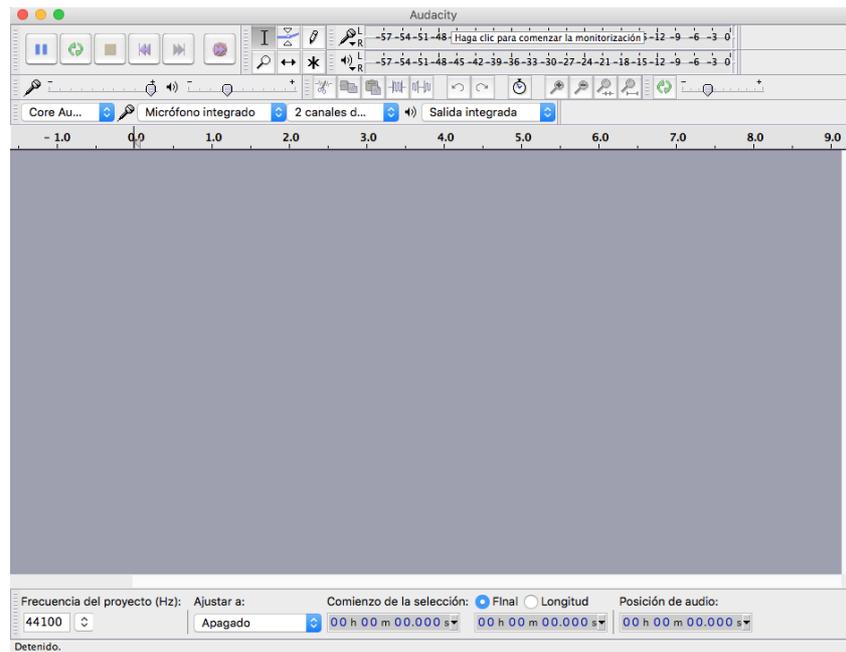


Fig. 17. Interfaz principal de Audacity

3.5.1. Características de Audacity.

Las principales características que posee Audacity se enuncian a continuación.

- Grabación de audio en tiempo real.
- Edición archivos de audio tipo Ogg Vorbis, MP3, WAV, AIFF, AU , LOF y WMP.
- Conversión entre formatos de tipo audio.
- Importación de archivos de formato MIDI, RAW y MP3.
- Edición de pistas múltiples.
- Inclusión de efectos al sonido, tales como eco, inversión, tono, etc.
- Posibilidad de usar plug-ins para aumentar su funcionalidad.

En lo que respecta a modelos 3D, la escena de realidad virtual se compone principalmente de una carretilla elevadora y una serie de naves industriales a modo de almacenes, con sus respectivas estanterías metálicas, palés y cargas de diferente masa.

3.6. Carretilla elevadora.

La carretilla que se va a estudiar es el modelo Caterpillar DP70, una carretilla elevadora contrapesada con capacidad de carga de 7000 kg y elevación de 4000 mm. Esta carretilla es muy apropiada para trabajos de carga, manipulación y almacenaje en grandes superficies y exteriores. El motor diésel S6S-T de seis cilindros, dos velocidades y una velocidad máxima de 27 Km/h sin carga, está alimentado con combustible diésel y funciona perfectamente en entornos duros, húmedos y sucios.

La DP70 combina fuerza, estabilidad y durabilidad con un rendimiento limpio y eficiente y un diseño pensado para el operario. Con todo esto, se intenta aumentar la productividad y reducir los costes de su operación.



Fig. 18. Caterpillar DP70

El modelo 3D de la carretilla elevadora empleado para crear la escena de realidad virtual es el utilizado en el Trabajo Fin de Máster “Simulación de accidente de vuelco de carretillas elevadoras” publicado en 2015, añadiéndole elementos estructurales necesarios para poder establecer el funcionamiento real de la misma, tales como las guías de la horquilla y el mástil, realizados en Solidworks 2014.

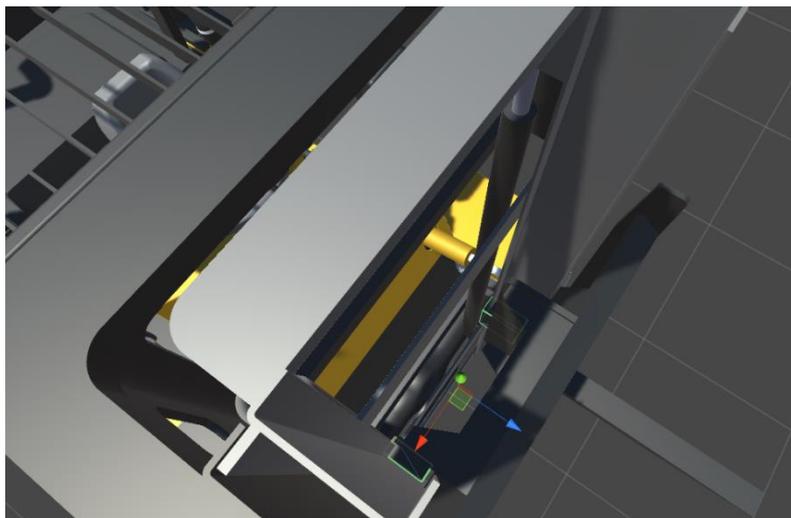


Fig. 19. Guías para la horquilla

La texturización de la carretilla elevadora se ha llevado a cabo en Unity, utilizando acero para chasis y mástil, y caucho para los neumáticos.



Fig. 20. Modelo 3D carretilla elevadora texturizada

Las especificaciones técnicas de la carretilla elevadora que proporciona la empresa Caterpillar se adjuntan en el Anexo I del presente proyecto.

3.7. Lugar de trabajo.

El almacén y los distintos elementos estructurales que contiene, facilita su rápido desmontaje y traslado, su reconfiguración o ampliación según se descubran nuevas necesidades. Permite la fácil y rápida regulación de los niveles de carga para ajustarlos a los volúmenes de la misma, lo que posibilita al máximo el espacio.

Consiste en un conjunto de estanterías y pasillos de almacenaje dispuestos estratégicamente en un almacén o nave industrial, de modo que permitan almacenar de forma segura y organizada el máximo número de unidades de carga aprovechando el espacio disponible. La distribución del almacén se realiza normalmente mediante estanterías laterales de simple acceso y centrales de doble acceso.

Permite, además, el acceso directo a cada palé o contenedor con la única ayuda de una carretilla. La altura de las estanterías metálicas, así como el ancho mínimo de los pasillos de acceso, dependerán únicamente del tipo de carretilla y de las dimensiones y del lado por el que se va a trabajar con el palé.

3.7.1. Nave Industrial.

Se ha utilizado el programa Solidworks 2014 y Unity para diseñar y texturizar (con hormigón armado y pavimento industrial), respectivamente, una nave industrial de 2000 m², aproximadamente, de forma rectangular con techo de cercha a dos aguas.

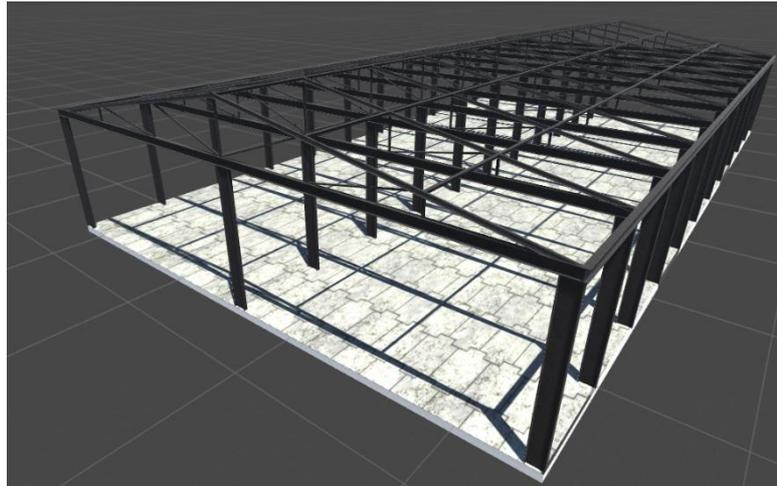


Fig. 21. Modelo 3D nave industrial texturizada

La distribución se ha hecho mediante dos estanterías centrales simples, una a cada lado de los pilares centrales, y dos estanterías dobles, dejando un pasillo de seis metros entre estanterías. La anchura de los pasillos no debe ser inferior en sentido único a la anchura del vehículo o a la de la carga incrementada en 1 metro, por lo que cumple con la norma.

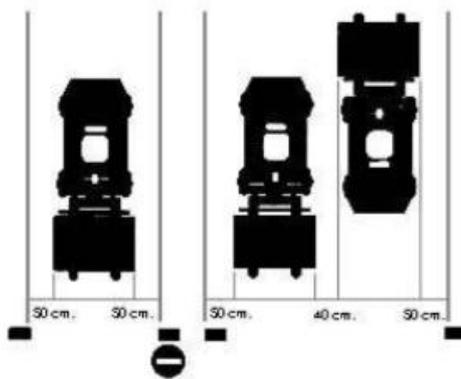


Fig. 22. Anchura de pasillos de la nave industrial en función a la normativa vigente

Además, se ha dotado a la nave de dos zonas de diez metros, a la entrada y al final del almacén, donde se dispone de espacio necesario para poder efectuar la carga y descarga de mercancías y preparación de pedidos.

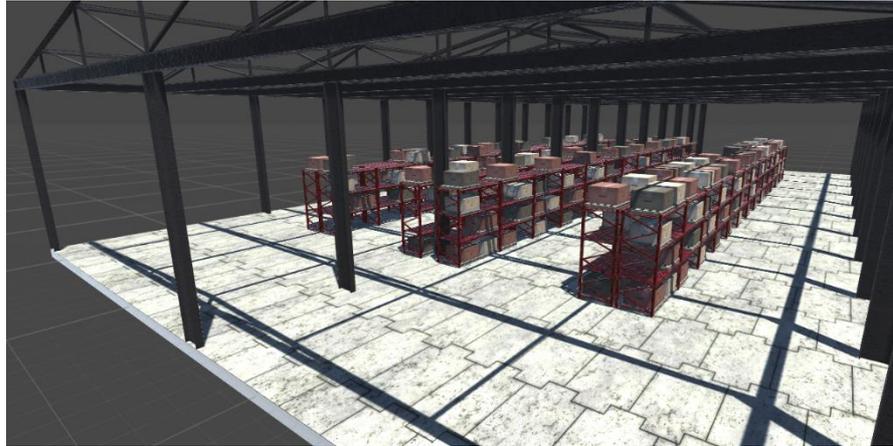


Fig. 23. Zona de carga y descarga

3.7.2. Estanterías y carga.

Teniendo en cuenta que la carretilla elegida para la elaboración de la escena de realidad virtual inmersiva es una carretilla contrapesada Caterpillar DP7, se ha decidido diseñar una estantería de 4000 mm de altura utilizando el programa Solidworks 2014 y su posterior texturización utilizando Unity.



Fig. 24. Estanterías metálicas texturizadas

La carretilla elevadora escogida soporta una carga máxima de 7000 Kg, por lo tanto, para estudiar las condiciones de vuelco en función de la carga manipulada se han establecido tres tipos de cargas texturizadas en Unity con distinta masa, distinguidas por colores (Fig. 25, 26 y 27).



Fig. 25. Carga de 2500Kg



Fig. 26. Carga de 5000kg



Fig. 27. Carga de 8000Kg

Una vez elaboradas y texturizadas estanterías y cargas, se realiza la composición de la estantería con las cargas dotando a la nave industrial.



Fig. 28. Composición estanterías y carga en el interior de la nave industrial

3.8. Sala de realidad virtual.

La implantación y puesta en práctica de la escena de realidad virtual inmersiva creada se realiza en el CEDIT (Centro de Desarrollo e Innovación Tecnológica). Es un edificio de la Universidad Politécnica de Cartagena, situado en el Parque Tecnológico de Fuente Álamo. Su uso contempla espacios para que los grupos de I+D colaboren con las empresas y para proyectos que requieran equipos y material destinado a plantas piloto o prototipos, además de servir de sede provisional a empresas de base tecnológica creadas por investigadores.

Cuenta con cinco laboratorios pesados, veinte laboratorios de I+D+i, salas para servidores informáticos, despachos, salas de reuniones y un salón de actos, estando uno de los laboratorios dedicado a inmersión en realidad virtual.

Esta sala está dotada de un proyector, equipo de sonido, unas gafas Oculus Rift de realidad virtual y una plataforma móvil, que contiene un volante, asiento y pedales.



Fig. 29. Sala de realidad virtual

3.8.1. Plataforma.

Consiste en una plataforma que contiene asiento, volante y pedales, cuya funcionalidad reside en experimentar las sensaciones de conducción de la carretilla elevadora. Es movida por tres motores, dotando el sistema de tres grados de libertad (traslación en el eje Z y rotación alrededor de los ejes X e Y).



Fig. 30. Plataforma móvil



Fig. 31. Motores de la plataforma móvil

3.8.2. Oculus Rift.

Oculus Rift es un casco de realidad virtual que ha sido desarrollado por Oculus VR. Durante su periodo como compañía independiente, Oculus VR ha invertido 91 millones de dólares para el desarrollo de Oculus Rift.

El primer prototipo del dispositivo utilizaba una pantalla de 5,6 pulgadas, pero tras un tiempo, Oculus decidió que, como el panel de 5,6 pulgadas no estaba disponible en suficientes cantidades, cambiaría su diseño para utilizar una nueva pantalla de 7 pulgadas.

El tiempo de refresco de píxeles de este nuevo panel es mucho más rápido, lo que reduce la latencia y el desenfoque de movimiento al girar la cabeza de una forma rápida. La pantalla LCD es más brillante y la profundidad de color es 24 bits por píxel. La pantalla de 7 pulgadas también hace que el 3D estereoscópico ya no tenga superposición de 100%, el ojo izquierdo ve área adicional a la izquierda y el ojo derecho ve área adicional a la derecha. Esto imita la visión humana normal.

