

DISEÑO DE UNA APLICACIÓN BASADA EN REALIDAD VIRTUAL INMERSIVA
QUE PERMITA EVIDENCIAR INFRACCIONES COMETIDAS AL MOMENTO DE
CONducIR.

MARCO ALEXANDER PACHÓN CUMBE
DTE.MPACHON@PEDAGOGICA.EDU.CO
CÓD.: 2009103042

CRISTHIAN CAMILO MONROY UNIVIO
DTE115_CMONROY@PEDAGOGICA.EDU.CO
CÓD.: 2007203045

UNIVERSIDAD PEDAGÓGICA NACIONAL
FACULTAD DE CIENCIA Y TECNOLOGÍA
DEPARTAMENTO DE TECNOLOGÍA
LICENCIATURA EN ELECTRÓNICA
BOGOTÁ, D.C. FEBRERO DE 2016

DISEÑO DE UNA APLICACIÓN BASADA EN REALIDAD VIRTUAL INMERSIVA
QUE PERMITA EVIDENCIAR INFRACCIONES COMETIDAS AL MOMENTO DE
CONducIR.

MARCO ALEXANDER PACHÓN CUMBE
CRISTHIAN CAMILO MONROY UNIVIO

TRABAJO DE GRADO PARA OPTAR AL TÍTULO DE LICENCIADOS EN
ELECTRÓNICA.

DIEGO MAURICIO RIVERA
DIRECTOR

UNIVERSIDAD PEDAGÓGICA NACIONAL
FACULTAD DE CIENCIA Y TECNOLOGÍA
DEPARTAMENTO DE TECNOLOGÍA
LICENCIATURA EN ELECTRÓNICA
BOGOTÁ, D.C. FEBRERO DE 2016

Nota de aceptación

Firma del Director del trabajo de Grado.

Firma del Jurado.

Firma del Jurado.

Bogotá D.C, 18 de Febrero de 2016

AGRADECIMIENTOS

Luego de la culminación de éste proyecto, que nos ha llenado de gran orgullo y satisfacción por el gran aprendizaje que hemos adquirido en su desarrollo, queremos dar inicio a este documento referenciando a todas aquellas personas que contribuyeron a su realización.

En primera instancia agradecemos a Dios quien nos proporcionó sabiduría, paciencia y constancia en este trabajo. En segundo lugar a nuestras familias y amigos quienes con su aliento y motivación nos dieron fuerza para seguir adelante en el desarrollo de la aplicación. Finalmente agradecemos al Licenciado Diego Rivera por su paciencia y confianza en la culminación del proyecto además de sus grandes aportes para el cumplimiento los objetivos propuestos.

RESUMEN ANALÍTICO EN EDUCACIÓN - RAE

1. Información General	
Tipo de documento	Trabajo de Grado
Acceso al documento	Universidad Pedagógica Nacional. Biblioteca Central
Título del documento	DISEÑO Y CREACIÓN DE UNA APLICACIÓN BASADA EN REALIDAD VIRTUAL INMERSIVA QUE PERMITA EVIDENCIAR INFRACCIONES COMETIDAS AL MOMENTO DE CONDUCIR.
Autor(es)	Monroy, Cristhian Camilo; Pachón, Marco Alexander;
Director	Rivera, Diego;
Publicación	Bogotá, Universidad Pedagógica Nacional, 2016. 91 p.
Unidad Patrocinante	Universidad Pedagógica Nacional.
Palabras Claves	Aplicación de conducción, infracciones de tránsito, realidad virtual inmersiva.

2. Descripción
<p>Se desarrolló una aplicación 3D basada en realidad virtual inmersiva denominada "TUTRÁNSITO" cuyo objetivo es enfrentar al usuario a una serie de retos relacionados a la forma de conducir los cuales están ligados a las normas de tránsito y seguridad vial. Ésta herramienta permite evidenciar infracciones típicas que se comenten diariamente al conducir y desencadenan diferentes situaciones que se reflejan en los agentes que intervienen.</p> <p>La aplicación "TUTRÁNSITO" fue concebida empleando tecnologías basadas en sprites bidimensionales y para desplegar en plataforma PC con viabilidad de distribución en Android e IOS. El usuario estará limitado al manejo de un vehículo en donde se conduce sobre las vías del escenario y cuyo fin radica en ir de un punto a otro sorteando en su trayecto diversos desafíos, que al ser superados, permitirán el acceso a un nuevo nivel de la aplicación. El éxito en cada uno de los niveles propenderá de la información oportuna que tenga el usuario acerca de la normatividad de tránsito y su formación en seguridad vial.</p>

3. Fuentes

Jorge, A.N.; Godoy Del Sol, H.; Ortis, M., "Caracterización de la mortalidad por accidentes del tránsito con participación de ciclos: un problema sociomédico," en *MediSur*. 2010, vol. 8, no. 4, pp. 57-62. Jul.-Ago. 2010. ISSN 1727-897X.

Pérez, F.J., "Presente y Futuro de la Tecnología de la Realidad Virtual," en *Creatividad y Sociedad*. 2011, vol., no.16, pp.1-39. Mar. 2011.

JONES, H. (1995): *Virtual reality applications*. Londres, Academic Press. Gálvez Mozo, A. (2004) *Posicionamientos y puestas en pantalla. Un análisis de la producción de sociabilidad en los entornos virtuales*. Barcelona: UAB.

Urkle, S. (1997) *La vida en la pantalla. La construcción de la identidad en la era de internet*. Barcelona: Paidós.

Bakar, N.A.A.; Zulkifli, A.N.; Mohamed, N.F.F., "The use of multimedia, Augmented Reality (AR) and Virtual Environment (VE) in enhancing children's understanding of road safety," in *Open Systems (ICOS), 2011 IEEE Conference on* , vol., no., pp.149-154, 25-28 Sept. 2011 doi: 10.1109/ICOS.2011.6079288

«Manifiesto por el Desarrollo Ágil de Software». [En línea]. Disponible en: <http://www.agilemanifesto.org/iso/es/>. [Accedido: 14- ene -2016].

L. Rising y N. S. Janoff, «The Scrum software development process for small teams», *IEEE Software*, vol. 17, n.o 4, pp. 26-32, ago. 2000.

H. Takeuchi y I. Nonaka, «The new new product development game», *Harvard Business Review*, 1986.

Club de Desarrolladores Stuff. Metodología SCRUM. [en línea]. 2008. [Consultado 15 de

Enero de 2016]. Disponible en: <http://www.agilemanifesto.org/iso/es/>

Klump, R., "Understanding object-oriented programming concepts," in *Power Engineering Society Summer Meeting, 2001* , vol.2, no., pp.1070-1074 vol.2, 15-19 July 2001 doi: 10.1109/PESS.2001.970207

4. Contenidos

El documento está compuesto por 8 capítulos. En el capítulo 1 Introducción se presenta el esquema del documento, se realiza una descripción de la problemática en forma detallada, la justificación y pertinencia para realizar el proyecto, por qué se elige la herramienta que se va a utilizar, se describen cuáles son los objetivos que se plantearon para su consecución y las delimitaciones del mismo. A continuación, en el capítulo 2, se recopila y detallan los principales trabajos y proyectos que contribuyeron al desarrollo de la aplicación, además, se puntualizan las temáticas pertinentes que permiten el proceso de desarrollo del proyecto y se realiza una conceptualización de los principales componentes teóricos necesarios para cada una de las fases desarrolladas en éste trabajo. En el capítulo 3 se detalla cada uno de los aspectos preliminares para el desarrollo, lo que permite responder a la pregunta ¿Qué se hizo?; de la misma manera se puntualizan los aspectos particulares concernientes a la metodología, generalidades de la aplicación, etapas de diseño, fases de desarrollo, mecanismos de funcionamiento y proceso de visualización. En los Capítulos 4 y 5 se detallan cada uno de los pasos ejecutados en el desarrollo de la aplicación, lo que permite ahondar en la cuestión de ¿Cómo se elaboró el proyecto?; aquí se puntualizan las técnicas usadas en cada una de las fases, se describen los software informáticos que se utilizaron, se desarrolla la parte de modelado y arte así como los requerimientos de programación. El capítulo 6 presenta una descripción detallada de las actividades y situaciones que se recrearon en la aplicación y que el usuario debe trabajar. Finalmente el capítulo 7 y 8 muestran algunos resultados y observaciones así como las conclusiones y referencias empleadas.

5. Metodología

La aplicación “TUTRÁNSITO” como proyecto se adaptó y elaboró bajo la metodología SCRUM, reconocida por su eficacia de trabajo en equipo, su adaptación a fallas y requerimientos de cambio y su versatilidad de desarrollo en ambientes complejos. Esta metodología cimienta sus bases en las metodologías ágiles caracterizadas por modelos iterativos e incrementales. Dicha metodología se caracteriza por el trabajo en equipos organizados estratégicamente los cuales trabajan en fases u etapas que sufren modificaciones a medida que avanza el desarrollo y ejecución del proyecto lo que posibilita mejoras significativas relacionadas a la reducción de tiempos en la creación y desarrollo sin afectar la calidad del software.

Esta característica principal de la metodología SCRUM permite que los requerimientos que se presenten en cada una de las fases del proyecto se vayan sustrayendo, mejorando o agregando de acuerdo al progreso en las iteraciones, de la misma manera, corrigiendo fallos y problemas que se puedan ocurrir. Finalmente, el Scrum master, para nuestro caso el director de proyecto, cumplió un papel importante para la correcta implementación de la metodología y poder obtener los resultados esperados en base a los objetivos propuestos.

6. Conclusiones

1. Se desarrolló una aplicación basada en realidad virtual inmersiva denominada “tutrasito” que permite evidenciar infracciones típicas que se comenten diariamente al conducir y que desencadenan diferentes situaciones que se reflejan en los agentes que intervienen. La herramienta cumple con los objetivos propuestos logrando describir y evidenciar algunos comportamientos de personas que diariamente conducen algún tipo de automotor y que no acatan las normas y señales de tránsito estipuladas por la ley.
2. Al ser la seguridad vial una problemática latente en la actualidad, tanto que sea

una asignatura obligatoria dentro de las instituciones educativas, hace de esta aplicación una herramienta necesaria y atrayente para los agentes que intervienen en dichas dinámicas, que posibilita espacios de meditación, reflexión y concientización de las consecuencias de sus actos. La culminación de la aplicación “TUTRÁNSITO” demuestra la viabilidad de narración de problemáticas de índole social a partir de la implementación de herramientas alternativas e innovadoras como ésta.

3. Debido al tiempo estipulado para la entrega de la aplicación no fue posible realizar un estudio de la población a la cual podría ser dirigida así como los resultados que arrojaría en dicha población, además para poder observar la aceptación de este tipo de herramientas en diferentes contextos, sin embargo, La aplicación “TUTRÁNSITO” está dispuesta a cambios y modificaciones a futuro en todas sus etapas que contribuyan a un mejoramiento de la misma en cuanto a desempeño, eficacia, rendimiento, etc.
4. Para trabajos futuros en cuanto a la aplicación, es posible pensar en nuevas actividades que permitan incluir otras situaciones diferentes a las contempladas en éste proyecto de la misma manera, el trabajo queda abierto para aquellos estudiantes y personas en general que quieran realizar una implementación realizando un estudio más concienzudo de la población a la cual quiere ser dirigida la herramienta y que permita generar estudios potencialmente más elaborados sobre este tipo de problemáticas.

Elaborado por:	Monroy, Cristhian Camilo; Pachón, Marco Alexander;
Revisado por:	Rivera, Diego;

Fecha de elaboración del Resumen:	18	02	2016
--	----	----	------

TABLA DE CONTENIDO

1	Introducción	1
1.1	Descripción del problema.....	2
1.2	Justificación.....	3
1.3	Objetivos.	5
1.3.1	General.....	5
1.3.2	Específicos	5
1.4	Delimitación.....	7
2	Marco teórico.....	10
2.1	Antecedentes.	10
2.1.1	Simulaciones de tránsito.....	10
2.1.2	Realidad virtual y conducción.	12
2.1.3	Videojuegos sobre tránsito y seguridad vial.	13
2.2	Marco Conceptual.	16
2.2.1	Realidad virtual (RV).....	16
2.2.2	Motores de videojuegos.....	24
2.2.3	Normas de tránsito.	25
2.2.4	Metodologías para diseño de software.	28
3	DISEÑO DE LA APLICACIÓN	31
3.1	Generalidades de la aplicación.	31
3.1.1	Investigación, análisis y propuesta de la aplicación.....	31
3.1.2	Historia	32
3.1.3	Planificación.....	33
3.1.4	Etapas de diseño.	35
3.1.5	Fases de desarrollo.	36
3.2	Mecanismos de funcionamiento.	40
3.3	Proceso de visualización.	43
4	DESARROLLO DE LA APLICACIÓN.	45
4.1	Modelado y arte.	45
4.1.1	Diagramación, esquematización y geometría 3D	46
4.1.2	Digitalización.....	47

5	PROGRAMACIÓN DE LA APLICACIÓN.....	55
5.1	Requerimiento de programación.....	55
5.2	Animación.....	55
5.2.1	Control general de la aplicación.....	55
5.2.2	Control de vehículos.....	56
5.2.3	Control de semáforos.....	56
5.2.4	Control de notificaciones.....	57
5.2.5	Control de interfaces.....	57
5.3	Referencia de los scripts.....	58
5.4	Descripción de los Scripts principales.....	59
6	DISEÑO E IMPLEMENTACIÓN DE LAS ACTIVIDADES O SITUACIONES DE MOVILIDAD.....	63
6.1	Descripción de las actividades.....	63
6.2	Referencia de niveles y actividades.....	68
7	RESULTADOS Y OBSERVACIONES.....	70
7.1	Pruebas de funcionamiento.....	70
7.2	Presentación de resultados.....	70
8	CONCLUSIONES Y TRABAJO FUTURO.....	73
	Referencias de libros y artículos.....	75

Lista de tablas

Tabla 1. Comparación de modelos de desarrollo de software.	29
Tabla 2. Características de la aplicación.	34
Tabla 3. Requisitos para desarrollo de la aplicación.....	38
Tabla 4. Referencia de Scripts utilizados en la programación de la aplicación.....	58
Tabla 5. Resumen de la descripción de actividades.	68

Lista de figuras

Figura 1. Situación 1: Vehículo parqueado en vía principal.....	8
Figura 2. Situación 2: No respetar la fila para el giro.....	8
Figura 3. Situación 3: No respetar la demarcación de bloqueo de cruce.	9
Figura 4. Situación 4: No respetar la demarcación de línea continua.....	9
Figura 5. Captura de imagen de una simulación de tránsito.....	11
Figura 6. Imagen capturada de herramienta de comprensión espacial basada en sistemas de visión artificial.	13
Figura 7. Pantallazo del videojuego sobre el SITP y sus mecánicas.....	14
Figura 8. Imagen del videojuego Tráfico Santiago de Cali.	15
Figura 9. Mapa conceptual sobre realidad virtual.	17
Figura 10. Captura del menú principal de la aplicación.	41
Figura 11. Imagen de notificación de éxito en el nivel.	42
Figura 12. Imagen de notificación de infracción cometida.	42
Figura 13. Captura de imagen del vehículo controlado explorando el escenario.	43
Figura 14. Arte inicial de la aplicación.	45
Figura 15. Arte final de la aplicación.....	46
Figura 16. Croquis o plano de la ciudad modelada en SketchUp.	47
Figura 17. Modelado inicial del escenario.....	48
Figura 18. Modelado de semáforo doble.	48
Figura 19. Primer modelo 3D de semáforo sencillo.	49
Figura 20. Semáforo sencillo mejorado.	49
Figura 21. Modelado de vías y señales de tránsito en SketchUp.	50
Figura 22. Escenario final de la aplicación.	50

Figura 23. Modelos finales de automóviles.	51
Figura 24. Automóvil controlado.	52
Figura 25. Vehículo autónomo.	52
Figura 26. Integración del escenario y los objetos en la escena.	53
Figura 27. Captura del fondo utilizado para el menú de inicio de la aplicación ...	54
Figura 28. Actividad 1. Giro en esquinas por el carril correcto.	63
Figura 29. Imagen relacionada a la actividad de giro.	63
Figura 30. Actividad 2. Imagen cambio de carril en línea continua.	65
Figura 31. Actividad 4. Imagen de atasco y vehículos sobre la intersección.	66
Figura 32. Imagen de parqueo indebido sobre vías principales.	67

CAPÍTULO 1.