El campo de visión es de más de 90 grados horizontales, 110 grados en diagonal, que es más del doble del campo de visión de la mayoría de los dispositivos de la competencia, y es la fuerza principal del dispositivo. Está destinado a cubrir casi todo el campo visual del usuario de vista, para crear un fuerte sentido de la inmersión. La resolución es de 1280 x 800 (Con relación de aspecto 16:10), que conduce a un efectivo de 640 x 800 por ojo (Con relación de aspecto 4:5). Sin embargo, el DK1 no cuenta con un 100% de coincidencia entre los ojos, la resolución horizontal combinado es efectivamente superior a 640. La imagen para cada ojo se muestra en el panel como una imagen tipo cojín que se corrige

a continuación, por las lentes en el visor para la generación de una imagen esférica-asignada para cada ojo.

El peso del visor es de aproximadamente 379 g, un aumento de alrededor de 90 gramos respecto al primer prototipo, debido al aumento de tamaño de la pantalla. El visor dispone de un dial de cada lado, el cual se puede activar con un destornillador, que permite ajustar que la pantalla se mueva más cerca o más lejos de los ojos.



Fig. 32. Oculus Rift

4. Resultados.

4.1 Escena de realidad virtual.

Se procede, en primer lugar, a describir paso a paso el desarrollo de la escena de realidad virtual inmersiva para cumplir con el primer objetivo de este trabajo.

Una vez se han importado los modelos de la carretilla elevadora y de la nave industrial a Unity, se ha establecido la escala 1:1 a cada elemento y se han texturizado, se comienza a crear la escena de realidad virtual inmersiva. En primer lugar se ha creado un plano que servirá de suelo.



Fig. 33. Plano del suelo

Cada elemento que se ha creado o importado en Unity constituye un GameObject. No obstante, los GameObjects no hacen nada por sí mismos. Éstos necesitan propiedades especiales antes de que puedan volverse un personaje, un ambiente, o un efecto especial.

Para elaborar la escena de realidad virtual inmersiva se deben establecer interacciones entre los GameObjects importados y creados, y las entradas que realice el usuario, es decir, crear scripts estableciendo el tipo de interacción o el tipo de comportamiento que se desea atribuir a cada modelo mediante programación en lenguaje C# o Java.

4.1.1. Carretilla elevadora.

En lo que respecta a la carretilla elevadora, en primer lugar se han debido crear los llamados BoxColliders, es decir, qué partes de la carretilla se debían de comportar como sólidos ante cualquier tipo de colisión.

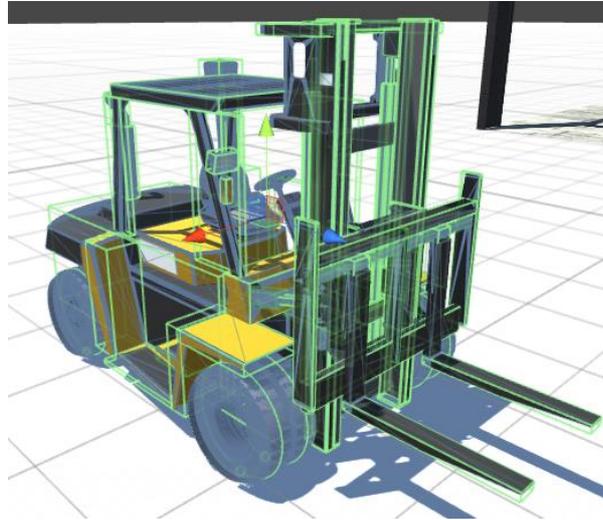


Fig. 34. Colliders de la carretilla elevadora

Mención especial tienen los neumáticos, puesto que existe un tipo de Collider específico para poder simular el comportamiento real de los neumáticos, llamado WheelCollider.



Fig. 35. WheelColliders en neumáticos

Una vez que se han aplicado los Colliders, se le asigna a la carretilla un Rigidbody. Los Rigidbodies permiten a los GameObjects actuar bajo el control de la física. El Rigidbody puede recibir fuerza y torque para hacer que sus objetos se muevan de un modo realista. Cualquier GameObject debe contener un Rigidbody para ser influenciado por gravedad, actue bajo fuerzas agregadas vía scripting, o interactuar con otros objetos a través del motor de física.

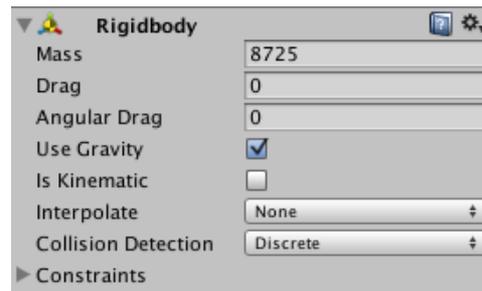


Fig. 36. Rigidbody carretilla elevadora

La carretilla elevadora ha sido dividida en tres GameObjects: Chasis y ruedas, horquilla y mástil. El motivo reside en que cada GameObject sólo puede contener un Rigidbody, y se necesita aplicar interacciones distintas a cada elemento.

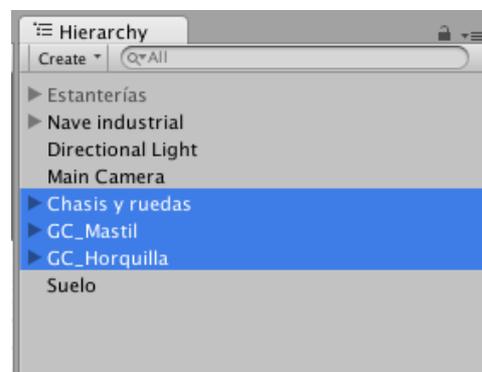


Fig. 37. GameObjects que componen la carretilla elevadora

A la carretilla elevadora se le han aplicado siete scripts a los que se les ha asignado el siguiente nombre: Motor, Movimiento de horquilla, Movimiento de mástil, Inclinación de mástil y horquilla, Centro de gravedad, Oculus Rift y Plataforma.

4.1.1.1. Motor.

Durante el desarrollo del Grado en Ingeniería en Tecnologías Industriales se ha aprendido a programar en C, muy similar al C# que admite Unity. Sin embargo, este script se ha realizado en lenguaje Java, debido a que para establecer que las ruedas traseras rotasen sobre su propio eje además de girar a izquierda o derecha se precisaba elaborar dos scripts distintos, por lo que es recomendable tener un solo script para cada comportamiento o interacción con el usuario.

El primer paso para realizar un script ha consistido en declarar las variables necesarias. Existen diferentes tipos de variables a declarar:

- WheelCollider. Se utiliza para declarar ruedas de vehículos.
- Float. Se utiliza para asignar valores con decimales.
- Int. Se utiliza para asignar valores de tipo entero.
- Transform. Se utiliza para declarar GameObjects.

- Strings. Se utiliza para declarar cadenas.
- GUIText. Se utiliza para insertar texto en la escena.
- AudioSource. Se utiliza para insertar archivos de audio.

```

1 #pragma strict
2
3 var ruedaFDE : WheelCollider;
4 var ruedaFDI : WheelCollider;
5 var ruedaFIE : WheelCollider;
6 var ruedaFII : WheelCollider;
7 var ruedaTD : WheelCollider;
8 var ruedaTI : WheelCollider;
9 var ruedaFDETrans : Transform;
10 var ruedaFDITrans : Transform;
11 var ruedaFIETrans : Transform;
12 var ruedaFIITrans : Transform;
13 var ruedaTDTrans : Transform;
14 var ruedaTITrans : Transform;
15
16 var AnguloGiroRuedas:int=-45;
17
18 var EngineTorque:float=30;
19 var MAXEngineRPM:float=2500;
20 var MINEngineRPM:float=1000;
21 var EngineRPM:float=0;
22 var FrenoMotor:float=50;
23 var BrakeForce:float=0;
24 var FrenoPedal:float=100;
25
26 var speed:float;
27
28 var CurrentGear:int=0;
29 var GearRatio:float[];
30 var sGUI : String = "";
31
32 var SpeedText:GUIText;
33 var RPMText:GUIText;
34 var GearText:GUIText;
35
36 var AudioSources:AudioSource[];
37
  
```

Fig. 38. Declaración de variables en script
 Motor

Una vez declaradas las variables necesarias, se ha elaborado la aceleración de la carretilla. Para conseguir la aceleración de una manera realista se hace uso del componente motorTorque, que consiste en aplicar rotación a las ruedas mediante el WheelCollider declarado anteriormente. Cuando el usuario pisa el acelerador, que se define mediante la sentencia Input.GetAxis("Vertical"), la variable Input pasa de un valor nulo a un valor de uno, con lo que se activa el torque que se aplica a las ruedas, provocando movimiento en la carretilla. Los Inputs se corresponden con las entradas que proporciona el usuario mediante el teclado o, como ocurre en este proyecto, mediante el volante y pedales situados en la plataforma móvil.

Para conseguir el giro de las ruedas traseras, en este caso, se ha hecho uso del componente steerAngle, que consiste en aplicar rotación sobre el eje Y (de color verde en Fig. 39) de los WheelColliders declarados anteriormente, permitiendo el giro a izquierda o derecha de la carretilla cuando el usuario gire el volante, mediante la entrada Input.GetAxis("Horizontal"). En este caso la variable Input pasa del valor nulo a un valor de -1 si se gira a la izquierda o a un valor de 1 si se gira a la derecha, activando el componente SteerAngle que produce el giro.

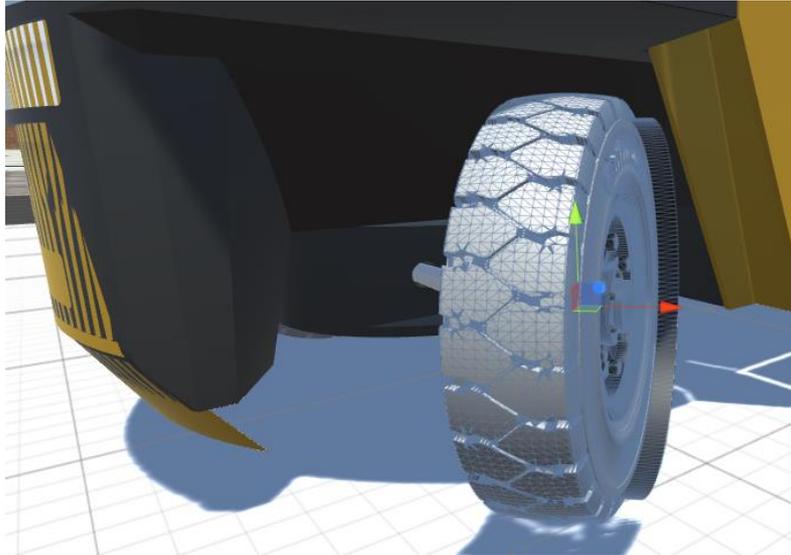


Fig. 39. Giro ruedas traseras

Seguidamente se ha elaborado el sistema de frenado de la carretilla elevadora. Para ello se han distinguido dos tipos de freno: freno motor y freno al pisar el pedal de frenado. Del mismo modo que existe una sentencia para provocar la aceleración de los WheelColliders, existe una sentencia para provocar el freno de los mismos: brakeTorque. De este modo, si el usuario deja de acelerar y no se pisa el pedal de freno se activa el freno motor, mientras que si se pisa el pedal de freno se activa el freno pedal.

```

120 //Aceleración y giro
121 ruedaFDE.motorTorque = EngineTorque * GearRatio[CurrentGear] * Input.GetAxis("Vertical");
122 ruedaFDI.motorTorque = EngineTorque * GearRatio[CurrentGear] * Input.GetAxis("Vertical");
123 ruedaFIE.motorTorque = EngineTorque * GearRatio[CurrentGear] * Input.GetAxis("Vertical");
124 ruedaFII.motorTorque = EngineTorque * GearRatio[CurrentGear] * Input.GetAxis("Vertical");
125 ruedaTD.steerAngle = AnguloGiroRuedas * Input.GetAxis("Horizontal");
126 ruedaTI.steerAngle = AnguloGiroRuedas * Input.GetAxis("Horizontal");
127
128 //Revoluciones Motor
129 EngineRPM = ((ruedaFDE.rpm + ruedaFDI.rpm + ruedaFII.rpm + ruedaFIE.rpm)/4) * GearRatio[CurrentGear];
130
131 //Velocidad
132 speed=GetComponent.<Rigidbody>().velocity.magnitude * 3.6f;
133
134 //Freno Motor
135 if(Input.GetAxis("Vertical")==0)
136     BrakeForce = FrenoMotor;
137
138 //Freno Pedal
139 if(Input.GetAxis("Vertical")<0 && EngineRPM>0)
140     BrakeForce = FrenoPedal;
141
142 //Durante aceleración freno=0
143 if(Input.GetAxis("Vertical")>0)
144     BrakeForce = 0;
145
146 //Durante marcha atrás freno=0
147 if(Input.GetAxis("Vertical")<0 && EngineRPM<=0)
148     BrakeForce = 0;
149
150 ruedaFDE.brakeTorque = BrakeForce;
151 ruedaFDI.brakeTorque = BrakeForce;
152 ruedaFIE.brakeTorque = BrakeForce;
153 ruedaFII.brakeTorque = BrakeForce;
  
```

Fig. 40. Aceleración y freno en script Motor

El siguiente paso ha consistido en incorporar la caja de cambios a la carretilla. Para ello se crea una función, que se define fuera de los límites de las funciones principales de lectura, y posteriormente se invoca la función dentro de la función FixedUpdate, para que Unity lo ejecute. Este modelo de carretilla contiene dos velocidades y una velocidad de marcha atrás. Para

ello se ha definido un bucle “for” para elevar marchas y otro bucle “for” para disminuir marchas. Se declara una variable llamada MAXEngineRPM inicializada con un valor de 2500, y otra variable llamada MINEngineRPM inicializada con un valor de 1000. Cuando las revoluciones del motor superan las 2500 rpm, mediante el bucle for, se eleva una marcha, mientras que si las revoluciones del motor son inferiores a 1000 rpm, mediante el bucle for de disminución de marcha, se disminuye una marcha. Si la velocidad es cero, es decir, la carretilla está parada, se pone la marcha neutra.

```

41 //función para cambiar de marchas
42 function CambioMarchas(){
43
44     if(speed== 0)
45     CurrentGear=0;
46
47     if(CurrentGear==0 && Input.GetAxis("Vertical") <0 && EngineRPM<0)
48     CurrentGear = GearRatio.Length-1;
49     else
50     if(CurrentGear==GearRatio.Length-1 && Input.GetAxis("Vertical") >0 && speed>0)
51     CurrentGear=0;
52     else{
53
54         if (EngineRPM >= MAXEngineRPM){
55             var AppropriateGear:int=CurrentGear;
56             for(var i:int=0; i < GearRatio.Length-1; i++){
57                 if(ruedaFIE.rpm * GearRatio[i] < MAXEngineRPM){
58                     AppropriateGear = i;
59                     break;
60                 }
61             }
62             CurrentGear = AppropriateGear;
63         }
64
65         if (EngineRPM <= MINEngineRPM){
66             for(var j:int= GearRatio.Length-2; j>=0; j--){
67                 if(ruedaFIE.rpm * GearRatio[j] > MINEngineRPM){
68                     AppropriateGear = j;
69                     break;
70                 }
71             }
72             CurrentGear = AppropriateGear;
73         }
74     }
75 }}
76
77 function FixedUpdate () {
78
79     //función de cambio de marchas
80     CambioMarchas();

```

Fig. 41. Función para el cambio de marchas en script Motor

Una vez terminada la configuración del motor de la carretilla elevadora, se ha incorporado el sonido del motor y el sonido característico de seguridad de los vehículos de carga y descarga cuando se circula marcha atrás. Para ello, al incorporar dos archivos de audio en el mismo script, es decir, dos AudioSource distintos, para no obtener errores a la hora de simular la escena, se deben declarar los archivos de audio en un array.

Para poder aplicar al sonido del motor el sonido característico que se produce al revolucionar el motor, existe una sentencia que se denomina pitch, que consiste en aplicar un cambio de tonalidad al sonido en cuestión. Para lograr un pitch que se ajuste lo máximo posible a la realidad, se ha establecido una relación entre las revoluciones que lleva el motor en cada instante y las revoluciones máximas del motor, lo que produce un tono distinto en cada distante.

```

185 //Sonido motor
186 AudioSource[0].pitch = Mathf.Abs(EngineRPM/MAXEngineRPM) + 0.5f;
187 if (AudioSources[0].pitch > 2.0f)
188     AudioSource[0].pitch = 2.0f;
189
190 //Sonido Marcha atras
191 if (EngineRPM>=0)
192     AudioSource[1].mute=true;
193
194 if (EngineRPM<0)
195     AudioSource[1].mute=false;

```

Fig. 42. Audio en script Motor

Por último, para que el usuario tenga referencia de la velocidad, revoluciones del motor y marcha en la que circula, se han incorporado unos GUITexts, que consiste en aplicar a la escena texto a modo de display. Sin embargo, no es posible incluir estos GUIs en la escena al implementar la cámara de las Oculus Rift.

```

165 //GUIs
166 if(SpeedText != null)
167   SpeedText.text=speed.ToString("f0")+ "\tKm/h";
168 if(RPMText != null)
169   RPMText.text=EngineRPM.ToString("f0")+ "\trpm";
170 if(GearText != null)
171   if(CurrentGear > 0 && CurrentGear < GearRatio.Length-1)
172     sGUI=(CurrentGear+1).ToString("f0");
173   else
174     if(CurrentGear == GearRatio.Length-1)
175       sGUI="R";
176   else
177     if(CurrentGear == 0 && Mathf.Abs(speed)>1)
178       sGUI=(CurrentGear+1).ToString("f0");
179   else
180     if(CurrentGear == 0 && Mathf.Abs(speed)<=1)
181       sGUI="N";
182
183   GearText.text=sGUI;
  
```

Fig. 43. GUIs en script Motor



Fig. 44. GUIs implementados en la escena

Una vez finalizado el script en Monodevelop se ha regresado a la pantalla principal de Unity para asignar el script al inspector del GameObject deseado, en este caso el GameObject llamado chasis y ruedas de la carretilla, junto con los dos AudioSource definidos.

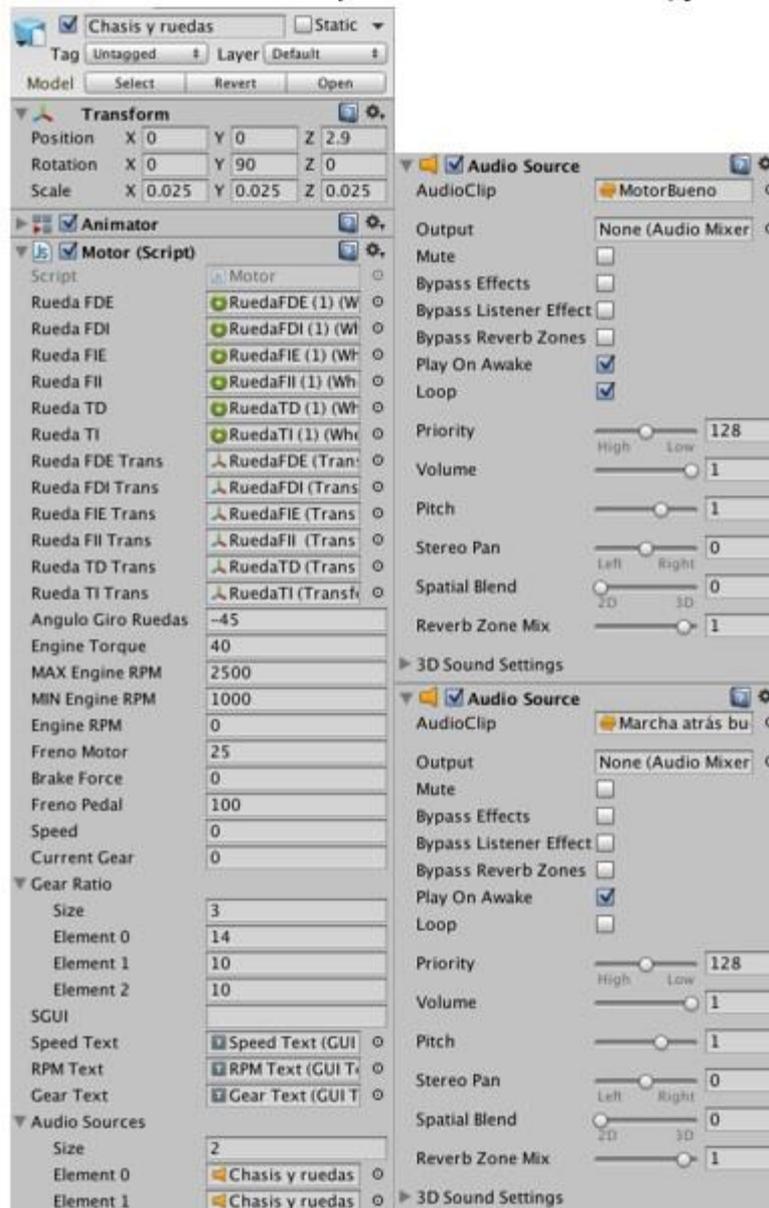


Fig. 45. Ventana de inspector del GameObject chasis y ruedas

4.1.1.2. Movimiento de Horquilla.

El siguiente GameObject a dotar de interacción entre el usuario y la escena es la horquilla. Para ello se ha creado un script, cuyo objetivo es conseguir que ante una entrada del usuario la horquilla se eleve y descienda.

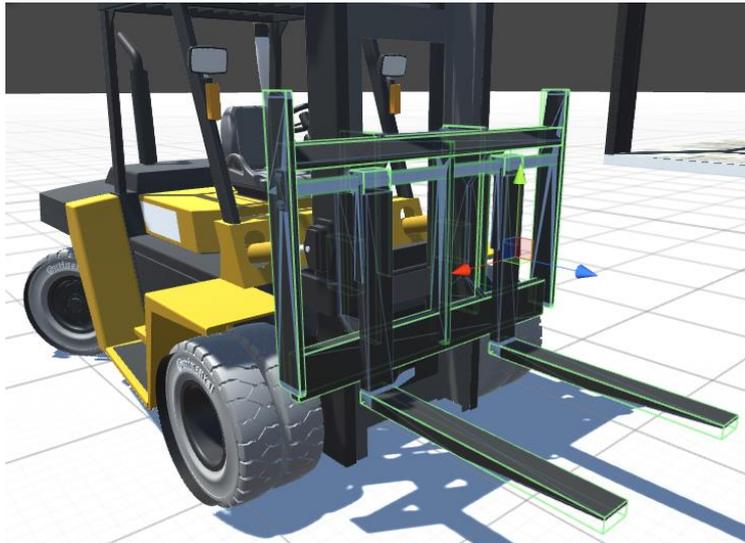


Fig. 46. Colliders de la horquilla

El resto de scripts se han elaborado en lenguaje C#, lenguaje aprendido a lo largo del Grado cursado. Para ello, lo único que se ha hecho es aplicar una fuerza ascendente o descendente a lo largo del eje Y (de color verde en la Fig. 46) en función del valor del Input que proporciona el usuario (1 para elevar y -1 para descender). Esta fuerza es aplicada al Rigidbody correspondiente mediante la sentencia AddForce.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class Elevador : MonoBehaviour {
5
6     private float motorForce=0;
7     public float fuerza=50;
8     public Rigidbody rb;
9     public AudioSource audio;
10
11
12     void Start () {
13
14         rb = GetComponent<Rigidbody>();
15         audio = GetComponent<AudioSource>();
16     }
17
18
19     void FixedUpdate () {
20
21         motorForce = Input.GetAxis ("Fire1")* fuerza;
22         rb.AddForce (0, motorForce, 0);
23
24     }
25 }
  
```

Fig. 47. Elevación y descenso en script de horquilla

Una vez determinado el comportamiento que se desea aplicar al GameObject en cuestión, se ha aplicado el efecto sonoro que debe acompañar al movimiento de la horquilla, mediante un nuevo AudioSource. En este caso, al ligar a este script un sólo archivo de audio no es necesario declararlo en un array.

```

void Update(){
    if(Input.GetAxis ("Fire1")!=0)
        audio.Play();
}
  
```

Fig. 48. Audio en script de horquilla

Finalizado el script, se ha asignado en el inspector del GameObject llamado Horquilla el mismo y el AudioSource correspondiente, para que se ejecute una vez comience la simulación.



Fig. 49. Ventana de inspector
GameObject horquilla

4.1.1.3. Movimiento del mástil.

En este modelo de carretilla elevadora, Caterpillar DP70, la horquilla se eleva hasta una altura de cuatro metros. Sin embargo, esta altura no se alcanza modificando únicamente la posición de la propia horquilla, puesto que ésta alcanza una altura máxima de dos metros. La carretilla contiene un mástil que también modifica su posición, pudiendo llegar la horquilla a los cuatro metros especificados.



Fig. 50. Colliders en mástil

El objetivo para este GameObject ha sido similar al de la horquilla: dotar de movimiento a lo largo del eje Y al mástil ante la entrada del usuario. Para ello, al igual que en el apartado anterior, se necesita aplicar al Rigidbody una fuerza para que el GameObject del mástil se eleve o descienda. Esta fuerza se aplica usando el comando AddForce.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class ElevadorSecund : MonoBehaviour {
5
6     private float motorForce2=0;
7     public Rigidbody rb2;
8     AudioSource audio2;
9
10
11     void Start () {
12
13         rb2 = GetComponent<Rigidbody>();
14         audio2 = GetComponent<AudioSource>();
15
16     }
17
18
19     void FixedUpdate () {
20
21         motorForce2 = Input.GetAxis ("Fire2")* 150;
22         rb2.AddForce (0, motorForce2, 0);
23
24     }
  
```

Fig. 51. Elevación y descenso en script de mástil

Una vez conseguido el comportamiento deseado se ha aplicado el sonido, similar al sonido de la horquilla, que debe acompañar al movimiento del mástil. Como ya se ha comentado en apartados anteriores, para añadir un archivo de sonido hay que declarar un AudioSource.

Del mismo modo que para los GameObjects anteriormente definidos, una vez finalizado el script con el comportamiento deseado, se asigna en el inspector del GameObject del mástil, junto con el AudioSource definido.

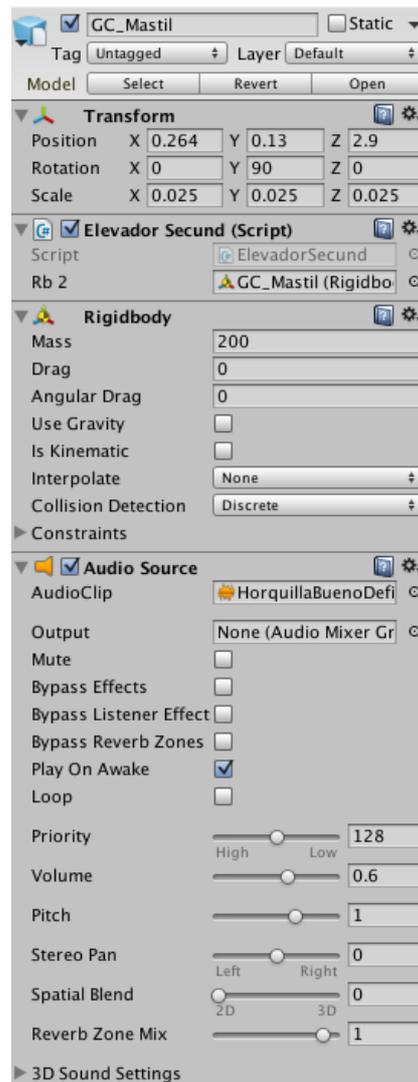


Fig. 52. Ventana de inspector de mástil

4.1.1.4. Inclinación de mástil y horquilla.

Para finalizar con la interacción entre mástil, horquilla y el usuario, se ha implementado el movimiento de inclinación hacia delante y hacia atrás del conjunto formado por el mástil y la horquilla.