1 INTRODUCCIÓN

El presente documento corresponde al informe de proyecto de grado en el cual se desarrolló el diseño y creación de una aplicación basada en realidad virtual inmersiva que posibilite evidenciar infracciones al momento de conducir. Éste propuso una alternativa de trabajo en respuesta al problema de disponer de una herramienta que permitiera concienciar al usuario de ciertas infracciones que se cometen al conducir y todas las implicaciones que conllevan para el resto de agentes que intervienen.

El trabajo de grado se planteó con el interés de generar el uso de nuevas tecnologías que permitan, por un lado, la interacción del usuario con la herramienta, y por otro lado, la visualización de diferentes infracciones de tránsito y situaciones de movilidad que se comenten en la cotidianidad, de forma didáctica. Por este motivo, se promueve el uso de la realidad virtual con las herramientas que la posibilitan, entre las que se encuentran las gafas de realidad virtual, los dispositivos de entrada (teclado de computadora, joystick), dispositivos de salida (audífonos, altavoces), entre otros.

El desarrollo del proyecto incluye una descripción teórica de cada uno de los elementos utilizados para su realización, entre los que se encuentran generalidades sobre realidad virtual, motores de videojuegos, Unity 3D, seguridad vial, entre otros; así como de las herramientas utilizadas para la visualización e interacción del usuario con la aplicación. Adicionalmente se describen las situaciones que se quieren mostrar a través de la aplicación y las mecánicas utilizadas para su elaboración.

1.1 DESCRIPCIÓN DEL PROBLEMA.

Hoy día se habla sobre el uso de una gran cantidad de herramientas tecnológicas que permiten evidenciar diversos fenómenos que se presentan diariamente así como de emular algunos de la realidad. Dentro de estas herramientas se encuentran aquellas que trabajan a partir de la realidad virtual y cuya finalidad es mostrar el mundo cotidiano a partir de una representación aproximada de mismo.

Cotidianamente se evidencian diversos tipos de complicaciones que se generan por las malas conductas de personas al volante, desde trancones, hasta accidentes de tránsito que, en ocasiones, dejan víctimas fatales [1]. Es indiscutible, que en cuestiones de movilidad, las señales de tránsito juegan un papel primordial en este tipo de situaciones, así como es irrefutable el hecho de decir que un gran porcentaje de la población desconoce las diferentes señales de tránsito así como otro porcentaje, no menos importante, evade dichas señales con pleno conocimiento de ellas, sin olvidar aquellos conductores cuya forma de conducir evoca los comportamientos agresivos de los cavernícolas de la antigüedad. Actualmente, los conductores cuentan únicamente con escuelas para infractores a las cuales se envían todos aquellos que han cometido algún tipo de infracción con el fin de realizar una capacitación o curso que permita concienciar a estos de todas aquellas faltas que comenten al conducir. Por otro lado, de acuerdo al documento registrado por el Ministerio de Educación y basados en la ley N° 1503 del 29 de Diciembre de 2011¹ la cual promueve la educación en seguridad vial, es indispensable la enseñanza de las normas de tránsito, así como en hábitos, comportamientos y conductas desde las instituciones educativas de Colombia.

En base a las situaciones descritas anteriormente se observa la necesidad de generar nuevos aportes educativos que permitan reconocer y entender las normas de tránsito, por tal motivo, este proyecto busca fomentar conciencia a los

¹ <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=45453>

conductores y la población en general de distintas infracciones que se cometen al conducir, así como la contribución de la seguridad vial para la disminución de accidentalidad por desconocimiento tanto de normas de tránsito como de hábitos, comportamientos y conductas al manejar.

1.2 JUSTIFICACIÓN.

La realidad virtual viene de años atrás en donde surgió como una idea de recrear un entorno virtual de la realidad a partir de pantallas y capsulas, en la actualidad, y con ayuda de nuevas tecnologías, permite pasar de ser espectadores de un videojuego o una animación a ser exploradores de nuevos ambientes creados virtualmente.

En lo más marcado de estos inventos tecnológicos surgieron las Oculus Rift² lanzadas en 2012 y que tomaron fuerza a mediados del 2013 consolidando la pertinencia de un nuevo mercado que no se encuentra explotado en su totalidad, esta idea de casco virtual no tardó mucho en llamar la atención de compañías que fabrican los Smartphone (celulares inteligentes) y otras compañías como Facebook y Google buscando una alternativa más asequible para que cualquier persona tenga un dispositivo de realidad virtual a su alcance. En lo más económico del mercado se encuentran las Cardboard³ fabricadas por Google siendo un prototipo en cartón con dos lentes de distancia focal y biconvexas que soportan un celular de 5 pulgadas el cual actúa como un sistema inmersivo, donde el usuario se sentirá dentro del mundo virtual.

Siendo la realidad virtual una tendencia mundial de desarrollo tanto para el entretenimiento como para otras aplicaciones, es de vital importancia darla a conocer e implementarla, ya que los avances realizados en otros campos han sido muy exitosos y significativos, tal es el caso de la medicina como se afirma a continuación:

² <https://www.oculus.com/en-us/>

³ <https://www.google.com/get/cardboard/>

“El campo de la terapia mediante realidad virtual ha progresado significativamente. Se han publicado diversos estudios controlados que demuestran la eficacia de esta nueva tecnología en el tratamiento de las fobias específicas, y en este momento existen ya estudios que informan de resultados preliminares, prometedores, en el tratamiento de trastornos más complejos como el trastorno de pánico con agorafobia, el trastorno de estrés postraumático o los trastornos de la conducta alimentaria.” (Botella, García, Rosa, & Quero, 2007)

De la misma manera en el campo de la educación se observan avances significativos en el uso de la realidad virtual como nos describe el siguiente fragmento:

“La realidad virtual en el proceso de enseñanza aprendizaje tiene grandes contribuciones, puesto que obviamente facilita el aprendizaje ya que, como se ha mencionado anteriormente, los estudiantes viven el aprendizaje de manera activa. Este tipo de aprendizaje mejora la calidad de la enseñanza, ya que no es lo mismo que el educador brinde ejemplos de diversas situaciones a que el estudiante las presencie de una manera artificial en el aula. Tal como se puede apreciar cuando los estudiantes aprenden algún contenido porque el hecho de que la manera como lo interiorizaron era atractiva, llamativa y dinámica. Así sucede muchísimas veces con la implementación de juegos, en donde los educandos son participantes activos y por ende aprenden de una manera inconsciente.” (Salas, 2014)

En relación a los artículos anteriores se observa que la realidad virtual está involucrada en diversos campos de aplicación, lo cual permite ahondar en situaciones que no se han tenido en cuenta y son de vital importancia dentro de una sociedad [2]. A partir de este proyecto se utilizaron dispositivos que permitieron la experimentación de nuevas tecnologías para fortalecer el campo educativo. Gracias del uso de estas herramientas, el usuario logró evidenciar algunas infracciones comunes que se cometen al conducir, puesto que la aplicación, además de permitirle interactuar de manera inmersiva, fue diseñada

para que esté propenso a cometer dichas infracciones. De la misma manera, tuvo la oportunidad de retroalimentar su desempeño dentro de la aplicación a partir de mensajes informativos donde se aclaró cuáles fueron las faltas cometidas.

1.3 OBJETIVOS.

1.3.1 General

- Diseñar una aplicación basada en realidad virtual inmersiva que recree situaciones típicas al conducir un vehículo para hacer evidentes las posibles infracciones que se comenten.

1.3.2 Específicos

- A. Programar las mecánicas de interacción del usuario con la aplicación a través de comandos del teclado y movimientos de la cabeza que controlen el vehículo.
- B. Diseñar un modelo de ciudad que permita representar las situaciones de movilidad y posibles infracciones de tránsito.
- C. Programar los comportamientos de autonomía de los vehículos que se mueven dentro de la ciudad.
- D. Diseñar las situaciones de movilidad para que el conductor este propenso a cometer infracciones.

En relación al objetivo general, la aplicación fue programada en su totalidad en el motor de videojuegos Unity 3D cuya versión personal, además de ser gratuita, brinda las herramientas necesarias para realizar desarrollos de alto nivel y competitivos de una manera versátil y sencilla. Por otro lado, la dinámica de desarrollo de la aplicación se realiza vía scripting lo que permite la creación de ésta con un acabado profesional. Adicionalmente, al ser un motor multiplataforma,

Unity 3D acepta exportar el desarrollo a distintas plataformas como IOS e incluso Android, base fundamental de este proyecto.

Cabe la pena aclarar, en cuanto a los objetivos específicos, que fueron desarrollados y documentados cada uno en capítulos diferentes, de esta manera, el primer objetivo se documentó en el capítulo 3, el segundo objetivo en el capítulo 4, el tercero en el capítulo 5 y el cuarto en el capítulo 6 respectivamente.

Ya refiriéndonos a los objetivos específicos, en primera instancia, se han programado las mecánicas de interacción del usuario con la aplicación a través de las teclas de desplazamiento o dirección (cursores) del PC, las cuales recrean los movimientos del vehículo que controla el usuario. Con respecto a la visualización de la aplicación, se hizo a través de gafas de realidad virtual cuyo principio de funcionamiento es generado a partir del movimiento de la cabeza.

Por otro lado, se realizó el modelamiento del escenario en el Software CAD Google SketchUp en donde se crearon las calles, las carreteras, los edificios, los semáforos, las señales de tránsito y otros detalles que hicieron parte del ambiente. Al terminar el modelo del escenario se procede a exportar dichas construcciones al motor de videojuego Unity 3D. Cabe la pena resaltar que los diseños de cada uno de los objetos que componen la ciudad fueron realizados por nosotros y no nos centramos específicamente en el aspecto visual sino en la funcionalidad de la aplicación, es decir, en las dinámicas que permitían que éste trabajara adecuadamente.

Posterior al diseño del modelo del escenario, se procede a realizar la programación de las funciones de cada uno de los elementos que componen la aplicación. Objetos como semáforos, vehículos autónomos, vehículos controlados, y sus respectivas decisiones, son programados vía scripting, los cuales se realizan a través de lenguaje de programación orientado a objetos más específicamente C Sharp (C#). Una vez finalizada la programación de los objetos que componen el escenario y ya funcionando de manera autónoma, se proceden a implementar las situaciones de movilidad previamente diseñadas y distribuidas en diferentes

sectores del escenario. Finalmente, una vez terminada la aplicación, se realizan las respectivas pruebas de funcionamiento interno con el fin de detectar posibles mejoras y modificaciones a los requerimientos de la aplicación y en concordancia a los objetivos propuestos y la aprobación tanto del experto como del equipo de trabajo.

1.4 DELIMITACIÓN.

Esta aplicación le permite al usuario una interacción de realidad virtual, en la cual se controla el automóvil a partir del teclado de un computador, al mismo tiempo, se encuentra en un sistema inmersivo en el entorno virtual que admite visualizar cualquier ángulo en tiempo real a través de las gafas de realidad virtual. La delimitación, propuesta desde el anteproyecto, da una mirada a las situaciones específicas que se recrearon en la aplicación, sin embargo, no son las únicas que se observan, se diseñaron condiciones de movilidad adicionales y otras están implícitas dentro de las dinámicas del proceso de conducción.

En este proyecto se recrearon 4 situaciones particulares que son ejemplos cotidianos en grandes ciudades. Se emularon las infracciones que se muestran en las ilustraciones posteriores, además, en ellas están implícitas algunas otras infracciones comunes como pasarse el semáforo en rojo o manejar en contravía, que aunque no son tan reiteradas, no dejan de ser importantes en cuanto a seguridad vial hace referencia. Finalmente, es necesario precisar que las imágenes son de nuestra autoría, además, que tampoco se contemplarán peatones u otros tipos de vehículos diferentes a automóviles. Las situaciones específicas que se modelaron son:



Figura 1. Situación 1: Vehículo parado en vía principal.

En la *Figura 1* se observa una vía principal de doble carril en el sentido que indican los vehículos, sobre ésta se encuentran parquedados los vehículos de color blanco los cuales infringen la norma de no estacionarse en vía principal.

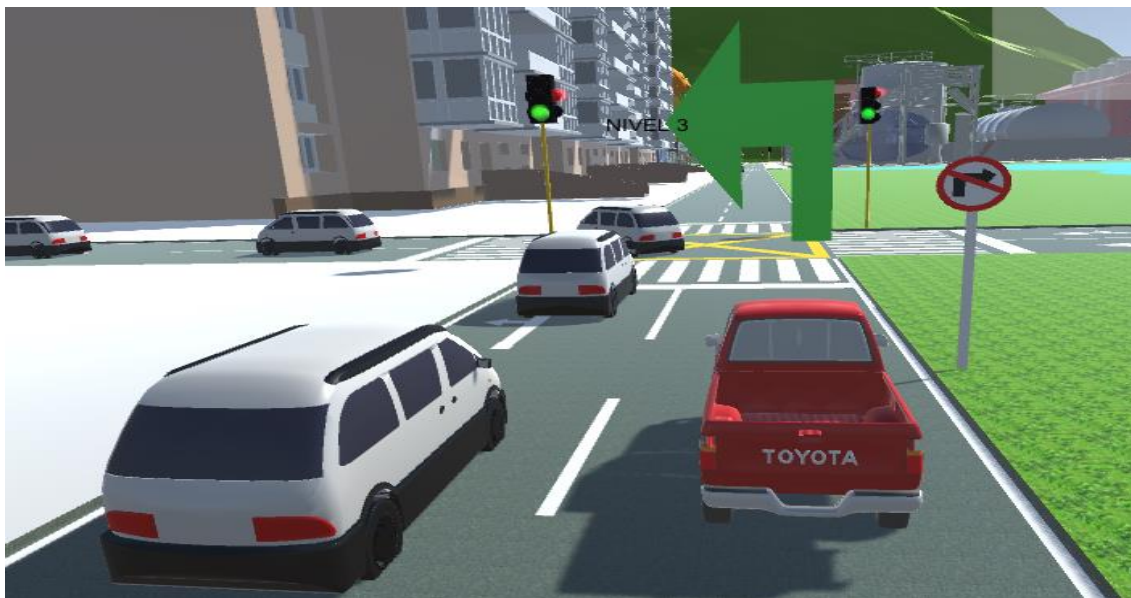


Figura 2. Situación 2: No respetar la fila para el giro.