Este modelo de carretilla permite una inclinación hacia atrás de 12°, mientras que la inclinación permitida hacia delante es de 6°, lo que facilita las operaciones de carga, traslado y descarga de mercancía.



Fig. 53. Inclinación conjunto mástil y horquilla hacia atrás

Para ello, se ha creado un script en el que ante la entrada del usuario, el GameObject que contiene al mástil y horquilla rota sobre el eje Z, utilizando la sentencia Rotate. Cada vez que el usuario pulsa el botón asignado como Fire3 o Fire4, el conjunto de mástil y horquilla rota 0.5° alrededor del eje Z.

```

1 using UnityEngine;
2 using System.Collections;
3
4 public class InclinacionMastilyHorquilla : MonoBehaviour {
5
6     // Use this for initialization
7     void Start () {
8
9     }
10
11    // Update is called once per frame
12    void Update () {
13
14        if (Input.GetAxis ("Fire3")>0)
15            transform.Rotate (0f, 0f, 0.5f);
16
17        if (Input.GetAxis ("Fire4")>0)
18            transform.Rotate (0f, 0f, -0.5f);
19    }
20 }
21
  
```

Fig. 54. Script Inclinación mástil y horquilla

4.1.1.5. Centro de gravedad.

Conseguir aplicar, a la carretilla elevadora, un comportamiento lo más realista posible del centro de gravedad ha sido primordial. El usuario debe experimentar en la escena de realidad virtual inmersiva los riesgos asociados a la manipulación de cargas de diferente masa en diferentes situaciones. Por tanto, este script se centra en la modificación del centro de gravedad de la carretilla en ausencia de carga, que Unity calcula automáticamente en función de la geometría, en función de la carga manipulada y de la altura a la que se circule con la misma, tal y como ocurre en la realidad con el denominado centro de gravedad combinado.

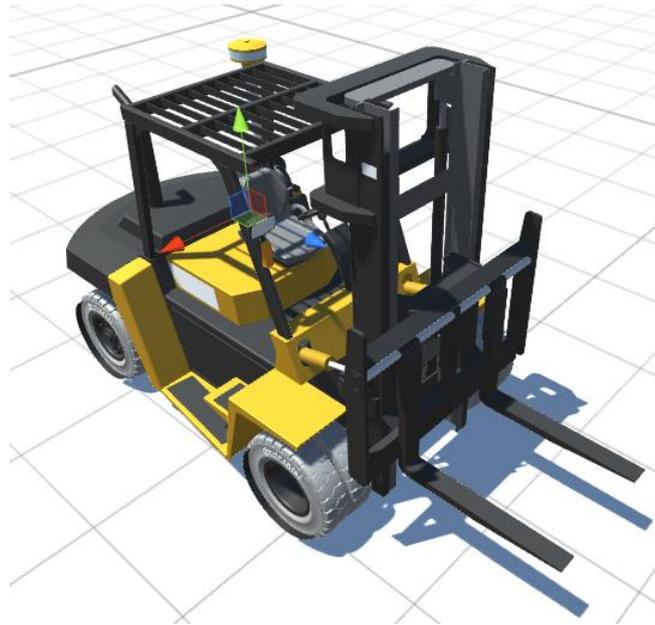


Fig. 54. Centro de gravedad de la carretilla elevadora en ausencia de manipulación de carga

En la escena de Unity se ha dotado a cada GameObject correspondiente a caja y palé de un Rigidbody estableciendo la masa de los mismos, dividiendo éstos, tal y como se ha establecido en capítulos anteriores, en cargas de 2500, 5000 y 8000 Kg.

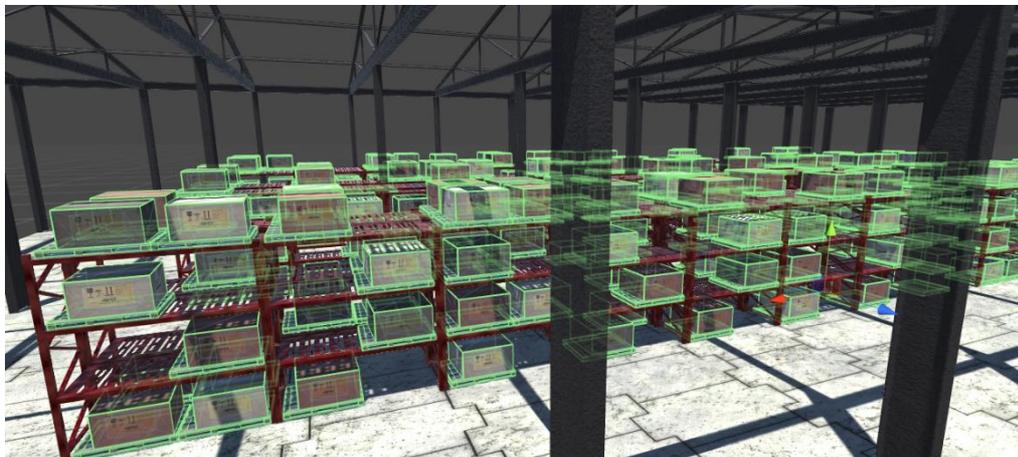


Fig. 55. Colliders aplicados a las cargas

Una vez asignado un Rigidbody a cada GameObject, se ha elaborado el script en el que se modifique el centro de gravedad de la carretilla en función de la carga que se manipule. Para ello, en primer lugar se han declarado las 230 variables Rigidbody que se han asignado, junto con el Rigidbody de la carretilla anteriormente definido para el script del motor.

4.1.1.7. Plataforma.

Para completar la experiencia de inmersión en la escena de realidad virtual, se necesita incorporar el funcionamiento de la plataforma a la escena de Unity, de manera que el movimiento que se produzca en la carretilla elevadora en la escena, como puede ser una situación de vuelco, se traduzca a movimiento en la plataforma. Para ello, se ha asignado el script en el que se describe el funcionamiento en el GameObject de la carretilla, para que sea únicamente el movimiento de la carretilla el que se experimente en la plataforma.

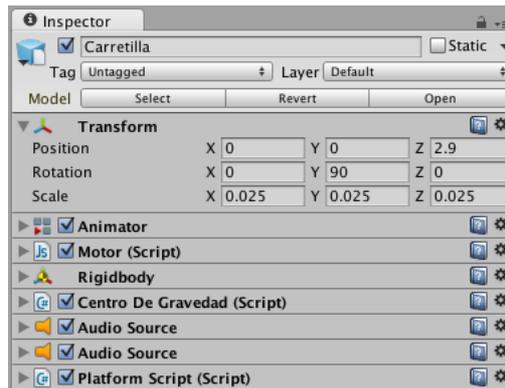


Fig. 58. Ventana de inspector del GameObject asociado a la carretilla incluyendo el script de la plataforma

4.1.2. Lugar de trabajo.

4.1.2.1. Nave industrial.

Del mismo modo que para la carretilla, una vez importado el modelo de la nave industrial se han aplicado los BoxColliders únicamente a los pilares de la misma, puesto que son las únicas partes que pueden interactuar con la carretilla elevadora mediante colisiones.

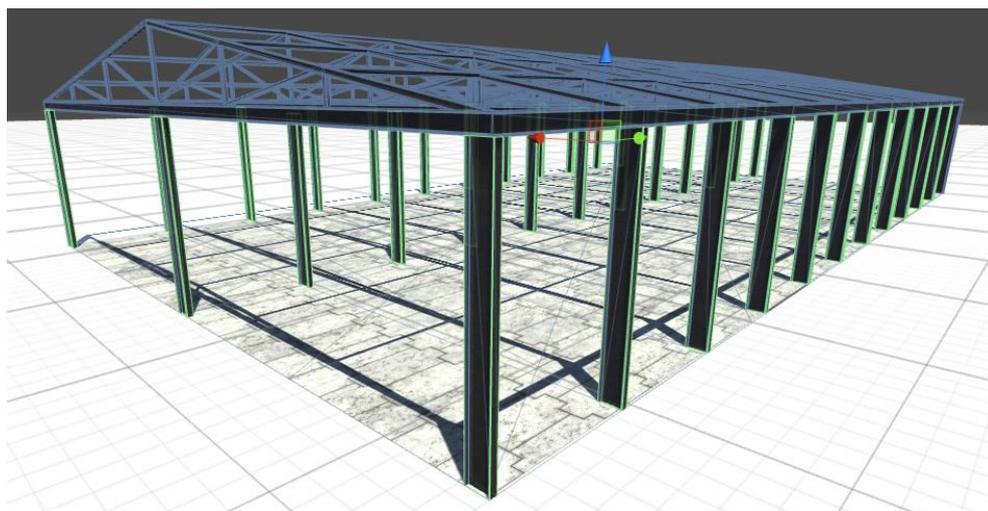


Fig. 59. Colliders en la nave industrial

A la nave industrial no ha sido preciso aplicarle Rigidbody ni script, puesto que no necesita recibir fuerzas ni interactuar con la escena ante una entrada del usuario.

4.1.2.2. Estanterías.

Al igual que en el resto de GameObjects, se han aplicado los BoxColliders a las estanterías, muy importantes en este caso, pues tienen que ser sólidas para sostener las cargas que se almacenen e interactuar con la carretilla cuando se proceda a la carga o descarga de las cargas junto con los palés.

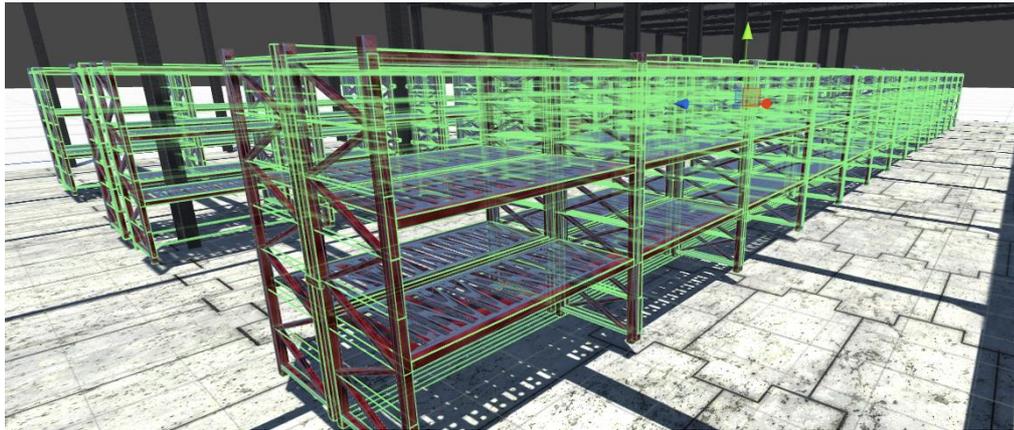


Fig. 60. Colliders en las estanterías metálicas

Del mismo modo que ocurre con la nave, las estanterías no precisan de ningún Rigidbody ni script, puesto que no hay que aplicarles ninguna fuerza ni interactuar con el usuario.

4.1.3. Edición de audio.

Una vez finalizada la asignación de scripts a los GameObjects que lo precisan, se han editado los archivos de audio correspondientes al motor, marcha atrás y horquilla, para posteriormente, asignarlos a los AudioSource creados en scripts. Cada archivo editado se exporta en formato mp3, formato que admite Unity.

4.1.3.1. Audio del motor de carretilla.

Originalmente, el audio de motor diésel de carretilla que se ha obtenido tiene una duración de 1:30 min aproximadamente.

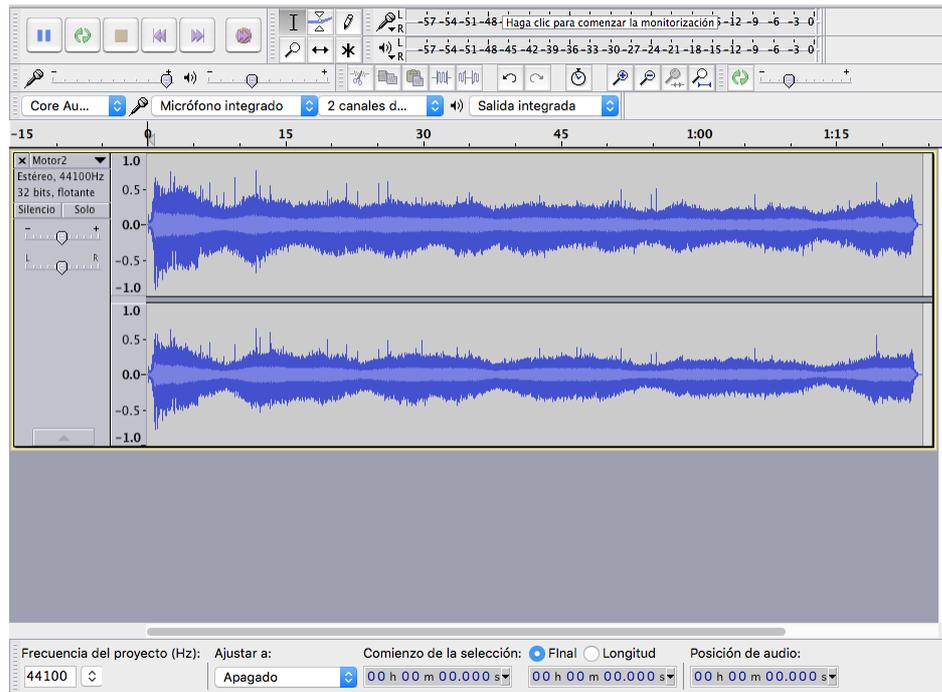


Fig. 61. Audio original del motor

Por tanto, se ha extendido el sonido para que no se produzcan cortes durante la simulación de la escena. Para ello, se ha eliminado la parte final del archivo en la que se apaga el motor, se ha seleccionado un tramo de audio adecuado, que no produzca saltos de tonalidad, se ha copiado y se ha pegado al final del audio. Se ha procedido según lo descrito de modo sucesivo hasta obtener un sonido de motor que dura dieciséis minutos, aproximadamente.

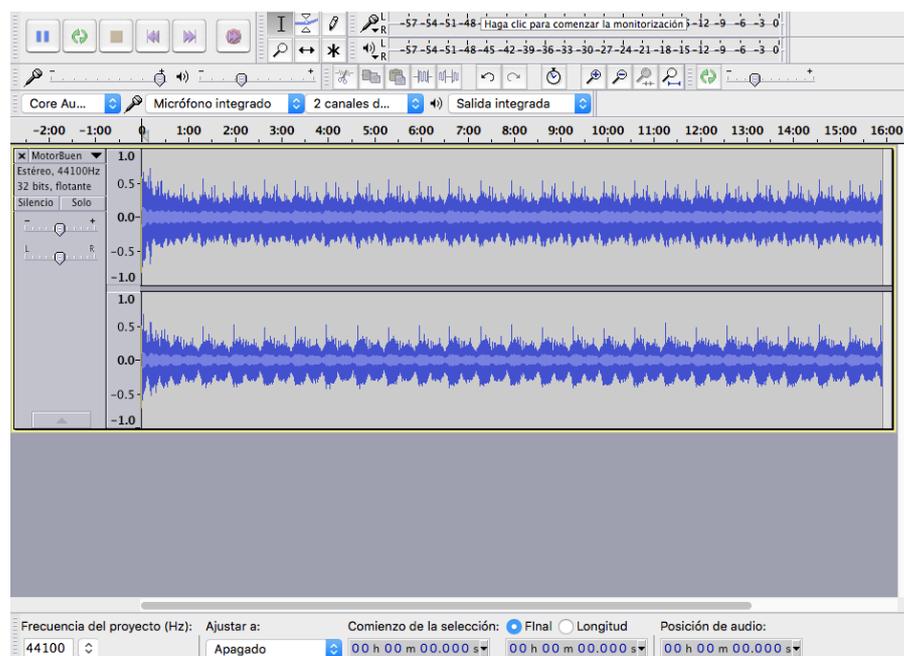


Fig. 62. Audio modificado del motor

4.1.3.2. Audio de marcha atrás de carretilla.

En lo que respecta al audio de marcha atrás de la carretilla elevadora, se ha tenido que elaborar prácticamente de cero a partir del sonido de un pitido encontrado que se ajusta al sonido característico de marcha atrás de los vehículos de manipulación de carga. El audio original encontrado ocupa apenas dos segundos.

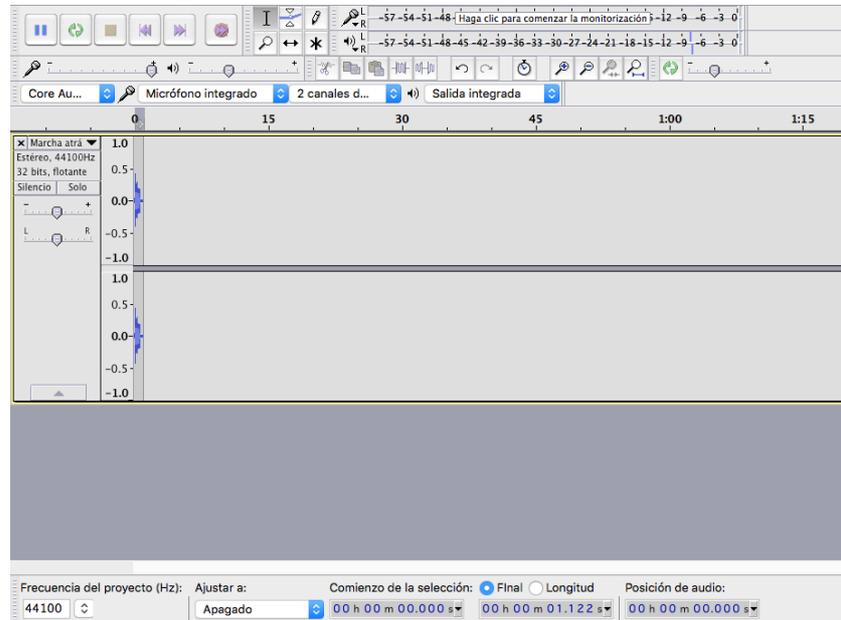


Fig. 63. Audio original de marcha atrás

Se ha repetido el mismo procedimiento llevado a cabo en el sonido del motor para extender la duración del archivo de audio hasta tener un sonido de marcha atrás de más de diez minutos.

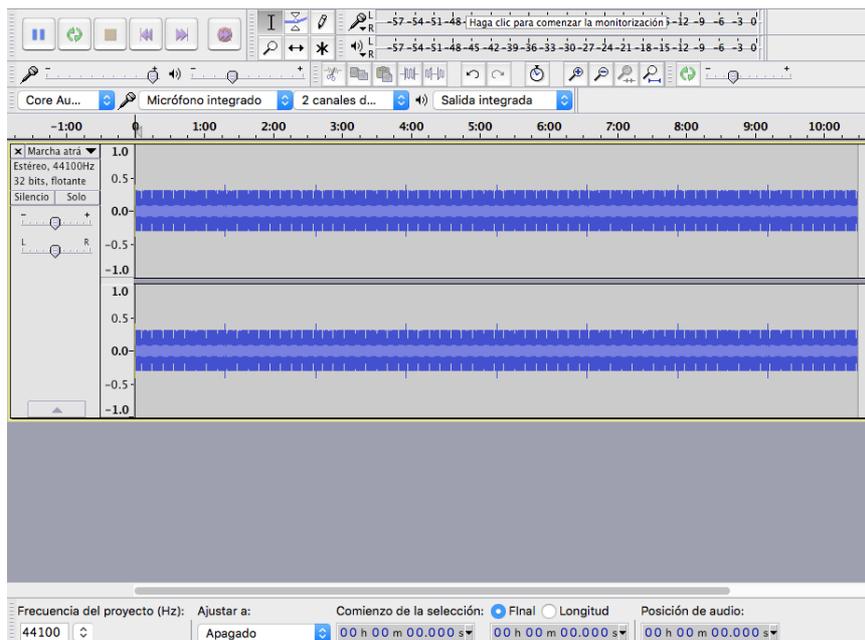


Fig. 64. Audio modificado de marcha atrás

4.1.3.3. Audio de horquilla y mástil de la carretilla elevadora.

Por último, en lo que a edición de audio se refiere, se ha editado el archivo de sonido de horquilla y mástil cuando se produce el descenso o elevación de los mismos. El archivo original obtenido tiene una duración de seis segundos, insuficientes para poder elevar o descender cargas pesadas, como las existentes en este proyecto.

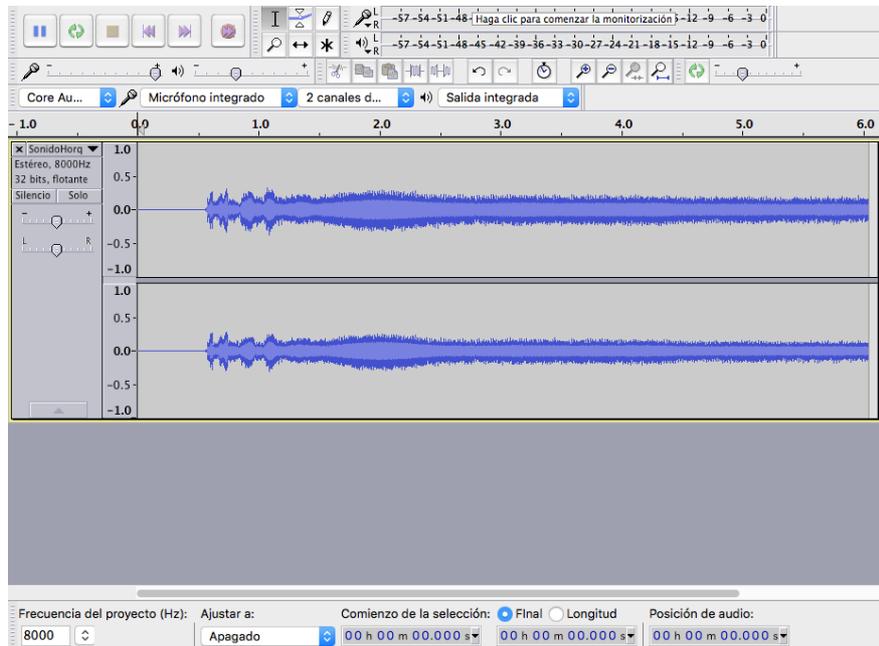


Fig. 65. Audio original horquilla y mástil

Se ha recortado el primer tramo en el que no hay sonido, para que éste se reproduzca instantáneamente ante la entrada del usuario. También se ha extendido la duración del audio, mediante el procedimiento descrito anteriormente, obteniendo una duración de doce segundos, suficientes para realizar carga o descarga satisfactoriamente.

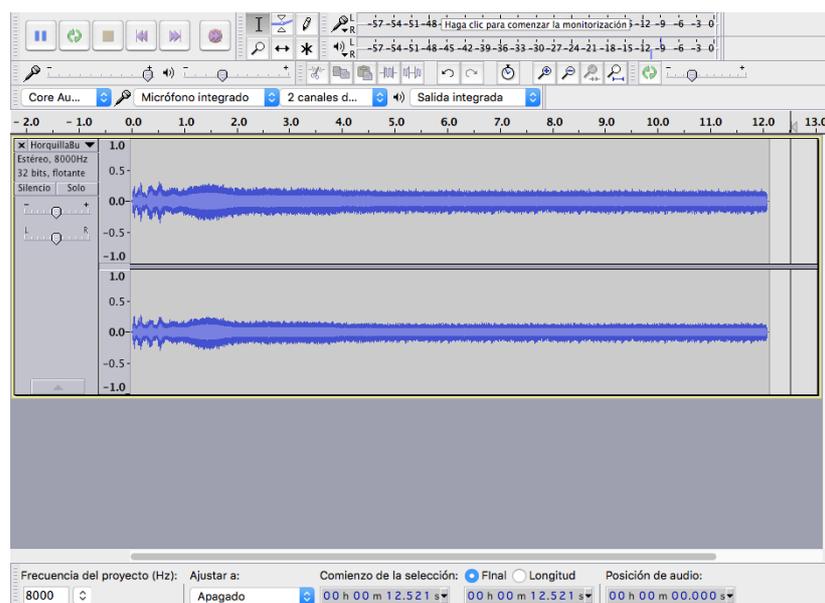


Fig. 66. Audio modificado horquilla y mástil

4.1.4. Escenario.

Una vez finalizada la edición y asignación de audio a los GameObjects que lo precisan, se ha elaborado el entorno que rodea a la carretilla y nave industrial, constituyendo la escena de realidad virtual inmersiva en su totalidad. El entorno que se pretende constituir es el de un polígono industrial en el que una carretilla elevadora tiene la misión de trasladar cargas de una nave industrial, donde se recibe mercancía y se almacenan de manera provisional, a otra nave industrial para su almacenamiento permanente.

4.1.4.1. Nave industrial.

En primer lugar se ha construido una acera en el entorno más próximo a la nave industrial para dotarla de un aspecto más realista.



Fig. 67. Aspecto final de la nave industrial

Seguidamente, para proporcionar a la escena aspecto de un polígono industrial y aportar profundidad, se añaden diversas naves industriales. La nave central se encuentra vacía, para que el usuario traslade allí toda la carga para su almacenamiento permanente.



Fig. 68. Composición de naves industriales

4.1.4.2. Carretera.

El siguiente paso ha consistido en crear la carretera en torno a las naves industriales, para ello se ha texturizado el plano que constituye el suelo de asfalto.

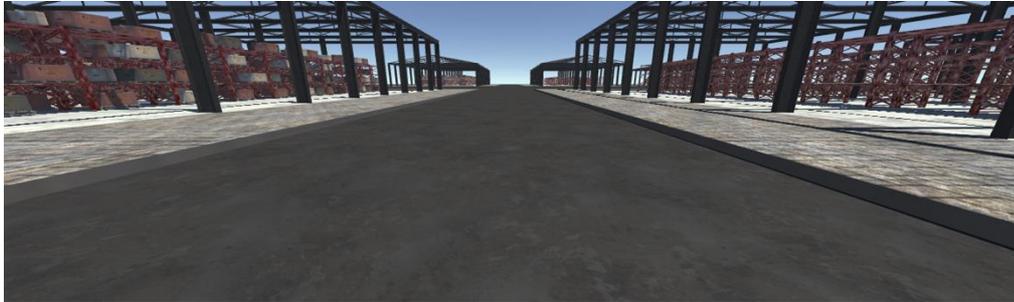


Fig. 69. Carretera

4.1.4.3. Muro.

Para que el entorno no sea demasiado extenso y la experiencia del usuario se centre en el traslado de cargas, situaciones en las que aparecen los riesgos asociados a la manipulación de carretillas elevadoras, se ha construido un muro de ladrillo visto que encierra tres naves.



Fig. 70. Muro que delimita la escena



Fig. 71. Miniatura de textura de ladrillo visto

4.1.4.4. Cielo.

Por último, para que la experiencia durante la simulación de la escena de realidad virtual sea lo más realista posible, se ha incluido cielo con nubes que tienen movimiento, tal y como ocurre en la realidad. Para ello, se ha descargado del Asset Store que proporciona Unity un paquete gratuito llamado Cloud System, que incluye distintos tipos de cielo y nubes para diferentes situaciones del día o climatología, junto con un script en lenguaje Java, que aplica movimiento a las nubes.