En la *Figura 2* se representa el giro indebido del vehículo rojo ya que éste debe tomar el carril izquierdo para girar, en este caso el auto debe tomar una ruta alternativa para cumplir el desafío del nivel.



Figura 3. Situación 3: No respetar la demarcación de bloqueo de cruce.

La *Figura 3* da evidencia de una infracción demasiado común y que genera atascos en las vías, se trata de la congestión que se genera en la demarcación de cruce, la cual está pintada en amarillo. Sobre este territorio la ley de tránsito estipula que está prohibido estacionarse.

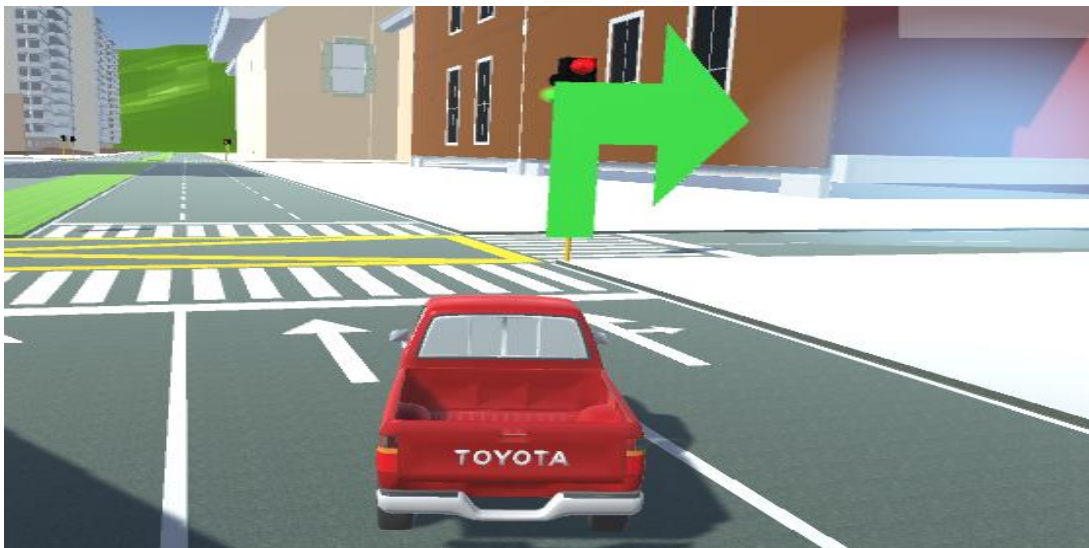


Figura 4. Situación 4: No respetar la demarcación de línea continua.

La demarcación con línea continua señala que no se puede realizar el cruce de carril, infracción que comete el carro rojo al momento de intentar girar a la derecha (*Figura 4*).

Capítulo 2.

2 MARCO TEÓRICO.

2.1 ANTECEDENTES.

Se han evidenciado diversas contribuciones con proyectos que hacen referencia al tema vial, de animaciones y realidad virtual. Con respecto a aplicaciones educativas que utilicen realidad virtual como herramienta de enseñanza y más específicamente en el tema de seguridad vial el material es escaso, sin embargo se referencian algunos trabajos que aportaron a la consolidación de este proyecto.

A continuación se describen algunos trabajos realizados y que contribuyeron a la consolidación y realización de éste proyecto, se muestran algunas imágenes que permiten construir ideas sobre el trabajo y se detalla los aspectos relevantes que favorecieron la consecución de la aplicación.

2.1.1 Simulaciones de tránsito.

Las simulaciones de tránsito son relevantes cuando de realidad virtual se trata, la mayoría de herramientas que estudian comportamientos y conductas de las personas al volante se basan en la simulación, incluso se han realizado proyectos enfocados a las poblaciones con deterioro cognitivo. Tal es el caso del proyecto VR- DAS (virtual reality -based driving -assessment system) [3] el cual permite el desarrollo de escenarios relevantes de conducción que pueden proporcionar medidas objetivas y cuantificables de comportamientos al conducir. El VR-DAS posibilita a los conductores la experiencia real de conducción, lo que facilita el estudio del comportamiento y la capacidad de respuesta que puede ser más predictiva en relación a la capacidad de conducción actual. Otros proyectos similares que se centran en el modelado de entornos para simuladores de conducción [4] [5] que permitan la integración con otros componentes de simulación para su ejecución en tiempo real. Estos últimos proyectos le apuntan a

realizar experimentos e investigaciones relacionadas a la simulación de vehículos y ambientes de conducción a diferencia del primero que se fundamenta en el estudio del comportamiento de los conductores, sin embargo poseen algo en común y es que lo hacen a partir de las simulaciones.



Figura 5. Captura de imagen de una simulación de tránsito.

(Fuente:

[https://www.google.com.co/search?q=simulacion+de+transito+vehicular&espv=2&biw=1366&bih=667&source=Inms&tbm=isch&sa=X&ved=0ahUKEwibgLP6g7XKAhVESyYKH32BFYQ_AUIBygB#imgrc=_\)](https://www.google.com.co/search?q=simulacion+de+transito+vehicular&espv=2&biw=1366&bih=667&source=Inms&tbm=isch&sa=X&ved=0ahUKEwibgLP6g7XKAhVESyYKH32BFYQ_AUIBygB#imgrc=_))

La figura 5 muestra una imagen tomada de un video donde se simula el tránsito de una ciudad, aquí se representa un cruce principal donde los automóviles circulan respetando la semaforización y la demarcación de la cebrá.

La implementación de la simulación dentro de éste proyecto contribuye a contextualizarlo desde dos aspectos, por un lado nos ayuda a pensar en cómo diseñar los gráficos, modelos y actividades particulares para lograr el mayor realismo posible recreando lugares específicos donde se evidencie el caso concreto en la vida real tal como propone el segundo trabajo. Por otro lado y a partir del primer antecedente citado, podemos desarrollar la aplicación basándonos en el comportamiento que pueden tener los conductores cuando enfrentan diversas situaciones que hacen parte de las dinámicas propias de conducir.

Teniendo en cuenta los proyectos citados anteriormente y ahora centrándonos más en la parte del desarrollo de la aplicación, el mayor aporte de éstos trabajos al

proyecto radica en las técnicas de autonomía, dinamismo y control que deben aplicarse en las etapas de consecución de la aplicación para generar la inteligencia del escenario y cada uno de los elementos que lo componen.

2.1.2 Realidad virtual y conducción.

Para la realización del proyecto “tutráfico” se quiso hacer uso de herramientas tecnológicas innovadoras que permitieran una experiencia nueva y diferente para los usuarios, es aquí donde la realidad virtual inmersiva cobra un papel relevante en el desarrollo e implementación de la aplicación. Una gran variedad de trabajos relacionados a éste tema podrían ser citados, sin embargo, se nombran aquellos que tiene más relevancia y fueron de gran aporte para la culminación del proyecto

El proyecto SS-ARX (stochastic switched autoregressive exogenous) [6] presenta el desarrollo de un modelo y reconocimiento del comportamiento de conducción basado en un modelo estocástico exógeno conmutado autorregresivo. Éste trabajo centra su atención en el desarrollo de un algoritmo basado en el modelo ARX para estimar parámetros relacionados con la conducta que desarrolla un conductor al momento de evitar una colisión. Por otro lado, un trabajo denominado Development of Immersive Virtual Driving Environment Using OpenCABIN Library [7] permite investigar como los conductores de la tercera edad pueden conducir de manera más segura a partir del desarrollo de un ambiente virtual de conducción. En el mismo sentido, la propuesta Immersive Driving Car Simulation for Children Using Natural User Interface Controller [8] permite el desarrollo de un mini auto físico real para la simulación inmersiva de conducción, el cual está enfocado en proveer al usuario de un entorno de simulación en donde éste podrá controlar el auto utilizando gestos con la mano que serán capturados por el Kinect. Mientras el usuario va conduciendo el vehículo encontrará varios marcadores AR (Argumented Reality) que va a visualizar diversas situaciones de tráfico lo que permitirá la creación de un contenido de entretenimiento educativo sobre la norma de tránsito. Otra opción no muy alejada de la finalidad del proyecto se muestra en la figura 6, éste trabajo presenta un software para el reconocimiento de la mano y

sus movimientos lo cual permite al usuario manipular objetos virtuales en tiempo real.



Figura 6. Captura de imagen de herramienta de comprensión espacial basada en sistemas de visión artificial.

(Fuente: Trabajo de grado. Herramienta para el desarrollo de la comprensión espacial basada en sistemas de visión artificial).

En base a los trabajos citados, se puede profundizar en el trabajo con sistemas inmersivos, en donde el usuario interactúa en forma directa como un personaje de la aplicación. Estos trabajos contribuyen a pensar la forma de recrear el ambiente donde se llevan a cabo las actividades propuestas así como las técnicas de interacción entre el usuario y la aplicación con el fin de implementar nuevas herramientas tecnológicas que permitan evidenciar situaciones cotidianas de manera atractiva para el usuario sin perder el objetivo fundamental del proyecto.

2.1.3 Videojuegos sobre tránsito y seguridad vial.

Otra de las perspectivas que posibilitó la consecución de ésta herramienta hace referencia al desarrollo de videojuegos relacionados a la temática de conducción y seguridad vial. Dentro de una variada gama de opciones se han destacado las más relevantes y que se aproximan en gran manera al propósito del proyecto.

En primera instancia encontramos una propuesta denominada Driver Behaviour Analysis through Simulation [9] la cual presenta el diseño y desarrollo de un simulador de conducción usando un motor de videojuegos 3D. El simulador fue empleado para analizar los comportamientos de conducción en carretera de los usuarios ubicados en un punto determinado de Nicosia, en Chipre.

Por la misma línea se encuentra un videojuego denominado SITP súbete⁴ que se presenta en la figura 7, éste recrea situaciones a la hora de utilizar el sistema integrado de transporte público en Bogotá. Aquí, el jugador interactúa manejando los diferentes buses que conforman el sistema y teniendo que cumplir una serie de retos relacionados con el mismo, además, se brinda información referente al sistema. El proyecto brinda una serie de información que emula el comportamiento en la vida real del sistema integrado de transporte público en Bogotá de una forma que no sean invasivas dentro del videojuego y permita una comunicación fluida con el jugador.



Figura 7. Pantallazo del videojuego sobre el SITP y sus mecánicas.

(Fuente: https://play.google.com/store/apps/details?id=com.twelvehitcombo.sitp&hl=es_419)

⁴ <http://www.transmilenio.gov.co/es/articulos/subete-al-juego-del-sitp>

Estos trabajos permiten ahondar en aspectos de diseño y modelamiento del escenario y elementos que se quieren recrear en la aplicación. El aporte de estos videojuegos se centra principalmente en el arte y la técnica de animación empleada, así como el aspecto visual de la misma. El videojuego del SITP súbete contribuye de manera significativa a la temática de educación en seguridad vial puesto que su desarrollo trabaja en torno a las dinámicas que se presentan dentro del sistema apoyándose en los acontecimientos que suceden en el contexto real.



Figura 8. Imagen del videojuego Tráfico Santiago de Cali.

(Fuente:

<https://www.facebook.com/traficosantiagodecali/photos/a.123638067829407.1073741827.123632777829936/123638071162740/?type=3&theater>)

Otro aporte de los videojuegos a la aplicación se muestra en la figura 8, Tráfico Santiago de Cali⁵ es un videojuego persuasivo que recrea experiencias inmersivas con el fin de recrear situaciones de la vida real que propicien el aprendizaje y la generación de conciencia y que a su vez disminuya la tasa de accidentalidad. Este proyecto de videojuego permite al jugador transitar y explorar las vías la ciudad de Santiago de Cali, así como su arquitectura y algunos de sus sitios más representativos. El juego recrea dinámicas dentro de los escenarios que incluyen tráfico vehicular real de la ciudad teniendo en cuenta aspectos como el flujo de vehículos, condiciones climatológicas, las diferentes horas del día, la forma de conducir de los habitantes, entre otras dinámicas propias del entorno de la ciudad.

⁵ https://www.icesi.edu.co/revistas/index.php/sistemas_teleomatica/article/viewFile/1259/1693

Las principales contribuciones de este trabajo, por un lado, se enfocan en el uso de herramientas interactivas para el aprendizaje, y por otro lado, retoma la temática de educación en seguridad vial permitiendo enfrentar al usuario en un contexto simulado de conducción, a diversas situaciones que suceden en su entorno estratégicamente diseñado en forma de videojuego. Conjuntamente, este trabajo contribuye significativamente con información relacionada a la metodología y proceso de desarrollo de la aplicación así como de las mecánicas e interacción de la misma.

2.2 MARCO CONCEPTUAL.

En esta sección se referencia la información necesaria y pertinente para el diseño y construcción de la propuesta “aplicación basada en realidad virtual inmersiva que permita evidenciar infracciones cometidas al momento de conducir”.

2.2.1 Realidad virtual (RV).

2.2.1.1 Definición de realidad virtual.

Dicho en palabras globales, la realidad virtual por lo general es un mundo virtual que se ha creado a partir computadores (sistemas informáticos) en donde el usuario tiene la sensación de estar dentro de éste, en el cual, dependiendo del nivel de inmersión, puede interactuar con dicho mundo y los objetos que lo componen. Visto desde otra perspectiva, la realidad virtual puede atribuirse a otros mundos virtuales generados por otros medios, un ejemplo claro puede ser a través de la imaginación (libros, sueños, cine, etc...) [10].

En relación a la simulación hecha en realidad virtual se realiza análogamente a través de escenas virtuales, en donde un mundo virtual es recreado y solo existe en los ordenadores o dispositivos destinados para tal fin. Dicho mundo virtual también recrea los lugares u objetos que hagan parte de él y que existen en la realidad. De la misma manera, la realidad virtual permite la captura de los movimientos naturales del usuario que luego serán proyectados en el mundo

virtual que se ha generado con antelación, proyectando en este los movimientos reales. El modelo de realidad virtual ideal radicaría en el hecho de permitir al usuario una inmersión total que logre interacción ilimitada con el mundo virtual, permitiendo al usuario, como mínimo, recrear los sentidos que se tienen en el mundo real (vista, oído, olfato, gusto y tacto). Sin embargo, los sistemas actuales que trabajan en realidad virtual solo se enfocan en 2 sentidos (vista y oído), puesto que los otros demandan más costos y una mayor dificultad para emularlos. La figura 9 muestra un esquema que describe del concepto de realidad virtual.