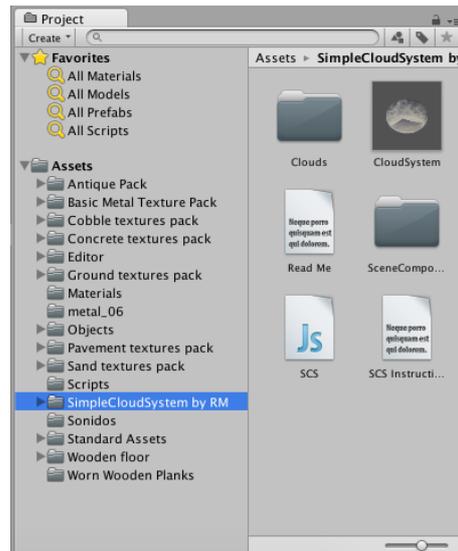


Fig. 72. Contenido de Cloud System en ventana de proyecto

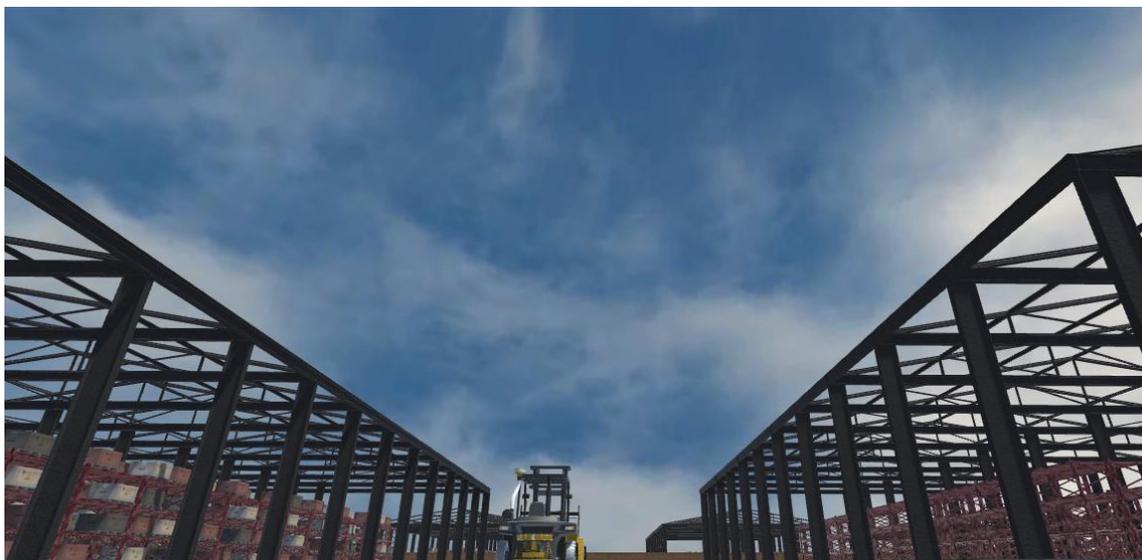


Fig. 73. Cielo insertado en la escena de realidad virtual

4.1.4.5. Iluminación.

Unity, por defecto, incluye en la ventana de jerarquía del menú inicial el GameObject de una fuente que aporta luminosidad a la escena denominada directional light. En el inspector se pueden modificar parámetros tales como

tonalidad, intensidad, ángulo de incidencia y posición de la fuente de iluminación, para poder adecuarla a la situación del día que se desea recrear.



Fig. 74. Ventana de inspector de la fuente de iluminación

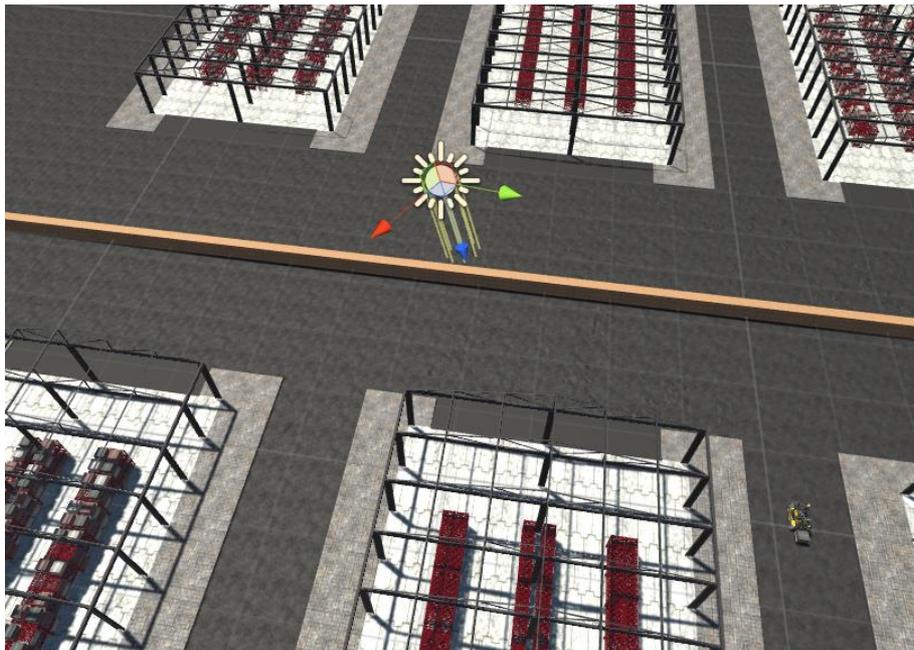


Fig. 75. Emplazamiento de la fuente de iluminación en la escena

4.1.4.6. Cámara.

Unity, al igual que ocurre con la iluminación, incluye por defecto un GameObject que contiene la cámara, en el menú de jerarquía al comenzar una escena. Lo que se ve a través de la cámara es lo que se va a visualizar a la hora de comenzar la simulación.

Se puede modificar la posición de la misma y asignarle scripts para que la cámara tenga un comportamiento determinado. En este caso, se ha

eliminado la cámara por defecto que aporta Unity y se ha incluido la cámara que aporta las gafas de realidad virtual Oculus Rift.

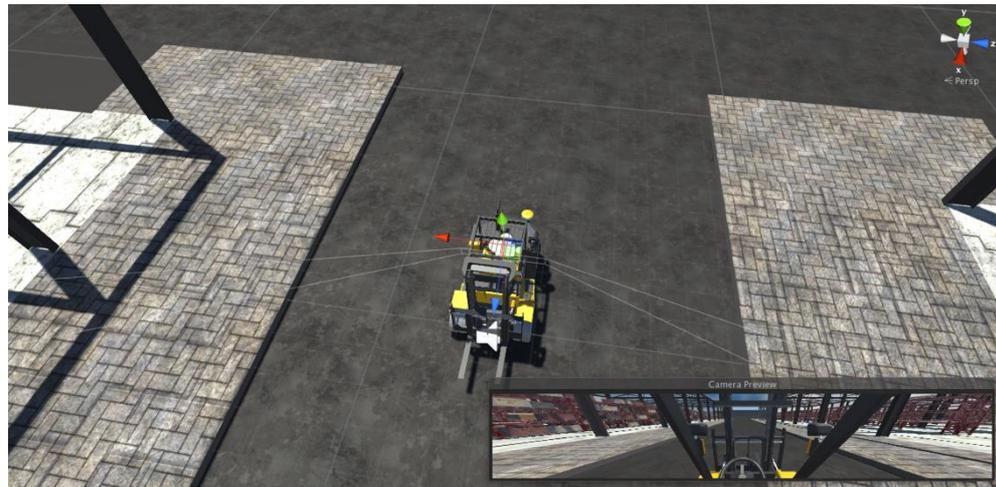


Fig. 76. Emplazamiento de la cámara ligada a Oculus Rift

Una vez implementado en Unity todo lo que se ha descrito, el aspecto final de la escena de realidad virtual inmersiva al siguiente.



Fig. 77. Aspecto final de la escena de realidad virtual inmersiva

4.2. Ensayos mecánicos.

Una vez se ha elaborado la escena de realidad virtual inmersiva, se ha sometido a la carretilla elevadora a situaciones límite de funcionamiento, dentro del entorno de la escena, para obtener en qué condiciones se produce el vuelco y poder establecer, así, una zona de seguridad de funcionamiento.

Se han llevado a cabo dos ensayos de conducción, manipulando cargas de distinta masa a distintas alturas para caracterizar el vuelco frontal y lateral que sufre la carretilla elevadora en estudio. Cada ensayo consta de cuatro simulaciones, evaluando la velocidad a la que se produce el vuelco para una carga y altura determinadas.



Fig. 78. Emplazamiento de la horquilla para los ensayos a realizar

4.2.1. Vuelco frontal.

Uno de los principales riesgos a la hora de manipular una carretilla elevadora consiste en el vuelco frontal, es decir, el vuelco que se produce, entre otras situaciones, en el frenado. Para caracterizar este riesgo en la carretilla que se está estudiando, se procede a realizar cuatro ensayos con cada uno de los tres tipos de carga, disponiéndolas a 4, 3, 2 y 1 m, respectivamente, pudiendo establecer, así, un rango de funcionamiento seguro en función de la carga manipulada. El ensayo que se lleva a cabo consiste en acelerar la carretilla en línea recta y determinar a qué velocidad, al pisar el pedal de freno, se produce el vuelco frontal.

4.2.1.1. Carga de 2500 Kg.

Se dispone en primer lugar una carga de 2500 Kg en la horquilla de la carretilla elevadora, y se han realizado simulaciones colocándola a cuatro, tres, dos y un metro de altura, respectivamente, tal y como se ha establecido anteriormente.



Fig. 79. Carga de 2500 Kg situada a 4 m del suelo

Los resultados obtenidos al realizar las simulaciones se adjuntan en la siguiente tabla.

Tabla 2. Resultados del ensayo de 2500 Kg en referencia al vuelco frontal

h (m)	v (Km/h)
4	17
3	20
2	22
1	25

Se han plasmado los datos en una gráfica para determinar la tendencia de los mismos.

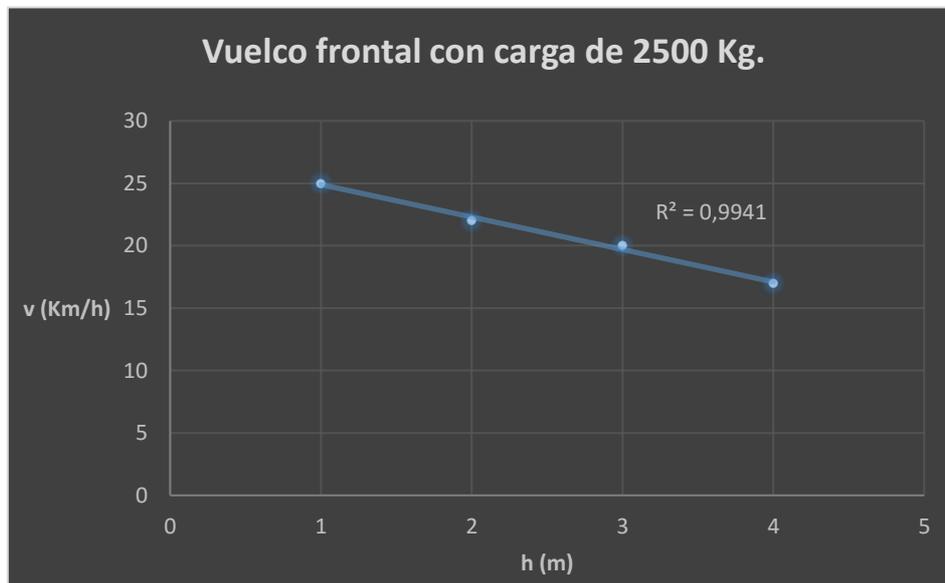


Fig. 80. Gráfica asociada al ensayo de 2500 Kg en referencia al vuelco frontal

Se ha observado que aplicando una regresión lineal a los datos obtenidos de la primera simulación se obtiene un coeficiente de determinación de 0.99412, por lo que se puede establecer que la tendencia de los datos obtenidos, en los que se produce vuelco frontal manipulando una carga de 2500 Kg, es totalmente lineal. La velocidad máxima admisible por la carretilla elevadora manipulando carga es de 22 Km/h, por lo que no se produce el vuelco frontal cuando la carga se dispone a un metro de altura.

4.2.1.2. Carga de 5000 Kg.

Seguidamente, se ha dispuesto una carga de 5000 Kg en la horquilla de la carretilla elevadora, y se ha realizado las simulaciones correspondientes, repitiendo el proceso anteriormente descrito.



Fig. 81. Carga de 5000 Kg situada a 3 m del suelo

Los datos obtenidos de la simulación se adjuntan en la tabla 3.

Tabla 3. Resultados del ensayo de 5000 Kg en referencia al vuelco frontal

h (m)	v (Km/h)
4	12
3	14
2	17
1	20

Del mismo modo que para el caso anterior, se ha elaborado una gráfica para obtener así, la tendencia de los datos obtenidos.

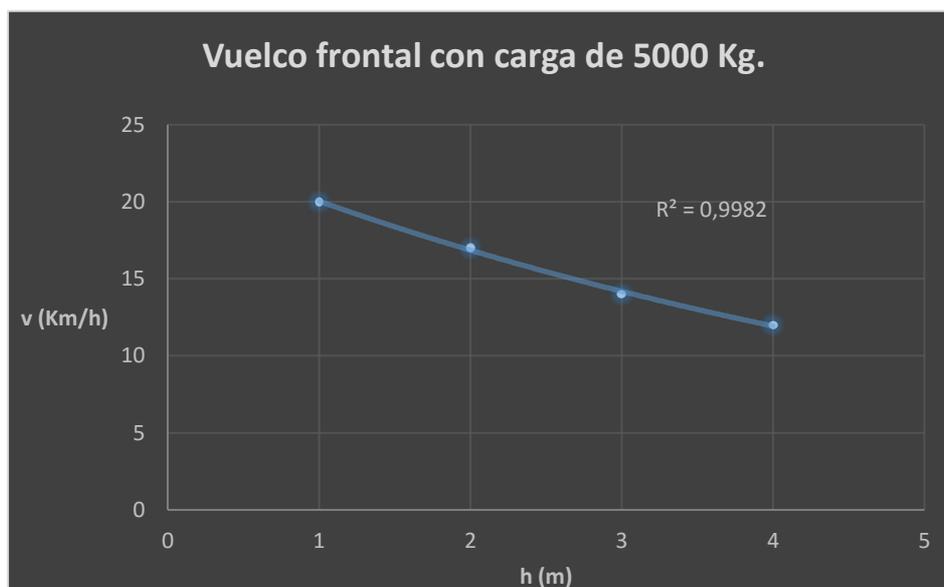


Fig. 82. Gráfica asociada al ensayo de 5000 Kg en referencia al vuelco frontal

Se ha observado que aplicando una línea de tendencia exponencial a los datos obtenidos en este ensayo, se obtiene un coeficiente de regresión de 0.99816, por lo que se puede confirmar que la tendencia de los datos obtenidos, en los que se produce vuelco frontal manipulando una carga de 5000 Kg, es exponencial.

4.2.1.3. Carga de 8000 Kg.

Por último, se ha dispuesto una carga de 8000 Kg, superior a lo admitido según la especificación de Caterpillar (7000Kg), en la horquilla de la carretilla elevadora, y se ha realizado el último ensayo para caracterizar el vuelco frontal de la carretilla elevadora en estudio.



Fig. 83. Carga de 8000 Kg situada a 2 m del suelo

Los datos obtenidos se adjuntan en la tabla 4.

Tabla 4. Resultados del ensayo de 8000 Kg en referencia al vuelco frontal

h (m)	v (Km/h)
4	2
3	2
2	2
1	2

En este caso la tendencia de los datos obtenidos es muy clara, pero para tener una mejor visualización de los mismos se elabora una gráfica que se adjunta a continuación.



Fig. 84. Gráfica asociada al ensayo de 8000 Kg en referencia al vuelco frontal

Se observa claramente que los datos poseen una tendencia totalmente constante, puesto que la carretilla vuelca frontalmente a 2 Km/h, indistintamente de la posición en la que se encuentre la carga.



Fig. 85. Inicio del vuelco frontal

4.2.2. Vuelco lateral.

Otro de los principales riesgos a tener en cuenta a la hora de manipular carretillas elevadoras consiste en el vuelco lateral, es decir, el vuelco que se produce al cambiar de dirección bruscamente. Para establecer un rango de funcionamiento seguro, en lo que a este riesgo se refiere, tal y como se ha hecho para caracterizar el vuelco frontal, se ha procedido del mismo modo. Se han realizado cuatro simulaciones con cada uno de los tres tipos de carga, disponiéndolas a las cuatro mismas alturas diferentes que en el ensayo anterior. El ensayo en cuestión consiste en acelerar la carretilla elevadora y observar a qué velocidad se produce el vuelco al producirse un giro brusco.

4.2.2.1. Carga de 2500 Kg.

Se ha procedido con la primera simulación de este ensayo, consistente en colocar una carga de 2500 Kg en la horquilla de la carretilla elevadora, y situarla a las diferentes alturas establecidas anteriormente.



Fig. 86. Carga de 2500 Kg situada a 1 m del suelo

Los resultados obtenidos en esta primera simulación se muestra en la tabla 5.

Tabla 5. Resultados del ensayo de 2500 Kg en referencia al vuelco lateral

h (m)	v (Km/h)
4	15
3	17
2	19
1	22

Para poder averiguar la tendencia que tienen los resultados de esta simulación y poder sacar conclusiones, se ha elaborado una gráfica, que se muestra en la figura 86.

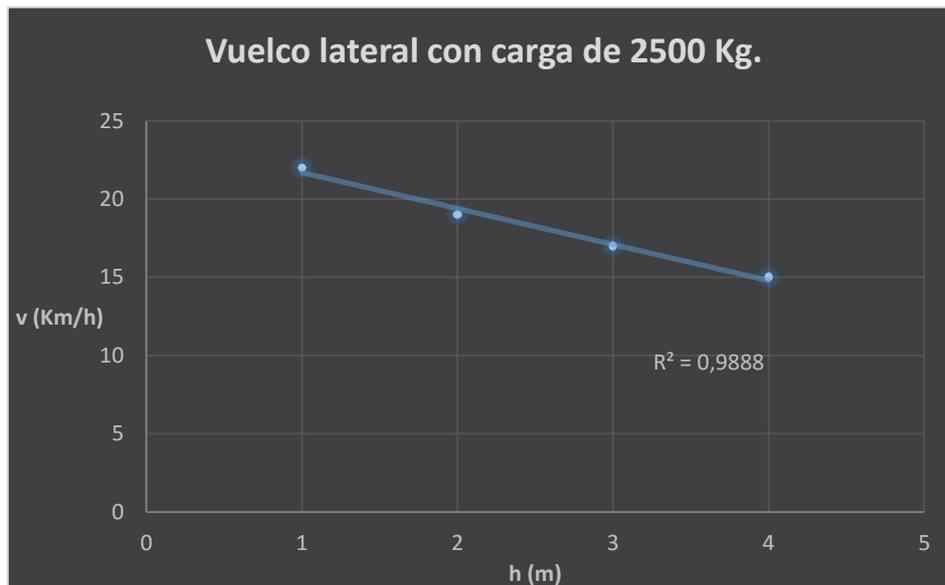


Fig. 87. Gráfica asociada al ensayo de 2500 Kg en referencia al vuelco lateral

Se puede observar, al aplicar una regresión lineal a los resultados obtenidos, que el coeficiente de regresión es de 0.98879, por lo que se puede confirmar que la tendencia de los resultados es lineal.

4.2.2.2. Carga de 5000 Kg.

Seguidamente, con la carga de 5000 Kg en la horquilla de la carretilla elevadora se han realizado las simulaciones correspondientes, repitiendo el proceso anteriormente descrito.



Fig. 88. Carga de 5000 Kg situada a 4 m del suelo

Los resultados y la gráfica asociada que se han obtenido se muestran en la figura 88 y la tabla 6.

Tabla 6. Resultados del ensayo de 5000 Kg en referencia al vuelco lateral

h (m)	v (Km/h)
4	11
3	14
2	17
1	19

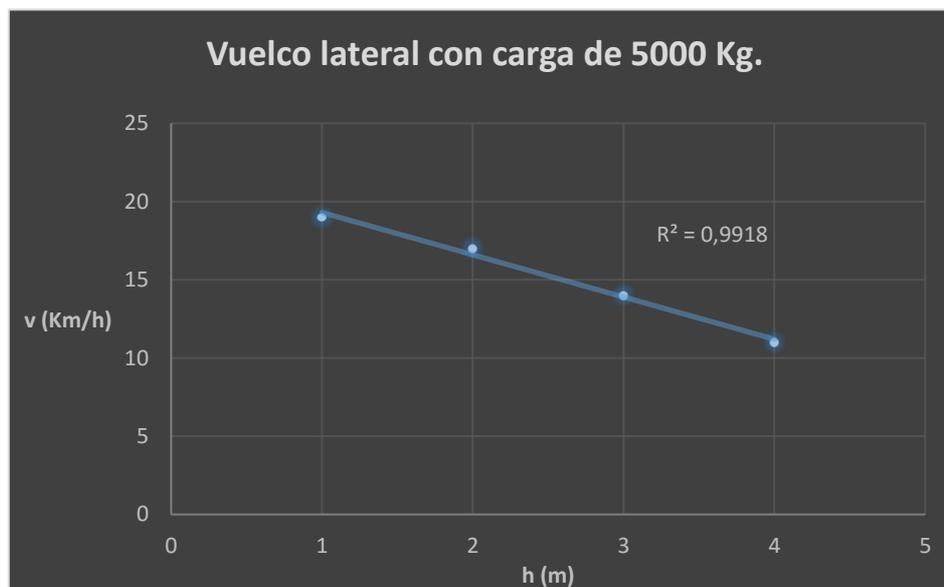


Fig. 89. Gráfica asociada al ensayo de 5000 Kg en referencia al vuelco lateral

Como se puede observar, al aplicar una regresión lineal a los resultados, se obtiene un coeficiente de regresión de 0.99184, por lo que se puede confirmar que la tendencia que poseen estos resultados es lineal.

4.2.2.3. Carga de 8000 Kg.

Por último, se procede a realizar las simulaciones pertenecientes a este último ensayo, que corresponde a la manipulación de una carga de 8000 Kg a diferentes alturas, tal y como se ha realizado en las simulaciones anteriores.

Al manipular la carga de 8000 Kg no se produce vuelco lateral, sino que la carretilla vuelca de manera frontal, tal y como ocurre en el ensayo anterior, puesto que el peso sobrepasa el valor máximo permitido en la especificación.



Fig. 90. Situación final ante el vuelco

5. Conclusiones.

Se ha llevado a cabo el modelado de diferentes componentes que forma parte de una nave industrial destinada al almacenamiento convencional paletizado (palés, cargas, estanterías metálicas, nave industrial y componentes estructurales añadidos al modelo de carretilla elevadora contrapesada) para poder implantar el funcionamiento real de la misma.

Se ha utilizado el programa de diseño CAD como es Solidworks 2014. Este software ha cumplido las expectativas previstas con éxito, ya que se ha podido modelar cada uno de los componentes y realizar su posterior ensamblaje en la escena de realidad virtual inmersiva.

Una vez modelado se ha adaptado el formato de los modelos exportados de Solidworks con el software 3ds Max para obtener el formato adecuado para importarlo a Unity. Este software ha servido de puente entre Solidworks y Unity.

Exportados los modelos a Unity, se ha realizado la texturización de los mismos y se ha elaborado la escena de realidad virtual inmersiva. Una vez elaborada la escena se ha puesto a prueba el funcionamiento de la carretilla mediante la realización de dos ensayos, para caracterizar el vuelco frontal y lateral de la carretilla elevadora que se ha estudiado, el modelo de Caterpillar DP70.

En lo que respecta al ensayo para caracterizar el vuelco frontal, se puede observar que si se manipula una carga de 2500 Kg a un metro de altura, no se produce el vuelco frontal, debido a que el centro de gravedad de la carretilla no se encuentra alejado del triángulo de equilibrio para producir el vuelco. Si se manipulan cargas de 8000 o 5000 Kg sí puede producir el vuelco frontal antes de llegar a la velocidad máxima establecida para manipular la carretilla elevadora con carga, 22 Km/h.

La tendencia obtenida en los resultados del primer ensayo, en lo referente a la carga de 2500 Kg, es lineal. Por tanto, se ha encontrado que existe una dependencia lineal entre

altura de la carga y velocidad de la carretilla para que se produzca el vuelco frontal de la misma.

En lo que respecta a las simulaciones manipulando una carga de 5000 Kg, la dependencia obtenida es exponencial, por lo que la dependencia que existe entre altura de carga y velocidad para producir el vuelco frontal de la carretilla es exponencial.

En cambio, manipulando cargas de 8000 Kg se obtienen unos resultados totalmente constantes, es decir, la carretilla vuelca frontalmente a la misma velocidad independientemente del emplazamiento de la carga. Esto se debe a que la masa de la carga en cuestión supera la máxima carga permitida por la especificación, 7000 Kg, situándose el centro de gravedad de la carretilla fuera del triángulo de equilibrio de manera espontánea. Sin embargo, si se manipulan cargas de 8000 Kg, no se produce vuelco lateral, puesto que el exceso de carga que sufre la carretilla elevadora provoca que el centro de gravedad se sitúe fuera del área del triángulo de equilibrio siempre por la parte delantera de la carretilla.

En lo referente al ensayo para caracterizar el vuelco lateral que se puede sufrir en la carretilla elevadora, se observa que, con carga de 2500 Kg se produce el vuelco lateral en todas las alturas ensayadas antes de llegar a la velocidad máxima admisible manipulando carga.

La tendencia de los resultados que se obtienen en este ensayo, tanto para 2500 como 5000 Kg, es lineal, lo que indica que existe una dependencia lineal entre el emplazamiento de la carga y la velocidad de la carretilla para que se produzca el vuelco lateral.

Una vez analizados los resultados obtenidos en cada ensayo se puede establecer que es más factible que aparezca una situación de vuelco lateral, puesto que, además de producirse a menor velocidad que el vuelco frontal, aparece en todas las situaciones estudiadas.

Un factor importante a tener en cuenta es que la velocidad a la que se producen ambos tipos de vuelco depende de la carga manipulada. Esto se debe a que cuanto más pesada es la carga, más rápido se desplaza el centro de gravedad de la carretilla fuera de los límites del triángulo de equilibrio.

Gracias a estas herramientas se han podido conseguir con éxito los objetivos del proyecto, logrando elaborar la escena de realidad virtual inmersiva y logrando experimentar en primera persona los riesgos más importantes al manipular una carretilla elevadora.

El uso de estos softwares pueden ayudar como herramienta multimedia para generar documentación y recursos con los que trabajar en prevención de riesgos laborales.

El proyecto da muestra de cómo la simulación de escenas y la realidad virtual son el camino que se abre a la prevención de riesgos laborales, tanto en el ámbito formativo como en la investigación de accidentes.

6. Bibliografía.

- Álvarez Núria, Enrich Carlos, Llena Sonia, Riera Luis. *El gran libro de 3d Studio Max 2009*. Editorial Marcombo. 2009.
- Aristizabal Remolina, María Jimena. *Simulación de accidente de vuelco de carretillas elevadoras*.
- Ayuda en línea de Unity.
<https://unity3d.com/es/community>
- Ayuda en línea de Solidworks 2014.
http://help.solidworks.com/2014/spanish/SolidWorks/sldworks/r_welcome_sw_online_help.htm
- Brazález Alfonso, Calleja Jesús, García Jon, García de Jalón Javier, Imaz Aitor, Larzabal Alberto, Mingo Iñigo, Rodríguez José Ignacio. *Aprenda Java*. Escuela Superior de Ingenieros. Universidad de Navarra. Año 2000.
<http://www4.tecnun.es/assignaturas/Informat1/AyudaInf/aprendainf/Java/Java2.pdf>
- Caterpillar. *DP70*.
http://www.cat.com/en_US/products/rental/site-support-equipment/forklifts-straight-mast/straight-mast-forklifts/10607864.html
- Ciaman formación. *Historia de la carretilla elevadora*.
<http://www.cursoscarretillerozaragoza.es/2013/08/06/historia-de-la-carretilla-elevadora/>
- Curso en español de C# para Unity. Youtube.
<https://www.youtube.com/watch?v=mugmZY41UMM>
- Fremap. *Innovaciones en tecnologías de simulación para mejorar la siniestralidad laboral en accidentes de tráfico*.
<http://www.prevencionfremap.es/actualidad/innovadoras-tecnologias-de-simulacion-para-mejorar-la-siniestralidad-laboral-en-accidentes-de-trafico.pdf>
- Instituto Nacional de Seguridad e Higiene en el trabajo. Real Decreto 1215/1997, de 18 de julio por el que se establecen las disposiciones mínimas de seguridad y salud para la utilización por los trabajadores de los equipos de trabajo. BOE no 188 07-08-1997.