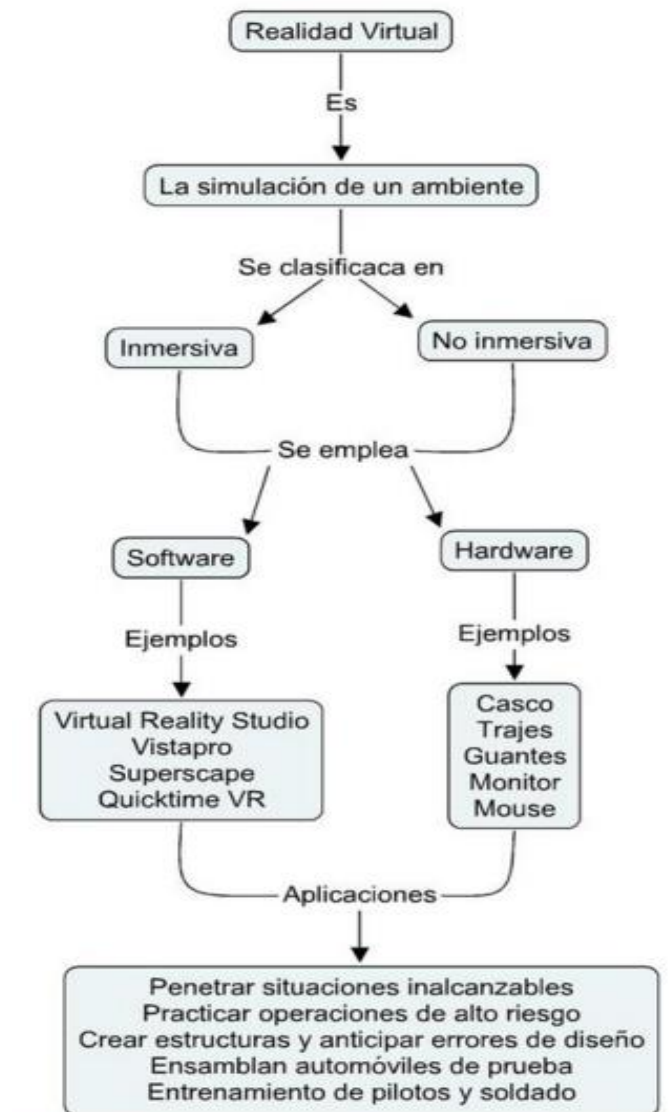


Figura 9. Mapa conceptual sobre realidad virtual.

(Fuente: <http://realidadvirtualmultimedia.blogspot.com.co/2015/11/realidad-virtual.html>)

2.2.1.2 Tipos de realidad virtual.

En relación a los tipos de realidad virtual, esta puede subdividirse de diversos modos, por un lado, de acuerdo al tipo de inmersión, a saber:

Realidad virtual inmersiva: Se caracteriza por lograr una inmersión total en el mundo virtual a través de dispositivos (gafas, HDM (Hypertext Desing Model), casco de realidad virtual, posicionadores, etc.). Este tipo de inmersión se caracteriza por una simulación que desaparece el mundo real en su totalidad.

Realidad virtual semiinmersiva: En este tipo de realidad virtual, el usuario interactúa pero sin estar sumergido en el mundo virtual, por ejemplo a través de dispositivos tecnológicos tales como un monitor. Este tipo de realidad virtual es muy común en los videojuegos ya que no requiere dispositivos de hardware especial para su visualización.

Existen otros tipos de realidad virtual que determinan si esta es individual o compartida, estas son:

Humano – máquina: Este tipo de realidad virtual tiene como limitante un usuario por mundo virtual. Un claro ejemplo de este tipo son los cines 3D, los videojuegos no multijugador, entre otros.

Humanos – máquina: Contrario al anterior, en este tipo de realidad virtual más de una persona puede interactuar con el mundo virtual y/o entre ellos.

2.2.1.3 Aplicaciones de realidad virtual. [11]

Los usos y aplicaciones más comunes de la realidad virtual son:

- Entretenimiento (videojuegos, cine 3D y 4D, aplicaciones, etc.) y Diseño asistido por computador (CAD).
- Aplicaciones educativas para la enseñanza de temas específicos de un área determinada.

- Creación de ambientes virtuales (tiendas, zoológicos, museos, bibliotecas, etc.).
- Medicina educativa, por ejemplo para simular operaciones y simulación del cuerpo humano.
- Entrenamiento de soldados, marineros, astronautas, etc.
- Tratamiento de enfermedades y/o fobias (entomofobia, acrofobia, claustrofobia, ansiedad, etc.).
- Simulación de comportamientos de las personas, por ejemplo para resolución de conflictos o estudio y tratamiento de la violencia o la adicción.
- Simulaciones de vuelos, navegación, conducción, etc.

2.2.1.4 Dispositivos tecnológicos utilizados en realidad virtual.

En principio se podría pensar que la parte de hardware o dispositivos que se usan en realidad virtual o realidad aumentada podría ser los mismos, sin embargo, de acuerdo a las aplicaciones para las que se requiera, cada una de ellas utiliza diversas herramientas. A continuación se exponen los instrumentos o dispositivos que se requieren para su funcionamiento y/o aplicación. Es preciso aclarar que detrás de cada uno de los dispositivos que se utilizan para experimentar tanto con realidad virtual como realidad aumentada, se encuentran empresas trabajando en ellos para lograr la mejor experiencia al usuario que desee adentrarse en este mundo. Varios de los dispositivos aún se encuentran en desarrollo, otros ya se encuentran disponibles al público. [12] [13]

1. Cascos o gafas de realidad virtual: Por lo general son los utilizados para la recreación y visualización dentro del mundo virtual. Dentro de los más reconocidos y utilizados del mercado podemos encontrar:

- Oculus Rift: Gafas de realidad virtual fabricadas por la empresa Oculus VR⁶. Este dispositivo se encuentra en fase de desarrollo sin embargo se puede adquirir una versión experimental. Estas se conectan al ordenador a través de un cable y posteriormente se da ejecución al software que permite la experiencia en el entorno virtual.
- Samsung Gear VR: Herramienta de realidad virtual fabricadas por la empresa Samsung en colaboración con Oculus VR. Se diferencia de las anteriores en la ausencia de pantalla debido a que estas son una carcasa que contiene una serie de botones y un sensor de movimiento, cuya utilidad es soportar un dispositivo móvil (generalmente de la misma marca) con el fin de realizar funciones de pantalla y procesador de información.
- PlayStation VR: Dispositivo de realidad virtual en forma de casco concebido por la empresa Sony. Su fabricación ha sido pensada para funcionar plenamente con la consola de videojuegos PlayStation 4.
- Google Cardboard: Este dispositivo en forma de gafas ha sido creado por Google. Generalmente pensado como una carcasa hecha de cartón que permite experimentar la realidad virtual a nivel casero permitiendo ubicar un Smartphone (teléfono inteligente) sobre ella.
- Otros dispositivos de visualización como las gafas de realidad aumentada Microsoft HoloLens⁷ (presentadas al público en 2015) o el proyecto HTC Vive⁸ (proyecto en desarrollo) hacen parte de las herramientas proyectadas para realidad virtual. Otro mecanismo como el Virtual Boy (consola de videojuegos lanzada por Nintendo en 1995) que incluía unas gafas que

⁶ Oculus VR: Compañía que desarrolla tecnología de realidad virtual. Adquirida por Facebook en el año 2014.

⁷ Microsoft HoloLens: Gafas inteligentes de realidad aumentada.

⁸ Htc VIVE: <http://htcvr.com/>

emulaban entornos tridimensionales fracasó comercialmente por lo que desapareció del mercado poco tiempo después de ser lanzada.

2. **Guantes:** Estos dispositivos son utilizados dentro del mundo virtual para recrear el sentido del tacto. Dentro de los más funcionales y conocidos se encuentran:

- **Gloveone:** Fabricado por la empresa NeuroGigital Technologies. Es un dispositivo en forma de guante cuya finalidad es recibir sensaciones que permitan estimular el sentido del tacto mediante una serie de sensores dispuestos adecuadamente y cuya finalidad es proporcionar al usuario una sensación de existencia de un objeto en su mano.
- **PowerClaw:** Esta es una especie de interfaz cuya finalidad radica en la estimulación de la piel para así lograr el desarrollo en el sentido del tacto. Este dispositivo permite generar diversas sensaciones de diversas características de los objetos como temperatura, movimiento y textura. Está pensado para interactuar junto con otros dispositivos y también con herramientas de software para aumentar y mejorar el nivel de experiencia dentro del mundo virtual.

3. **Periféricos:** Usualmente los periféricos se utilizan, por un lado, para ingresar o sacar información al sistema, y por otro, como dispositivo de procesamiento de información. Algunos periféricos conocidos son:

- **Dispositivos móviles:** Como se mencionaba anteriormente, a pesar de que algunos dispositivos de visualización para realidad virtual ya poseen pantalla propia que posibilita una mejor experiencia dentro del mundo virtual como por ejemplo las Oculus Rift, otros han sido fabricados como carcasas que soportan los periféricos que permiten la inmersión en la virtualidad. Los teléfonos móviles inteligentes (Smartphones) han posibilitado la mejora de

la experiencia dentro del mundo virtual debido a su tamaño, definición y utilidad. Estos son usados principalmente para visualización de aplicaciones, sin embargo, también han permitido nuevas experiencias en el tema de los videojuegos.

- Teclado y mouse de computador: Estos dispositivos de entrada son utilizados como controles de juegos y/o aplicaciones para computador. Programando en un software especializado los botones del teclado, los botones del mouse o el apuntador del mouse estos pueden adoptar los movimientos que van a ejecutar los personajes u objetos que hacen parte de la aplicación o videojuego.
- Joystick (Palanca de mando) o Gamepad (control de mando): Son dispositivos que permiten el control de personajes u objetos dentro de videojuegos y/o aplicaciones.
- Otros periféricos: Para que la experiencia dentro del mundo virtual sea mucho más realista, compañías de varios países han desarrollado otras herramientas que lo posibilitan, un ejemplo de ello es el Virtuix Omni⁹, accesorio complementario para las Oculus Rift que consiste en una plataforma en donde el usuario puede emular el movimiento que realiza al caminar el cual será reflejado, casi que idénticamente, dentro del mundo virtual. Una versión similar de este periférico es de origen austriaco la cual se ha denominado Cyberith Virtualizer¹⁰.
- Otros sistemas: aproximadamente desde los años 1900 se ha implementado un tecnología denominada sistemas Cave (Cave automatic virtual environment) la cual permite recrear un ambiente de realidad virtual en un cuarto con forma de cubo, en el cual se proyectan diferentes

⁹ Virtuix Omni: <http://www.virtuix.com/>

¹⁰ Cyberith Virtualizer: <http://cyberith.com/product/>

imágenes en sus paredes y en donde el usuario podrá verlas a través de unas gafas 3D además para hacerlo más realista se colocan altavoces en distintos puntos.

2.2.1.5 Software utilizado en realidad virtual.

Tanto para la realidad virtual como para la realidad aumentada, se hace indispensable contar con un ordenador y un tipo de software especializado, los cuales en conjunto, permitirán recrear dichos mundos. Hablar de esta interacción entre ordenadores y las aplicaciones se conoce comúnmente como Reality Engine¹¹ traducido al español como motor de la realidad.

Tanto la parte de hardware como de software son imprescindibles para la creación de un sistema de realidad virtual óptimo que centra su atención en crear los equipos más potentes y con características de procesamiento amplias, esto debido a las grandes cantidades de información que capturan, codifican y se transmiten a través de los periféricos de entrada lo que finalmente permite de ordenador y software den respuesta adecuadas a los eventos que el usuario dentro del mundo virtual o aumentado.

Una de las principales ventajas que posee la virtualidad, es que su información es actualizada y almacenada continuamente en la memoria del ordenador lo que hace posible ofrecer una simulación más detallada y precisa del entorno real. Debido a esto, la realidad virtual o la realidad aumentada propende de un dispositivo que permita el procesamiento, administración y concepción de mundos virtuales que respondan a los distintos requerimientos del usuario para lo que se dispone de un núcleo del sistema de realidad virtual que se ha denominado la máquina de realidad. Posibilita

En principio podría pensarse que la realidad virtual y la realidad aumentada son meramente dependientes del hardware, sin embargo, el software juega un papel igual de importante pues permite generar y transmitir las instrucciones lógicas al ordenador lo que podría limitar la eficacia y potencia del sistema si no se cuenta

¹¹ <https://en.wikipedia.org/wiki/RealityEngine>

con un equipo de cómputo adecuado, es allí donde la parte de hardware y su respectivo software se complementan.

2.2.2 Motores de videojuegos.

Cuando se habla del término motor de videojuegos podría hacerse la idea de un sistema que posibilite la consecución total de un videojuego y no se está muy lejos de la realidad. Un motor de videojuegos comprende una serie de instrucciones de programación que permitan una secuencia de diseño, creación y representación de un juego de video. La principal función de un motor es proporcionar al videojuego diferentes características que permitan la construcción del mismo, entre las que se encuentran un motor de renderizado para gráficos en 2 o 3 dimensiones, un motor para la detección de colisiones físicas de objetos y las respuesta a este, administración de memoria y un escenario gráfico¹².

En la actualidad existe gran variedad de motores de videojuegos que utilizan diversos lenguajes de programación. Cada desarrollador elige el motor más adecuado para su conveniencia y requerimiento, de la misma manera, existen versiones gratuitas al público interesado en el desarrollo de videojuego o aplicaciones, sin embargo, para poder acceder a herramientas más especializadas es necesario adquirir una licencia con valor comercial.

2.2.2.1 Motor de videojuegos Unity 3D.

Uno de los motores de videojuegos más versátiles, manejables y exequibles al público es Unity 3D. Esta plataforma de desarrollo de videojuegos y aplicaciones interactivas multiplataforma en 2D y 3D fue creada en el año 2005 por la compañía estadounidense Unity Technologies¹³. Unity está disponible como plataforma de desarrollo para Windows, OS X y Linux. Esta herramienta permite la creación de videojuegos y aplicaciones para las consolas Xbox 360 y One, Play Station 3, 4 y Vita, Nintendo Wii y Wii U, iPad, iPhone, Android, Windows Phone, entre otras.

¹² <http://blogthinkbig.com/motores-graficos/>

¹³ <http://www.moddb.com/engines/unity>

Gracias a este soporte multiplataforma y al plugin web de Unity, es posible la creación de videojuegos de navegador (que puede ejecutarse mediante un navegador web) tanto para Windows como para Mac. A continuación se nombran las variedades de plataformas que soporta Unity 3D.

- ✓ Soporte para creación de juegos para dispositivos móviles. Actualmente disponible como plataforma para desarrollo para Android, iOS, Windows Phone y Tizen.¹⁴
- ✓ Unity 3D permite crear juegos para Windows, OS X, Linux, Xbox 360 y One, Play Station 3, 4 y Vita, Wii, Wii U.
- ✓ Soporte para la creación de juegos de escritorio compatibles con Windows, Mac y Linux.
- ✓ Plataforma para desarrollo de realidad virtual y realidad aumentada. En la actualidad existe soporte para Oculus Rift, Gear VR y Play Station VR pero se añaden más plataformas constantemente.
- ✓ Con la implementación del plugin Unity Web Player, Unity 3D permite desarrollar juegos en 3D y llevarlos a la web.

2.2.3 Normas de tránsito.

Las normas de tránsito son un componente principal cuando de seguridad vial trata, estas posibilitan la regulación en la circulación sobre la vía pública previniendo accidentes de tránsito entre los agentes que intervienen en la calle. Cuando se habla de normas de tránsito indiscutiblemente debe hablarse de leyes, las cuales están contempladas y compiladas en un documento denominado código nacional de tránsito ley 769 de 2002¹⁵ las cuales van dirigidas no solo a los conductores de vehículos automotor carros o motos sino a todos los actores de la

¹⁴ <http://computerhoy.com/noticias/software/que-es-tizen-que-podemos-esperar-este-nuevo-9265>

¹⁵ <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=5557>

movilidad llámense conductores, pasajeros o peatones. Todas aquellas personas que decidan salir de su sitio de alojamiento y van a ingresar a la vía pública tienen que aplicar estas normas de comportamiento, de igual manera sucede para los conductores de carro o moto que voluntariamente tramitaron su licencia de conducción y gracias a esto asumieron el deber de cumplir el contenido y alcance de dicha norma.

Las normas de tránsito se rigen también a partir de la constitución política de Colombia la cual en su Artículo 24 divulga *“Todo colombiano, con las limitaciones que establezca la ley, tiene derecho a circular libremente por el territorio nacional, a entrar y salir de él, y a permanecer y residenciarse en Colombia.”*¹⁶ Y bajo unos principios rectores para éste código que son la seguridad de los usuarios, calidad, oportunidad, cubrimiento, libertad de acceso, plena identificación, libre circulación, educación y descentralización. Finalmente, esta norma ha sido implementada obligatoriamente en la enseñanza de la educación Preescolar, Básica Primaria, Básica Secundaria y Media Vocacional como se establece en la norma (Ley 769 de 2002) en el Artículo 56.

2.2.3.1 Educación en seguridad vial.

La seguridad vial se ha tomado el ámbito educativo hasta el punto de convertirse en una asignatura obligatoria dentro de aula de clase. Este proyecto liderado bajo la batuta de la ministra de educación María Fernanda Campos Saavedra y soportado en el documento denominado “Orientaciones pedagógicas para la movilidad segura: Un enfoque de educación vial ‘Saber Moverse’” pretende dar las orientaciones pedagógicas que en las instituciones educativas deben implementarse con el fin de desarrollar competencias para la movilidad vial, en donde afirma *“Queremos formar a nuestros niños, adolescentes y jóvenes en competencias para afrontar los riesgos de la movilidad vial, por eso, más allá de la enseñanza de medidas de seguridad o señalización, promoveremos la formación de personas con comportamientos, actitudes y hábitos idóneos para moverse por*

¹⁶ <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=4125>

las calles con seguridad, cooperando con los otros en el tránsito y en relación armónica con el entorno".¹⁷

Este documento se construye como apoyo a la ley 1503 de 2011 a través del cual se promueve la formación de hábitos, comportamientos y conductas en la vía¹⁸ y está dirigido a todas las instituciones educativas, así como a la Secretaría de Educación. Su contenido está basado en la fundamentación de la seguridad vial, la orientación por competencias que permita una movilidad segura y la guía para el desarrollo del programa pedagógico transversal en el tema de seguridad vial. El objetivo primordial de estas orientaciones pedagógicas es la exploración de una inmejorable seguridad en la movilidad que permita reducir los índices de accidentalidad en la vía.

Gracias a esta necesidad de llevar la temática de seguridad vial al aula de clases y hacerla imprescindible para la enseñanza, se construye un módulo ligado al proyecto educativo institucional y la seguridad vial en el marco de proyectos transversales instaurados por la secretaria de educación.

2.2.3.2 El Proyecto Educativo Institucional (P.E.I) y la seguridad vial.

En la actualidad, aproximadamente a finales del siglo XX y comienzos de XXI, la prevención empieza a tomar un papel importante como herramienta para cuidar y preservar la vida de los seres humanos. En base a esto, surgen diversas ideas y propuestas alrededor de la educación no formal, formal e informal, las cuales centran su atención en la idea de cultura ciudadana, que en principio, es pensada como el conocimiento práctico que debe vivir cada individuo para relacionarse con sus semejantes en diversos aspectos de su vida. [14][15]

En concordancia, es aquí donde la cultura ciudadana empieza a tomar relevancia; y es en las instituciones educativas donde comienza a forjarse; es por ello que se plantean una serie de proyectos trasversales entre los que se encuentra uno relacionado a la formación en seguridad vial, los cuales ya han empezado a

¹⁷ <http://www.mineducacion.gov.co/cvn/1665/w3-article-343571.html>

¹⁸ <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=45453>

hacerse obligatorios en el currículo escolar. Gracias a estos proyectos, se plantea una estrategia pedagógica para articular de manera sistemática dichos procesos a partir de las diferentes áreas del conocimiento y el proyecto educativo institucional (P.E.I.). Para fortalecer el proceso, la secretaria de educación ha dispuesto un módulo como una propuesta de organización curricular interdisciplinaria que gira en torno al eje transversal de Educación en Seguridad Vial.

El módulo denominado “El proyecto educativo institucional y la Seguridad Vial”¹⁹ se ha dividido en 4 capítulos en los cuales todos los agentes que participan en el proceso de formación (directivos, docentes y comunidad educativa) pueden retroalimentarse para reestructurar y proyectar el currículo escolar y el plan de estudios de la institución educativa alrededor del eje transversal de seguridad vial escolar. La construcción de este módulo tiene dos objetivos, por un lado, el de proponer una estrategia que permita la articulación de contenidos de las diferentes áreas del conocimiento alrededor del eje transversal de educación en seguridad vial, y por otro lado, desarrollar una serie de elementos, tanto metodológicos como teóricos, del proceso de planificación educativa en torno a dicho eje transversal que permita lograr una mejor comprensión de la estrategia de articulación de contenidos.

2.2.4 Metodologías para diseño de software.

Para el diseño y desarrollo de software se emplean diversas metodologías y modelos que permiten la resolución de problemas. La elección de estas metodologías depende de los requerimientos del proyecto, de esta manera, aquellas basadas en planificación y control además de hacer énfasis en especificación de requisitos y modelado se conocen como metodologías tradicionales. Por otro lado existen unas orientadas y centradas específicamente a la planificación durante el proceso de desarrollo de un determinado proyecto, estas se caracterizan por desarrollos en periodos de corto plazo, por lo que se han

19

http://www.educacionbogota.edu.co/archivos/Sitios_de_Interes/2013/Marzo/Proyecto%20educativo%20%20seguridad%20vial.pdf

denominado metodologías ágiles [15]. Aunque estas metodologías surgieron en el ámbito de la elaboración de software, prontamente fueron trascendiendo a otros contextos, entre ellos el desarrollo de aplicaciones basadas en dispositivos móviles. Estas metodologías ágiles surgen en oposición y como alternativa a los métodos tradicionales y provienen de una serie de principios que pueden encontrarse en el “Manifiesto Ágil” [16], los cuales se basan en el desarrollo iterativo (repetitivo) e incremental (creciente) [17], donde los requerimientos y soluciones evolucionan con tiempo de acuerdo a las necesidades del proyecto. La metodología está planteada en una serie de etapas de desarrollo que permiten pasar de una idea al análisis y finalizan con la creación y aprobación del proyecto.

Una de las metodologías ágiles más representativas se denomina SCRUM que adopta una estrategia de desarrollo incremental, caracterizada por el trabajo en equipos organizados estratégicamente los cuales trabajan en fases u etapas que sufren modificaciones a medida que avanza el desarrollo y ejecución del proyecto lo que posibilita mejoras significativas relacionadas a la reducción de tiempos en la creación y desarrollo sin afectar la calidad del software. [18] [19] [20] [21]

La tabla 2 muestra las ventajas y desventajas de metodologías tradicionales en comparación a la metodología SCRUM para el diseño y desarrollo de software.

Tabla 1. Comparación de modelos de desarrollo de software.

MODELO	VENTAJAS	DESVENTAJAS
CASCADA	<ul style="list-style-type: none"> -La planificación es sencilla. -La calidad del producto resultante es alta. -Permite trabajar con personal poco cualificado. 	<ul style="list-style-type: none"> -Modelo lineal. -Se necesita tener todos los requisitos al principio. -No hay posibilidad de corregir errores a tiempo. -Aumento en los costos de desarrollo.
ESPIRAL	<ul style="list-style-type: none"> -No necesita una definición completa de los requisitos para empezar a funcionar. -Es más fácil validar los requisitos. -El riesgo en general es menor. -El riesgo de generar retrasos es menor. 	<ul style="list-style-type: none"> -Es difícil evaluar los riesgos. -Necesita de la participación continua por parte del cliente. -Cuando se subcontrata hay que producir previamente una especificación completa de lo que se necesita y esto lleva tiempo.

<p>INCREMENTAL</p>	<ul style="list-style-type: none"> -Con un paradigma incremental se reduce el tiempo de desarrollo inicial. -También provee un impacto ventajoso frente al cliente, que es la entrega temprana de partes operativas del software. -Permite entregar a cliente un producto más rápido en comparación del modelo de cascada. 	<ul style="list-style-type: none"> -Requiere de mucha planeación, tanto administrativa como técnica. -Requiere de etapas claras para conocer el estado del proyecto.
<p>SCRUM</p>	<ul style="list-style-type: none"> -La creación iterativa logra evidenciar el progreso del producto en diferentes etapas de desarrollo. -La producción es adaptativa a los percances que puedan suceder, gracias a la implementación de controles diarios y de entrega de iteración. -Es ideal para proyectos pequeños ya que su fortaleza reside en la capacidad de trabajo de cada pequeño equipo de trabajo. 	<ul style="list-style-type: none"> -Exige un gran conocimiento técnico por parte de los desarrolladores, ya que cada equipo debe ser autosuficiente y autorregulado, lo que obliga a cada uno de estos posea competencias en todas las áreas de desarrollo. -No se fundamenta en la documentación creada en etapas iniciales como en otras metodologías de desarrollo. -No genera todas las evidencias de documentación de otras metodologías.

Capítulo 3.

3 DISEÑO DE LA APLICACIÓN

3.1 GENERALIDADES DE LA APLICACIÓN.

3.1.1 Investigación, análisis y propuesta de la aplicación.

El primer paso en la investigación realizada surgió de la necesidad de recrear un escenario en donde se puedan evidenciar claramente las infracciones que se plantean en la delimitación. Luego de evaluar diferentes posibilidades en cuanto al modelo de escenario se llega a la conclusión que éste debía ser creado en su totalidad pues no es posible determinar un lugar específico de la ciudad donde ocurran o se den todas las infracciones y que sirva como punto de referencia en el modelado.

Una vez modelado el escenario se procede a la elección del motor de videojuegos para la implementación, dicha elección se hizo en base a las siguientes características: a) Que el software no requiriera de soporte y plataforma muy avanzados con el fin de permitir el acceso a cualquier usuario tanto al software como a los archivos ejecutables. b) Se estudió la posibilidad de poder comercializar el producto final evitando los requerimientos de autor que demandan los softwares. c) Finalmente se tuvo muy en cuenta la interfaz de usuario del software debido a la trascendencia del trabajo con herramientas amigables, versátiles y flexibles. Se elige como motor de desarrollo de la aplicación a Unity 3D, pues además de cumplir las características mencionadas anteriormente, es una herramienta que se conoce y con la cual se ha trabajado gracias a la asignatura electiva de diseño de videojuegos.

Finalmente se desarrolló una aplicación en 3D empleando tecnologías basadas en “sprites bidimensionales”²⁰ para la plataforma PC con viabilidad de distribución en Android e IOS gracias a la posibilidad de juego en dispositivos con pantallas táctiles, cuyo objetivo es enfrentar una serie de retos relacionados a la forma de conducir los cuales están ligados a las normas de tránsito y seguridad vial. El usuario estará limitado al manejo de un vehículo en donde se conduce sobre las vías del escenario y cuyo fin es ir a un determinado punto sorteando en su trayecto diversos desafíos, que al ser superados, permitirán el acceso a un nuevo nivel de la aplicación. El éxito en cada uno de los niveles propenderá de la información oportuna que tenga el usuario acerca de la normatividad de tránsito y su formación en seguridad vial.

3.1.2 Historia

La aplicación permite reflejar la problemática seguridad vial que de un tiempo atrás hasta la actualidad ha venido en detrimento. Desde el desconocimiento de las señales de tránsito hasta el incumplimiento de las leyes estipuladas en el código nacional de tránsito del año 2002 pretenden ponerse en evidencia a través de esta herramienta. La aplicación contiene una serie de niveles relacionados directamente a la forma de conducción del usuario, el cual permitirá acceder al siguiente nivel una vez haya cumplido con el anterior.

En cada nivel el usuario podrá enfrentarse a una situación de movilidad diferente que requiere de su conocimiento y aplicación sobre las normas de tránsito en la cual podrá retroalimentar su desempeño a través de mensajes en la pantalla que informen cuando ha cometido una infracción, lo que podrá generar, en ocasiones, un reinicio automático del nivel.

El desarrollo de la aplicación surge de la observación y experiencia como individuos participes y directos de la problemática de educación en seguridad vial tanto de los autores como del director del proyecto. La finalidad es, además de

²⁰ sprites bidimensionales: Tecnologías que permiten dar impresión de espacios en dos y tres mensionés.
[https://msdn.microsoft.com/es-es/library/bb203919\(v=xnagamestudio.40\).aspx](https://msdn.microsoft.com/es-es/library/bb203919(v=xnagamestudio.40).aspx)

poner en evidencia la problemática, concienciar a los conductores y agentes que participan e intervienen en las dinámicas de tránsito, movilidad y seguridad vial, de las consecuencias que se pueden desencadenar gracias a los comportamientos inadecuados en la vía por parte de los conductores que desconocen o no acatan las normas y señales de tránsito existentes, entre las que se encuentran la formación de atascos y congestiones, la generación de accidentes de tránsito, la emisión de multas y sanciones entre otros.

3.1.3 Planificación.

A continuación se presentan algunas generalidades que permitieron planificar de forma ordenada y estratégica la consecución de la aplicación.

Descripción: Esta aplicación basada en realidad virtual inmersiva tiene como objetivo fundamental concienciar al usuario de ciertas infracciones que se cometen al conducir y todas las implicaciones que conllevan para el resto de agentes que intervienen. La aplicación permite revivir una experiencia libre y realista de conducción en un ambiente que recrea un sector de una ciudad. El usuario tiene la posibilidad de explorar las vías, contemplar el paisaje y la arquitectura creada. De la misma manera, ésta acepta reproducir escenarios que incluyen diversas características como flujo vehicular y forma de conducir de los automóviles. El usuario estará destinado a afrontar ciertos retos de conducción que han sido diseñados y programados como dinámica propia de la aplicación.

Plataforma: La aplicación se ha venido desarrollando para una implementación en plataforma PC con el fin de garantizar el mayor acceso de usuarios a ella, sin embargo, gracias al motor de videojuegos que se ha utilizado es posible aplicarlo a cualquier tipo de plataforma.

Público objetivo: Esta aplicación no está dirigida a una muestra en particular, sin embargo está pensada para una población con mayoría de edad y que cumpla

los estándares estipulados en el Código Nacional de Tránsito Terrestre consignado en la LEY 769 de 2002²¹.

Características de la aplicación: La Tabla 1 presenta las principales características de la aplicación.