- Logística, Transporte y Almacenaje. *Aumentan los accidentes laborales en el sector del transporte y almacenamiento*. 2016.
<http://www.logisticaytransporte.es/noticias.php/Aumentan-los-accidentes-laborales-en-el-sector-del-transporte-y-almacenamiento.-cl.-accidente-laboral/64713#>
- Meco Miami, Inc. Caterpillar DP70.
<http://www.mecomiami.com/preview.asp?lang=es&id=1376572207>
- Medea. Asociación de Empresas Distribuidoras de Equipos de Manutención.
<http://medea.org.es>
- Microsoft. *Manual del lenguaje C#*.
[https://msdn.microsoft.com/es-es/library/zkxk2fwf\(v=vs.90\).aspx](https://msdn.microsoft.com/es-es/library/zkxk2fwf(v=vs.90).aspx)
- Ministerio de Empleo y Seguridad Social. *Estadística de Accidentes de Trabajo*. 2015.
http://www.empleo.gob.es/es/estadisticas/monograficas_anuales/EAT/2015/index.htm
- Ministerio de Industria, Energía y Turismo. Secretaría general de Industria y de la pequeña y mediana empresa. *Temas de interés en Calidad y Seguridad Industrial*.
<http://www.f2i2.net/legislacionseguridadindustrial/SeguridadIndustrial.aspx>
- Mitsubishi. *Elevación total*.
<http://www.elevaciontotal.es/?A=ST&E=1001>
- Monge Malave, David. *Seguridad en el manejo de carretillas elevadoras*.
<http://www.grupotp.org>
- Oculus Rift.
<https://www.oculus.com>
- Simumak. *La prevención de riesgos laborales con simuladores*.
<http://simumak.com/es/la-prevencion-de-riesgos-laborales-con-simuladores>
- Unity.
<https://unity3d.com/es>
- Universidad Politécnica de Cartagena. Centro de Desarrollo e Innovación Tecnológica.
http://www.upct.es/contenido/perfil_empresa/CEDIT.php

7. Anexo I.

Especificaciones técnicas de la carretilla Caterpillar DP70.



DP60 - DP70

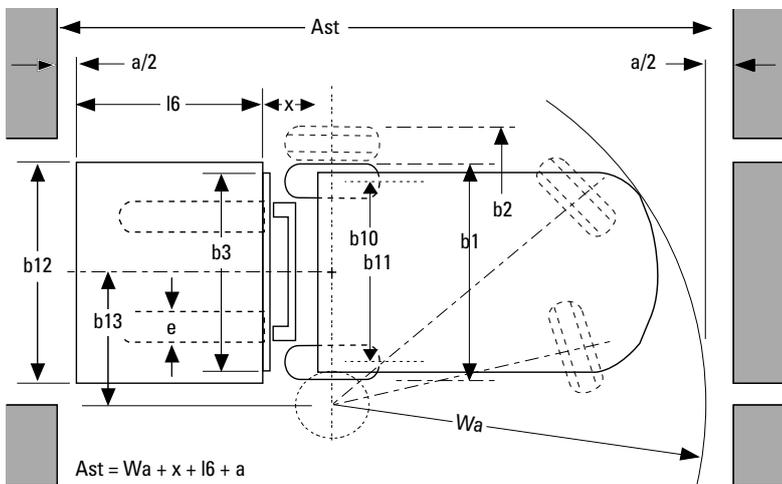
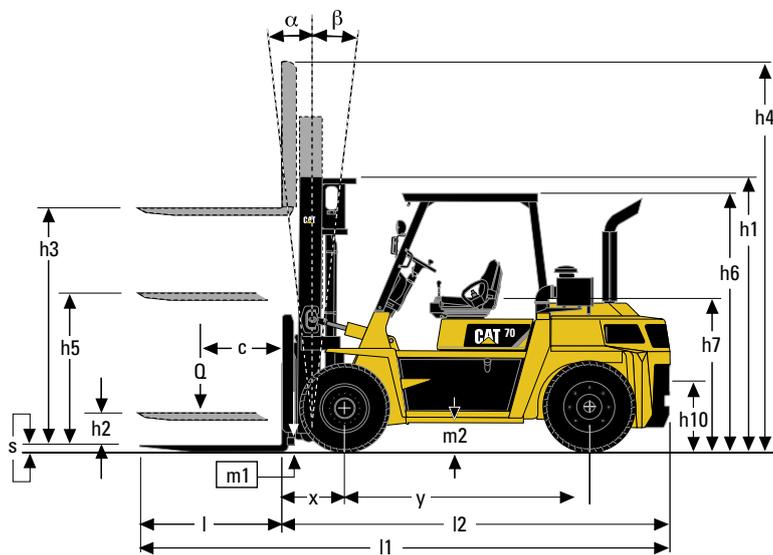
Specifications

Engine powered lift trucks

6.0 - 7.0 tonnes

Characteristics			Cat Lift Trucks	Cat Lift Trucks
1.01	Manufacturer (abbreviation)		DP60	DP70
1.02	Manufacturer's model designation		Diesel	Diesel
1.03	Power source: (battery, diesel, LP gas, petrol)		Seated	Seated
1.04	Operator type: pedestrian, (operator)-standing, -seated			
1.05	Load capacity	Q (kg)	6000	7000
1.06	Load centre distance	c (mm)	600	600
1.08	Load distance, axle to fork face	x (mm)	585	585
1.09	Wheelbase	y (mm)	2300	2300
Weight				
2.01	Truck weight, without load / including battery (simplex mast, lowest lift height)	kg	8715	9325
2.02	Axle loading with maximum load, front/rear (simplex mast, lowest lift height)	kg	13255 / 1460	14660 / 1665
2.03	Axle loading without load, front/rear (simplex mast, lowest lift height)	kg	4160 / 4555	4050 / 5275
Wheels, Drive Train				
3.01	Tyres: V=solid, L=pneumatic, SE=solid pneumatic - front/rear		L / L	L / L
3.02	Tyre dimensions, front		8.25 x 15- 12PR	8.25 x 15- 12PR
3.03	Tyre dimensions, rear		8.25 x 15- 12PR	8.25 x 15- 12PR
3.05	Number of wheels, front/rear (x=driven)		4x / 2	4x / 2
3.06	Track width (centre of tyres), front	b10 (mm)	1650	1650
3.07	Track width (centre of tyres), rear	b11 (mm)	1650	1650
Dimensions				
4.01	Mast tilt, forwards/backwards	α/β °	6 / 12	6 / 12
4.02	Height with mast lowered (see tables)	h1 (mm)	3070	3070
4.03	Free lift (see tables)	h2 (mm)	200	200
4.04	Lift height (see tables)	h3 (mm)	4000	4000
4.05	Overall height with mast raised	h4 (mm)	5280	5280
4.07	Height to top of overhead guard	h6 (mm)	2420	2420
4.08	Seat height	h7 (mm)	1350	1350
4.12	Tow coupling height	h10 (mm)	485	485
4.19	Overall length	l1 (mm)	4810	4855
4.20	Length to fork face (includes fork thickness)	l2 (mm)	3590	3635
4.21	Overall width	b1/b2 (mm)	2175	2175
4.22	Fork dimensions (thickness, width, length)	s / e / l (mm)	60 x 150 x 1220	60 x 150 x 1220
4.23	Fork carriage to DIN 15 173 A/B/no		IV A/No	IV A/No
4.24	Fork carriage width	b3 (mm)	1780	1780
4.31	Ground clearance under mast, with load	m1 (mm)	140	140
4.32	Ground clearance at centre of wheelbase, with load (forks lowered)	m2 (mm)	255	255
4.33	Working aisle width with 1000 x 1200 mm pallets, crosswise	Ast (mm)	5095	5180
4.34	Working aisle width with 800 x 1200 mm pallets, crosswise	Ast (mm)	4895	4980
4.35	Turning circle radius	Wva (mm)	3310	3395
4.36	Minimum distance between centres of rotation	b13 (mm)	1260	1260
Performance				
5.01	Travel speed, with/without load	km/h	23.0 / 27.5	22.0 / 27.5
5.02	Lifting speed, with/without load	m/s	0.44 / 0.47	0.44 / 0.47
5.03	Lowering speed, with/without load	m/s	0.55 / 0.55	0.55 / 0.55
5.05	Rated drawbar pull, with/without load	N	42800 / 28700	42700 / 27700
5.07	Gradeability, with/without load	%	31 / 29	27 / 27
5.09	Acceleration time (10 metres) with/without load	s	-	-
5.10	Service brakes (mechanical/hydraulic/electric/pneumatic)		Hydraulic	Hydraulic
IC Engine				
7.01	Manufacturer / Type		S6S-T	S6S-T
7.02	Rated output to ISO 1585	kW	63.9	63.9
7.03	Rated speed to DIN 70 020	rpm	2300	2300
7.04	Number of cylinders / cubic capacity	cm ³	6 / 4996	6 / 4996
7.05	Fuel consumption according to VDI 60 cycle	l/h / kg/h	-	-
Miscellaneous				
8.01	Type of drive control		Torque converter	Torque converter
8.02	Maximum operating pressure for attachments	bar	191	191
8.03	Oil flow for attachments	l/min	70	70
8.04	Noise level, value at operator's ear (EN 12053)	dB(A)	-	-
8.05	Towing coupling design / DIN type, ref.		-	-





A_{st} = Working aisle width with load
 a = Safety clearance (200 mm)
 l_6 = Pallet length (800 or 1000 mm)
 b_{12} = Pallet width (1200 mm)

Lower Cost of Ownership

- Robust chassis and heavy duty steer axle invite long life and high performance.
- The powerful 6 cylinder S6S-T diesel engine is designed to the highest standards. It is quiet and fuel efficient. This engine meets EURO level IIIA emission regulations.
- Load sensing hydraulic control valve saves energy for efficient operation.

Unmatched Productivity

- The small diameter steering wheel and full hydrostatic steering assistance ensure precise and rapid steering response with little effort.
- Use of small diameter lift cylinders and strategic locating of hydraulic lines ensure excellent forward visibility through the wide mast.
- For safe and efficient braking, these models are fitted with vacuum assisted hydraulic brakes.
- Easy service keeps your machine in sound working condition; routine maintenance is performed quickly: Engine, radiator, coolant recovery bottle, engine oil dipstick, fuel shut-off valve, air cleaner and battery are all within easy view and reach.
- The comprehensively equipped instrument panel provides valuable on-the-go information and warning signals to the driver, allowing maximum control.
- Tough rubber engine mounting blocks and U-jointed drive shafts virtually eliminate vibrations to the operator's compartment.
- Two-speed automatic transmission.
- Rapid access to the transmission and torque converter.
- Axle and mast lubrication points are easy to reach. Mast and carriage load rollers are permanently lubricated.
- The capacity of the air cleaner has been increased, lengthening service life.

Safety and Ergonomics

- Adjustable full suspension seat with bolster support, hip restraint and seat belt.
- Ergonomic operation using Lever Control or Fingertip Control, allowing the operator to control hydraulics precisely and with minimum effort.
- Access to the operator's compartment is easy thanks to the conveniently placed grab bars and footsteps.
- The tiltable steer column, adjustable full suspension seat and generous foot and leg room ensure a relaxed driving position for all operators.
- The trunnion mounted, oscillating steer axle ensures a comfortable ride and excellent ground contact - even on the poorest work surfaces.
- Tough gas cylinders ensure effortless raising of the engine hood.

Options

- Lever hydraulic control (electric over hydraulic)
- Wide range of sideshifter and fork positioner
- Fully enclosed cabin with/without air conditioning

Cat® Lift Trucks.

Your partner in materials handling.

Mast Type	DP60 - DP70					DP60	DP70
	h3 mm	h1 mm	h4 mm	h2 mm	h5 mm	Q @ 600mm kg	Q @ 600mm kg
Simplex	3000	2570	4280	200	-	6000	7000
	3300	2720	4580	200	-	6000	7000
	3500	2820	4780	200	-	6000	7000
	3700	2920	4980	200	-	6000	7000
	4000	3070	5280	200	-	6000	7000
	4500	3320	5780	200	-	6000	7000
	5000	3670	6280	200	-	6000	7000
	5500	3920	6780	200	-	5800	6700
	6000	4170	7280	200	-	5600	6400
	6500	4420	7780	200	-	5400	6200
Triplex	7000	4670	8280	200	-	5000	5800
	3700	2375	4980	-	1095	5500	6200
	4000	2475	5280	-	1195	5500	6200
	4300	2575	5580	-	1295	5500	6200
	4500	2645	5790	-	1365	5500	6200
	4700	2725	5980	-	1445	5400	6200
	5000	2825	6280	-	1545	5300	6200
	5500	3075	6780	-	1795	5100	6000
	6000	3245	7290	-	1965	4900	5700
	6500	3425	7780	-	2145	4700	5400
	7000	3575	8280	-	2295	4300	4600
	7500	3745	8790	-	2485	3600	3800
	8000	3925	9280	-	2645	2900	3100



Mast Performance and Capacity

h1	Height with mast lowered
h2	Standard free lift
h3	Standard lift height
h4	Height with mast raised
h5	Full free lift
Q	Lifting capacity, rated load
c	Load centre (distance)

Consult your dealer for the maximum back tilt allowed to obtain the capacities specified.

catliftruck@mcfenl
www.catliftruck.com

CESC1083(08/08)ok
 Copyright ©2008, MCFE. All rights reserved.
 CATERPILLAR, CAT, their respective logos, 'Caterpillar Yellow' and the 'Power Edge' trade dress, as well as corporate and product identity used herein, are trademarks of Caterpillar and may not be used without permission.
 Printed in The Netherlands

NOTE: Performance specifications may vary depending on standard manufacturing tolerances, vehicle condition, types of tyres, floor or surface conditions, applications, or operating environment. Trucks may be shown with non-standard options. Specific performance requirements and locally available configurations should be discussed with your Cat lift trucks Dealer. Cat Lift Trucks follows a policy of continual product improvement. For this reason, some materials, options and specifications could change without notice.