Tabla 2. Características de la aplicación.

Característica	Descripción
Experiencia libre y real de conducción.	Permite revivir una experiencia libre y realista de conducción en un escenario virtual que recrea un contexto real.
Exploración.	El usuario tiene la posibilidad de explorar las vías, contemplar el paisaje y la arquitectura creada.
Tráfico.	La aplicación reproduce escenarios que incluyen diversas características como flujo vehicular y otras dinámicas.
Comportamiento vial.	La inteligencia artificial de los vehículos y de los elementos que componen la ciudad simula la dinámica propia de una ciudad real.
Actividades.	La dinámica de la aplicación posibilita que el usuario enfrente diversos retos de conducción que han sido programados dentro de ésta.

Aprendizaje: Dentro de la aplicación los usuarios se ven enfrentados a diversas situaciones de movilidad que emulan realidades presentes a diario en las vías; éstas exigen un manejo de los automotores bajo circunstancias difíciles de conducción las cuales ponen en evidencia su comportamiento, tolerancia y paciencia como conductores.

Mientras se interactúa con la aplicación, el sistema muestra avisos o mensajes acumulativos en los cuales el usuario podrá evidenciar que está cometiendo una o varias infracciones durante la ejecución de la actividad. Una vez iniciada la actividad el usuario estará enfrentado a un reto que consiste en ir de un punto a otro sin cometer infracciones, si no es el caso y se ha cometido una infracción, la actividad se reiniciará automáticamente después de unos segundos luego de

²¹ <http://www.alcaldiabogota.gov.co/sisjur/normas/Norma1.jsp?i=5557>

haber mostrado, a través de un aviso, que ha cometido determinada infracción, lo que permite que haya una retroalimentación dentro de ella, esto con el fin de generar un punto de reflexión donde los usuarios pueden determinar si su comportamiento en la vial es el más adecuado.

Niveles de juego: La aplicación consta de 4 niveles de jugabilidad, los cuales representan cada una de las actividades o retos a los que el usuario debe enfrentarse. Una vez se haya superado un nivel, automáticamente el usuario aparecerá en el siguiente reto el cual debe superar para continuar, de lo contrario, si no cumple con la actividad se mantendrá en el nivel o si comente alguna infracción la aplicación reiniciara automáticamente el nivel. Cada uno de los niveles de juego fue recreado en el mismo mapa pero en diferente ubicación y se generaron totalmente en Unity 3D.

3.1.4 Etapas de diseño.

Gracias a la adaptabilidad que posee la metodología SCRUM a diferentes proyectos en relación a sus requerimientos se puntualizaron una serie de etapas de diseño que contribuyen al desarrollo de la aplicación. A continuación se describen dichas etapas, a saber:

- 1. Planeación y requerimientos:** Durante esta etapa se determinan los requerimientos para la aplicación en relación al arte (incluyendo modelado, animación, montaje y diseño de interfaces), la programación y la visualización. Así mismo se requiere la planeación de cada una de las actividades que se llevaran a cabo en cada uno de estos requisitos.
- 2. Diseño:** Luego de recoger y analizar los requerimientos generados en la etapa anterior, el equipo de trabajo presenta las propuestas que permitan satisfacer éstos requerimientos y se establecen los roles de cada uno de los miembros del equipo conforme la evaluación de las propuestas presentadas.

3. **Desarrollo de la aplicación:** En esta etapa se da inicio al desarrollo de la aplicación acorde a los requerimientos planteados en la etapa 1 de manera separada para cada uno de los ítems y en los distintos software que requiere cada uno, posteriormente se evalúan las diferentes fallas, errores y posibles mejoras que se pueden realizar a la aplicación.
4. **Montaje:** Aquí se integra y consolida el trabajo elaborado en el motor de videojuegos Unity 3D, posteriormente se da inicio a las pruebas de funcionalidad en torno a las mecánicas de interacción con los diferentes elementos de la aplicación, se realiza una nueva corrección de errores y se construyen los requisitos finales del proyecto. En esta etapa también se realiza la programación de cada uno de los elementos que permiten las dinámicas de interacción dentro de la aplicación.
5. **Cierre:** Aquí se llevan a cabo las pruebas finales del funcionamiento interno de la aplicación en busca de posibles mejoras y modificaciones a requerimientos con fallas, esto bajo condiciones de aprobación tanto del experto como del equipo de trabajo y relacionada a los objetivos trazados según el producto final.

3.1.5 Fases de desarrollo.

La aplicación ha sido pensada y desarrollada a partir de 5 fases principales que son:

- **Modelado y texturizado:** En esta primera etapa se abarca la construcción de los modelos 3D relacionados con la aplicación; vehículos, escenario y demás objetos. Para su desarrollo se utilizó Google SketchUp y Unity 3D.

- **Animación:** La siguiente fase de desarrollo hace referencia a un proceso denominado Scripting de Animación o Legacy²² el cual consiste en configurar los modelos 3D, en este caso los vehículos, para que adopten la cualidad de animados. Este proceso se trabajó en Unity 3D.
- **Montaje:** Esta fase abarca la importación, arquitectura, ubicación y adecuación de los modelos elaborados en las etapas preliminares. En esta etapa se da inicio a las pruebas de funcionalidad con el motor de videojuegos que se haya elegido para la implementación. Para este caso, se utilizó *Unity 3D v5.1.2f1 personal* como Framework de desarrollo, gracias a su versatilidad, herramientas de creación de contenido y visualización avanzada.
- **Programación:** En esta fase, la más significativa en el desarrollo, se realiza la programación de las funciones de cada uno de los elementos elaborados en las fases anteriores. La ejecución de esta etapa se realiza en conjunto y simultáneamente con las fases de montaje y diseño de interfaces.
- **Diseño de interfaces:** Es la fase final del desarrollo y está compuesta por el diseño de menús, alertas y mensajes en pantalla. Junto con la programación, es una de las etapas más trascendentales y por ende es de vital importancia su planeación y ejecución exitosa, ya que por medio de ésta se busca transmitir gran parte del mensaje de concientización al usuario.
- **Proceso de visualización:** Esta fase consiste en la sincronización de los dispositivos que permiten navegar en el ambiente de realidad virtual

²² <http://docs.unity3d.com/es/current/Manual/AnimationScripting.html>

inmersiva con el fin de realizar las pruebas finales a la aplicación y corregir los inconvenientes que persisten en en desarrollo.

Como se mencionaba anteriormente, la aplicación se desarrolló en base a los requerimientos que iban surgiendo en cada una de las etapas de diseño planteadas. En la tabla 4 se referencian los requerimientos necesarios tanto de arte como de programación y visualización para el desarrollo de la aplicación. Estos requerimientos se amplían en los capítulos 4, 5 y 6.

Tabla 3. *Requisitos para desarrollo de la aplicación.*

REQUERIMIENTOS PARA LA APLICACIÓN	
ARTE	Diseño de escenario.
	Diseño de vehículos.
	Diseño de interfaces.
	Diseño de niveles.
	Diseño de actividades (retos).
	Diseño de visualización.
PROGRAMACIÓN	Control general de la aplicación.
	Control de vehículos.
	Control de semáforos.
	Control de notificaciones
	Control de interfaces.
VISUALIZACIÓN	Selección de plataforma.
	Herramientas de visualización.
	Sincronización de los dispositivos.

3.1.5.1 *Diseño de escenarios.*

Este requerimiento del arte hace referencia al esquema o bosquejo de la ciudad que se va a implementó para el desarrollo de las actividades que se han propuesto. La ciudad se diseñó de tal manera que fuera un escenario en donde el usuario estuviera propenso a cometer las infracciones de tránsito y de esta manera pueda evidenciar su desempeño como conductor. El diseño dela ciudad emula el comportamiento real de una ciudad y las dinámicas que se presentan en ella. Para darle más realismo, se ambienta con sonidos que permite al oído

reconocer el contexto en el cual se desarrolla la aplicación. Otros detalles visuales como las casas, edificios, arboles, montañas, empresas, lagos entre otros permiten al escenario cobrar vida propia.

3.1.5.2 *Diseño de vehículos.*

Los vehículos en principio fueron diseñados a partir de figuras geométricas básicas que no requirieran muchos recursos de máquina y sin centrarse en el aspecto visual de los mismos. Los primeros vehículos permitieron realizar pruebas de funcionamiento de los algoritmos que se implementaron para las dinámicas de éstos por ello no requerían mayor trabajo estético. La implementación de nuevos modelos más elaborados estéticamente no impidió que los algoritmos ya realizados tuvieran que modificarse ya que se acomodaron a ellos. Los detalles de los nuevos modelos de autos se asemejan a los reales por lo que la implementación de éstos mejora significativamente el arte de los mismos.

3.1.5.3 *Diseño de interfaces.*

Al momento de diseñar el menú se tuvo en cuenta tres elementos que fueron el fondo, los botones y la ubicación de éstos. En cuanto al fondo se decidió poner una imagen de la ciudad desde una vista superior, donde se puede observar la arquitectura y varios componentes del escenario. Los botones tomaron una forma clásica y sencilla que le dio un aspecto de seriedad a la aplicación. Finalmente se optó por distribuir todos los botones en el centro de la imagen que permitió una mejor visualización de éstos.

3.1.5.4 *Diseño de niveles.*

Luego de determinar cómo en diferentes partes de la ciudad se generaban una serie de dinámicas relacionadas al tránsito de vehículos, la forma de conducir y las infracciones que se comenten, se decidió diseñar los niveles de la aplicación dentro del mismo escenario pero ubicados en distintos puntos del mapa. Es por ello que cuando el usuario supera un nivel y pasa al siguiente va a tener la

sensación en varias ocasiones de estar en el mismo nivel por los elementos similares que encuentra entre niveles, sin embargo, aunque la arquitectura es similar, siempre va a empezar en diferentes lugares del escenario.

3.1.5.5 Diseño de actividades o situaciones de movilidad.

Las actividades o retos que se diseñaron en la aplicación surgen a partir de la observación del equipo de trabajo sobre situaciones de movilidad que ocurrían frecuentemente en su contexto. Se la diversa gama de condiciones e infracciones que se observaron se determinó que hubo 4 que predominaron, el cruce indebido en el carril incorrecto, el parqueo prohibido sobre vías principales, el estacionamiento ilegal sobre intersecciones y el irrespeto a la demarcación de línea continua. Estas situaciones fueron las que se se adecuaron en el escenario de diversas formas sin obviar otro tipo de infracciones como pisar la cebrá cuando el semáforo está en rojo, pasarse los semáforos en rojo, atropellar a otros vehículos, entre otras.

3.1.5.6 Diseño de visualización.

Desde la propuesta de anteproyecto planteada siempre se referencio la idea de realizar una visualización basada en realidad virtual inmersiva, es por ello que las gafas de realidad virtual se hacen una herramienta trascendental en el diseño de la visualización, por supuesto es posible poder observar la aplicación sin ninguna herramienta adicional a la que proporciona la propia vista, sin embargo la aplicación fue diseñada para que el usuario experimente este tipo de nuevas tecnologías en situaciones que emulan su cotidianidad, por lo que la visualización de la aplicación aún sigue exigiendo el uso de las gafas así como el de otra serie de herramientas como el teclado convencional de un computador y teléfono móvil.

3.2 MECANISMOS DE FUNCIONAMIENTO.

La aplicación comienza dentro del menú de inicio donde el usuario podrá elegir una de las cuatro opciones que presenta éste; inicio, ayuda, créditos y salir hacen

parte del menú. La figura 10 presenta una imagen donde se muestran las cuatro opciones que hacen parte del menú de inicio de la aplicación.

Al elegir la opción de inicio, el usuario entra directamente al primer nivel en donde tiene la posibilidad de conducir un único automóvil que será dirigido por medio de las teclas de dirección del teclado convencional de un computador. La visualización de la posición actual del automóvil así como de su entorno se hará a través de unas gafas de realidad virtual. Ya inmerso en el mundo virtual, el automóvil, controlado por el usuario, podrá navegar a través del mapa explorando las vías y contemplando el paisaje y la arquitectura creada, sin embargo, el objetivo principal en cada uno de los niveles es cumplir una serie de desafíos programados bajo la dinámica de llegada de un punto a otro sin cometer ningún tipo de infracción. Una vez se haya cumplido el objetivo del nivel, se muestra un mensaje de éxito en pantalla y automáticamente se prosigue al siguiente nivel así sucesivamente hasta completar todos los niveles. Las figuras 11, 12 y 13 ilustran algunas imágenes de las mecánicas que se llevan a cabo dentro de la aplicación.



Figura 10. Captura del menú principal de la aplicación.



Figura 11. Imagen de notificación de éxito en el nivel.



Figura 12. Imagen de notificación de infracción cometida.

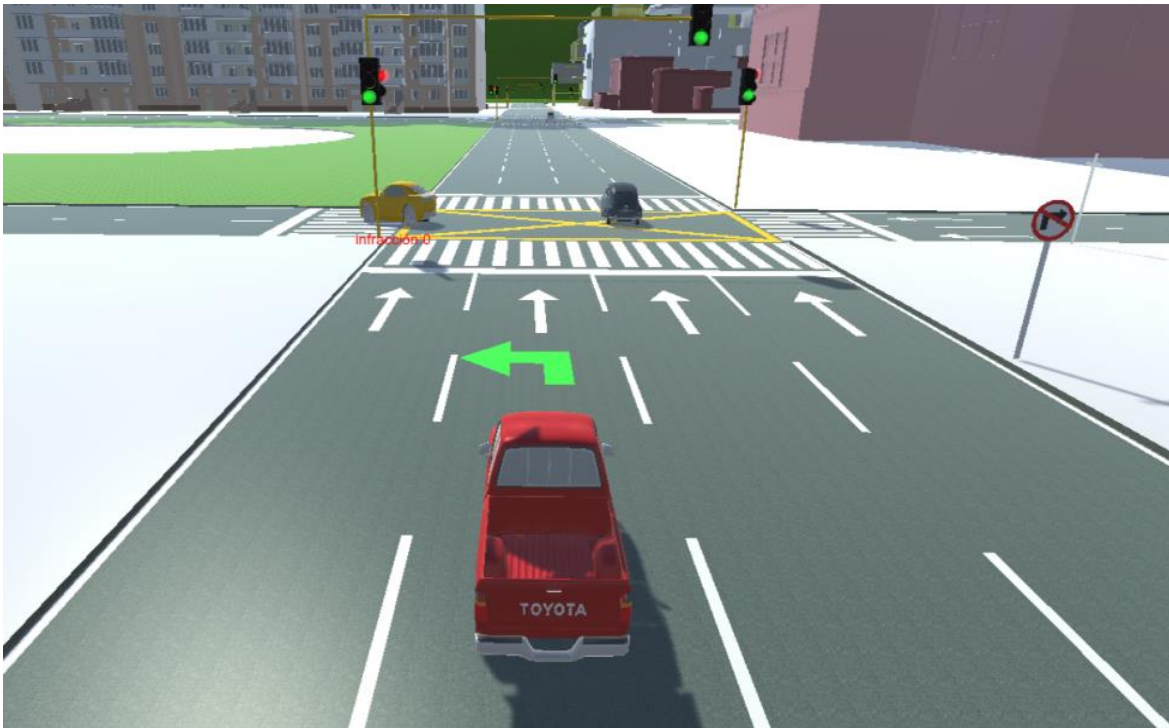


Figura 13. Captura de imagen del vehículo controlado explorando el escenario.

Otra de las características relacionadas a las mecánicas de la aplicación es que ésta no requiere ser detenida en determinado momento, es posible pausar la aplicación y continuar dentro del mundo virtual sin que ocurra alguna eventualidad pues no se ha contemplado un tiempo determinado para la culminación de cada nivel. Además de esto, se tendrá un botón para el reinicio de la aplicación en caso de que ésta se bloquee u ocurran sucesos inesperados así como una tecla que permite visualizar el mapa que muestra la ubicación actual del automóvil dentro del escenario y el punto al cual debe dirigirse el usuario. En el Capítulo 4 se profundiza más acerca de la interfaz de usuario de la aplicación.

3.3 PROCESO DE VISUALIZACIÓN.

Para el proceso de visualización de la aplicación se hace necesaria la sincronización de los dispositivos que se van a usar para recrear la realidad a partir de un ambiente virtual inmersivo como se referencia en el objetivo general del proyecto.

Gracias a la herramienta multiplataforma que ofrece Unity 3D es posible el despliegue de la aplicación en diversas plataformas, sin embargo, para este caso particular el desarrollo de la aplicación se realizó para su despliegue bajo la plataforma de dispositivos PC. Para realizar éste proceso se contó con un computador portátil Lenovo G40-70 donde se desarrolló totalmente la aplicación. Otro dispositivo necesario para la visualización es un teléfono móvil cuyo requerimiento principal es una pantalla de 5 pulgadas o más puesto que las gafas de realidad virtual, las cuales soportan el móvil, han sido diseñadas y construidas para teléfonos con esta característica.

Con relación al software es necesario descargar un programa, tanto en el computador como en el móvil, para la sincronización de ambos dispositivos. Trinus VR²³ es una aplicación gratuita que permite al usuario revivir juegos y aplicaciones en realidad virtual sin demandar costos elevados, éste utiliza la pantalla y los sensores del móvil para transformarlo en un portal para aplicaciones y juegos de computador.

Una vez instalado Trinus VR tanto en el computador como en el teléfono móvil, se procede a sincronizar los dispositivos que puede hacerse de dos formas, vía cableada (USB) o vía inalámbrica (WIFI). Si se hace por medio de un cable USB, se conectan ambos dispositivos al cable, al abrir Trinus VR y cliquear el botón de encendido en ambos dispositivos, éstos se reconocerán de manera automática posibilitando la inmersión dentro del mundo virtual. Si se realiza a través de una red WIFI es necesario que ambos se conecten a una misma red, una vez conectados, se obtiene la dirección IP del dispositivo móvil en la configuración de la red en el teléfono, luego en la aplicación Trinus VR del PC se digita la dirección IP del móvil en los ajustes de red y finalmente se encienden ambos botones de la aplicación para comenzar con la experiencia inmersiva. Si no se cuenta con una red WIFI disponible para la conexión de ambos dispositivos, es posible crear una red inalámbrica propia y conectarlos.²⁴

²³ <http://trinusvr.com/>

²⁴ <http://www.elandroidelibre.com/2015/04/como-crear-una-red-wifi-desde-nuestro-ordenador.html>

Capítulo 4.

4 DESARROLLO DE LA APLICACIÓN.

4.1 MODELADO Y ARTE.

Es imprescindible señalar que el arte de la aplicación fue hecho totalmente por el equipo de trabajo, éste se fue construyendo al paso, al tiempo que se realizaban las modificaciones y correcciones pertinentes. Tanto el escenario como los vehículos y otros elementos como los semáforos, en principio fueron hechos en software CAD especializado a partir de polígonos y figuras sencillas que al exportarse al correspondiente motor de videojuegos no consumieran mucho recurso de máquina y sirvieran como objetos de prueba, posteriormente se reemplazaron por modelos más elaborados visualmente. En la figuras 14 y 15 se presentan los dos contextos, primero el arte inicial de los elementos y luego el arte definitivo respectivamente.

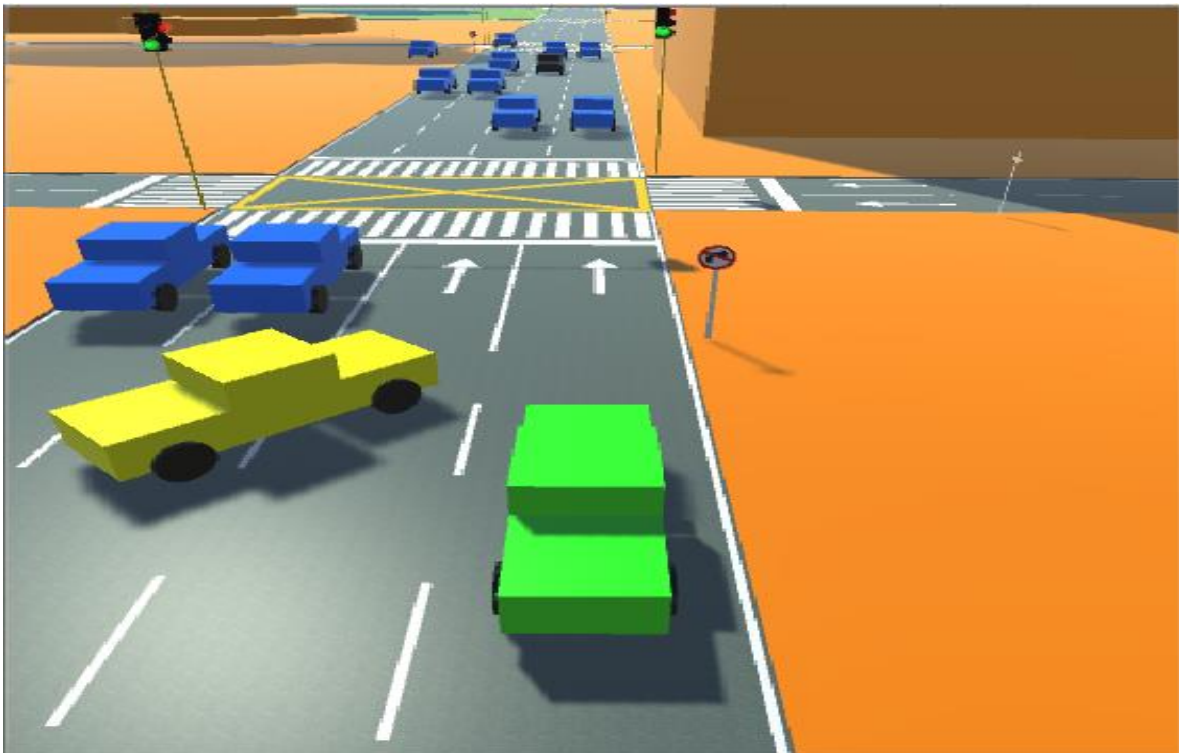


Figura 14. Arte inicial de la aplicación.



Figura 15. Arte final de la aplicación.

En ésta aplicación los vehículos son los elementos principales, aquí se pueden observar coches autónomos que han sido programados para que transiten libre e independientemente por el escenario cumpliendo las características propias de un automóvil, es posible agregar tantos carros autónomos como el usuario desee. Por otro lado, existe un automóvil controlado por el usuario a través de las teclas de dirección del teclado el cual ha sido programado para que quien lo conduce tenga control total sobre él.

4.1.1 Diagramación, esquematización y geometría 3D

Ésta se proyectó en dos etapas, por un lado, la diagramación o esquematización de interfaces, elementos (vehículos y semáforos) y escenarios a partir de bocetos hechos a mano que para efectos de la aplicación no requerían de mucha elaboración y permitían modificación a requerimientos lo que posibilitó el uso de figuras geométricas básicas y algunas construidas a partir de polígonos. En una segunda etapa, se procede a la digitalización de los bocetos en el respectivo software CAD que de la misma manera estaban propensos a modificaciones.

4.1.2 Digitalización.

4.1.2.1 Escenario y objetos (semáforos, señales de tránsito).

El siguiente paso para la consecución de la aplicación consiste en digitalizar los esquemas y bocetos generados anteriormente. Digitalizar no es más que realizar el modelado de los elementos y escenarios en un software CAD especializado. La figura 16 presenta la digitalización del primer plano del escenario.

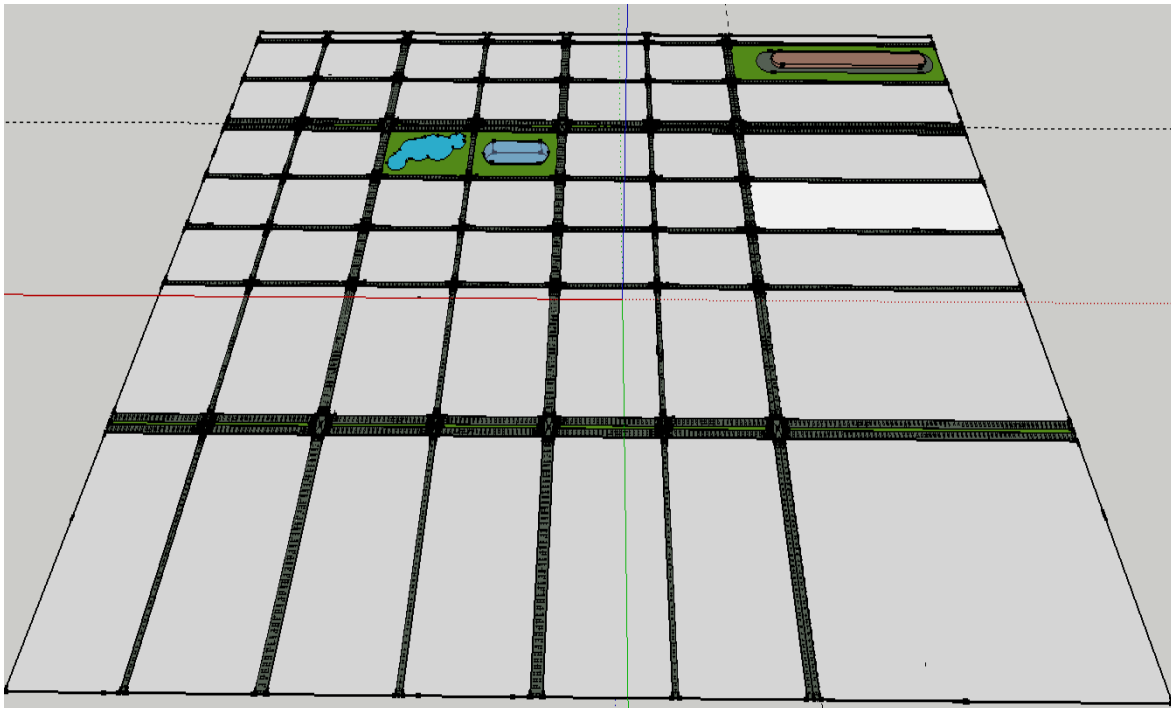


Figura 16. Croquis o plano de la ciudad modelada en SketchUp.

Una vez modelado el elemento se exporta con extensión .fbx a Unity 3D donde queda listo para ser utilizado en el momento que se requiera. Los elementos que se han exportado al motor de videojuegos se importan a la escena y se acomodan para conformar el escenario completo. La figura 17 muestra el escenario inicial creado para la aplicación, aquí se ven formas tridimensionales que emulan la arquitectura de modelo. La figura 18 muestra un tipo de semáforo doble que se implementa en vías específicas mientras que las figuras 19 y 20 presentan las formas de los semáforos construidos con sus respectivas modificaciones.

Finalmente la figura 21 presenta el modelado de las vías y señales de tránsito con su respectiva demarcación que han sido creadas en el escenario junto con los demás elementos.

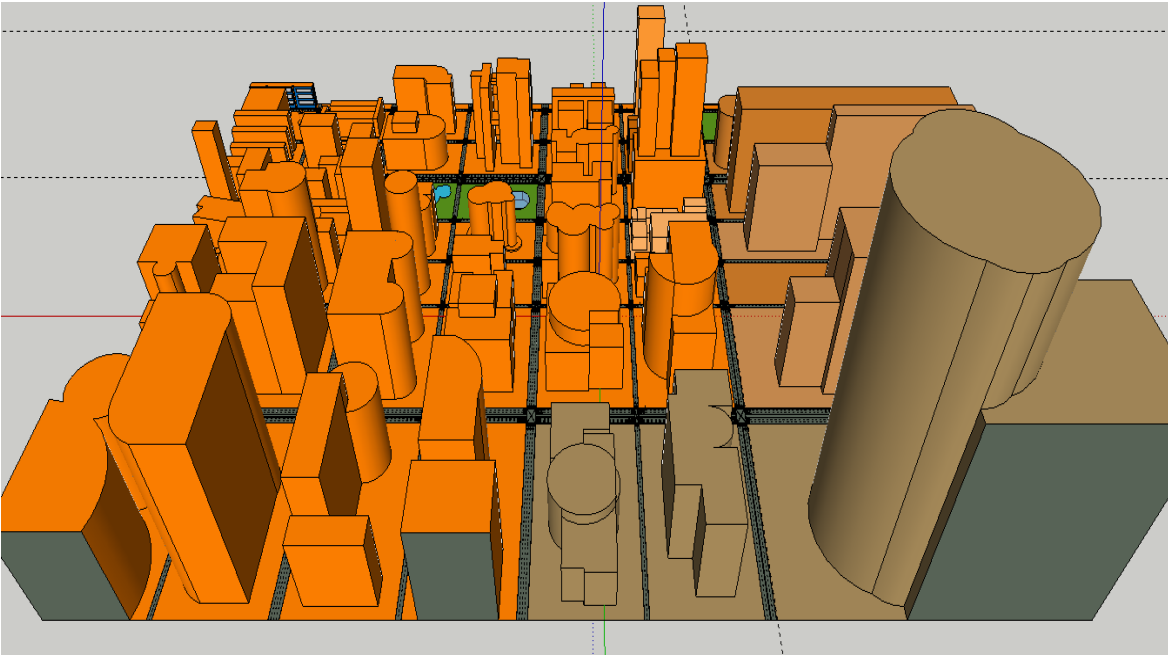


Figura 17. Modelado inicial del escenario.

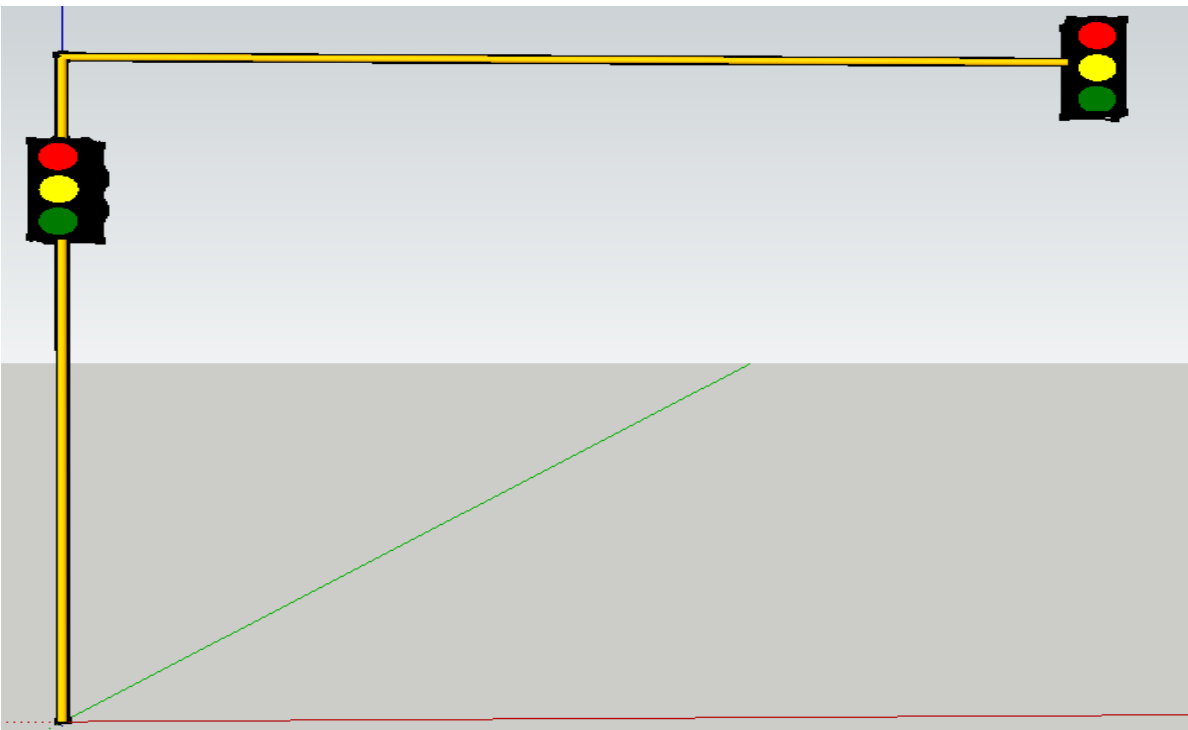


Figura 18. Modelado de semáforo doble.

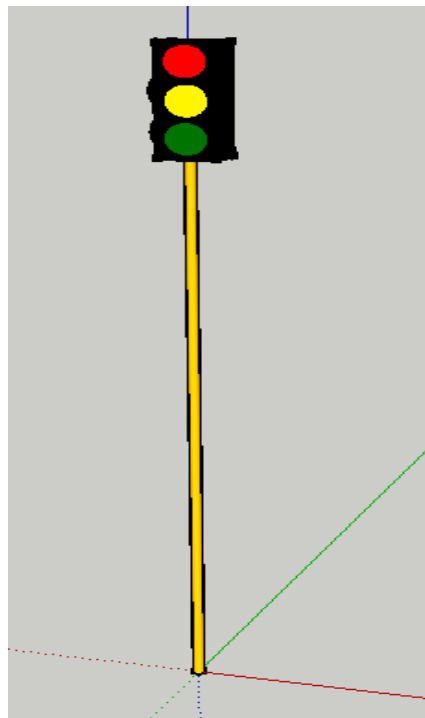


Figura 19. Primer modelo 3D de semáforo sencillo.

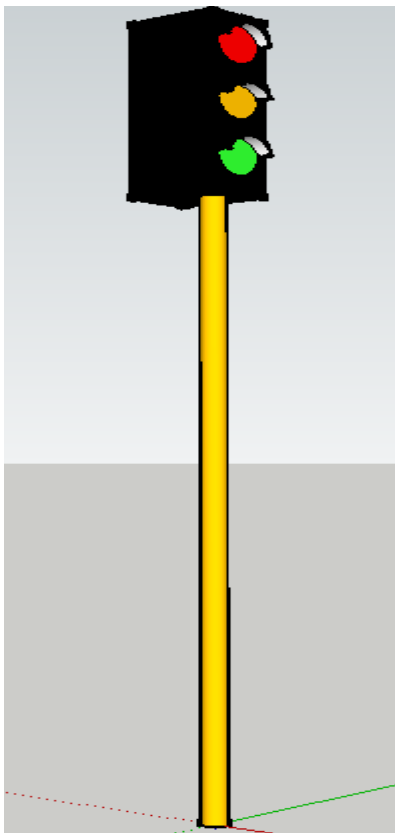


Figura 20. Semáforo sencillo mejorado.



Figura 21. Modelado de vías y señales de tránsito en SketchUp.

Para generar un aspecto visual más llamativo y realista, se optó por realizar modificaciones tanto al modelo del escenario como al modelo de los vehículos. Con respecto al escenario, se tomó el modelo generado inicialmente y se eliminaron las figuras que representaban la arquitectura, éstos se reemplazaron por modelos más elaborados que mantuvieran sus características de diseño con pocos polígonos. El escenario final se ilustra en la figura 22.



Figura 22. Escenario final de la aplicación.

4.1.2.2 Vehículos.

Los vehículos fueron modelados directamente desde el motor de videojuegos para mayor facilidad de orientación de éstos, de la misma forma que los demás elementos, en principio se construyeron a partir de figuras tridimensionales como cubos y esferas posteriormente se reemplazaron por autos más elaborados visualmente teniendo como resultado los vehículo finales que se muestra en la figura 23.

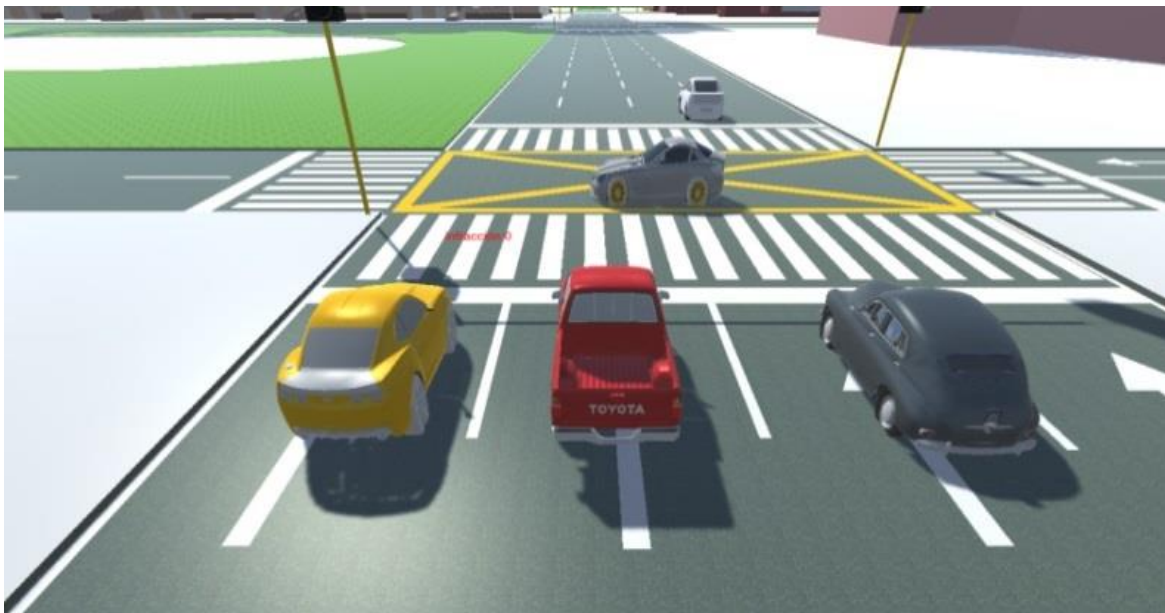


Figura 23. Modelos finales de automóviles.

Dentro del escenarios será posible encontrar dos tipos de vehículos, por una lado, un automóvil controlado que será aquel que el usuario puede manejar con las teclas de dirección del computador, éste ha sido programado con condiciones y características específicas y diferentes al otro tipo de coches. El otro tipo de vehículo es aquel capaz de moverse de manera autónoma dentro del escenario, las dinámicas de movimiento de los vehículos autónomos no depende exclusivamente de ellos, el escenario ha sido adecuado estratégicamente con objetos y elementos específicos que permiten que el automóvil ejecute diferentes acciones. En la figura 24 se presenta el modelos del vehículo controlado que es único, en la figura 25 se muestra uno de los modelos de los carros autónomos, de éstos se podrán encontrar diversos modelos dentro del escenario.



Figura 24. Automóvil controlado.



Figura 25. Vehículo autónomo.

Una vez modelados todos los elementos y exportados a Unity 3D se procede a integrarlos en la escena, para ello se arrastran a ella, en primera instancia el escenario, posteriormente se añaden los vehículos, semáforos y demás elementos que conforman la ciudad. Unity posee una herramienta muy útil llamada Prefab, ésta permite almacenar un objeto con sus componentes y propiedades actuando como una plantilla a partir de la cual se pueden crear o modificar parámetros del objeto en la escena y pueda reutilizarse varias veces dentro de ella. Utilizando la herramienta Prefab se logró distribuir por la ciudad varios de los elementos que cumplen las mismas características como semáforos y automóviles. La figura 26 muestra la integración de los objetos que componen la aplicación dentro de una escena en Unity.



Figura 26. Integración del escenario y los objetos en la escena.

Como se observa en la figura 26, los carros tienen diferentes colores así como los andenes y edificios, esto se logra manipulando los materiales de los objetos los cuales inciden en su visualización. El material modifica parámetros de textura, sombreado, luminosidad, entre otros, para este caso particular se utilizó para realizar modificaciones de color sobre los objetos. Los colores de las vías y señalizaciones se realizaron desde el modelado en SketchUp, sin embargo es posible modificarlos desde Unity una vez se hayan puesto sobre la escena.

4.1.2.3 Interfaz

Para diseñar el menú de la aplicación se tuvo en cuenta 3 elementos principales, la imagen de fondo, los botones y el contenido de cada uno de ellos. El aspecto visual tanto del menú como de los botones no tuvo un desarrollo elaborado.

En relación al fondo, se representó a partir de una imagen en vista superior que simboliza la ciudad final modelada, aquí se muestra la arquitectura de la ciudad con algunos de los objetos que lo componen, esto permite dar un indicio del contenido de la aplicación.



Figura 27. Captura del fondo utilizado para el menú de inicio de la aplicación

En cuanto a los botones, éstos fueron diseñados de forma sencilla dando un aspecto clásico similar a lo que se pueden encontrar en las ventanas emergentes o notificación de errores en Windows, cada uno contiene una tipografía denominada palo seco la cual da un aspecto de seriedad a los botones del menú.

Los botones que conforman el menú de inicio son: botón de inicio que permite ejecutar la aplicación, botón de ayuda que describe los controles y las teclas a usar para la interacción del usuario con la aplicación, botón de créditos donde se visualiza y detalla los nombres de los creadores de la herramienta y botón de salir que permite abandonar totalmente la aplicación.

Capítulo 5.

5 PROGRAMACIÓN DE LA APLICACIÓN.

5.1 REQUERIMIENTO DE PROGRAMACIÓN.

El siguiente paso en el proceso de desarrollo de la aplicación hace énfasis en la programación de los objetos y que la componen. La programación se ha hecho bajo la técnica POO o programación orientada a objetos. De la misma forma que el arte, el proceso de programación tiene una serie de requerimientos relacionados con los Scripts (programas o códigos) que permiten el funcionamiento de los elementos que componen la aplicación, A continuación se referencian y describen estos requisitos.

5.2 ANIMACIÓN.

El desarrollo de las animaciones para la aplicación se realizó a partir de una técnica llamada Scripting de animación la cual realiza animación de objetos por medio de Scripts o archivos de órdenes. Los vehículos y los semáforos fueron los únicos objetos animados a partir de la técnica mencionada. Con respecto a los semáforos, se programó la secuencia de los mismos, mientras que en relación a los autos, se animaron los movimientos que podían realizar. Para ambos casos se explica el procedimiento a continuación.

5.2.1 Control general de la aplicación.

Los scripts relacionados a este requerimiento conciernen directamente al menú principal de la aplicación. Una vez se ejecuta ésta, el usuario queda automáticamente dentro del menú de inicio donde encontrará 4 botones correspondientes a inicio, ayuda, créditos y salir respectivamente. El botón de inicio permite acceder a la aplicación mientras el de salir la cierra completamente. Los otros dos botones, tanto el de ayuda como el de créditos, dirigen al usuario a

escenas donde se accede a información referente a los controles de la aplicación y datos de los desarrolladores de la misma. Dentro de las escenas de ayuda y créditos el aplicador podrá encontrar un botón adicional que permite el retorno al menú principal mientras que para una vez se encuentre dentro de la aplicación la tecla Esc del computador permitirá salir completamente de ésta. En el menú de ayuda se permiten conocer las mecánicas para la ejecución.

5.2.2 Control de vehículos.

Este requerimiento de programación permite determinar y generar los scripts que controlan los automóviles. Básicamente se crearon dos tipos de scripts, uno que permite el control de los coches autónomos y otro para el auto controlado. Los vehículos autónomos poseen condiciones que les permiten desplazarse libremente por el escenario así como ejecutar acciones propias de los vehículos reales. En cuanto al coche controlado, sus mecánicas de funcionamiento se han limitado al uso de las teclas de dirección de computador o a las teclas W, S, A y D. Éste último posee condiciones diferentes a las de los coches autónomos y de igual forma cumplen tareas específicas dentro de la aplicación.

5.2.3 Control de semáforos.

En este requerimiento de han desarrollado los scripts que simulan el comportamiento real de un semáforo. Dentro de la aplicación existen dos tipos de semáforos, uno sencillo que contiene la secuencia de luces tanto para los autos como para los peatones y que ha sido ubicado sobre vías de uno o dos carriles. Por otro lado, un semáforo doble ubicado sobre vías de 3 o más carriles con la misma secuencia para ambos y teniendo en cuenta que para efectos del modelado uno solo tendría las luces para el peatonal. La sincronización de los semáforos se hizo a poro medio de los tiempos de encendido y duración de cada luz, es por ello que mientras un semáforo puesto en una carrera tiene una secuencia definida aquel que ha sido ubicado sobre una calle tendrá la secuencia inversa. Adicionalmente los scripts de los semáforos permiten el control de otros objetos que hace parte de la dinámica propia del escenario, por ejemplo se

controla la aparición de muros invisibles para el frenado de los coches autónomos o se activan los Halos que dan una apariencia de luminosidad a cada color del semáforo. Existen 5 tipos de scripts para los semáforos que en el fondo tienen la misma programación, lo que los diferencia son acciones concretas que deben ejecutar y que se relacionan al control de cubos para el giro de coches entre otros.

5.2.4 Control de notificaciones.

La función relacionada a estos scripts es controlar los mensajes de notificación que podrán ser visualizados por el usuario. Se han programado dos tipos de notificaciones, una simbólica que muestra en pantalla figuras que permiten al usuario evidenciar que está cometiendo algún tipo de infracción. Otro tipo de notificaciones corresponde a mensajes textuales que el usuario puede observar para retroalimentar su desempeño dentro de la aplicación, es decir, si ejecutando una actividad el usuario pisa una cebra estando el semáforo en rojo, automáticamente se muestra un mensaje en pantalla avisando la consecución de una infracción junto con una frase alusiva a la misma, asimismo habrá un contador de multas que incrementa cada vez que se comete una. Cada vez que se culmine exitosamente un nivel, el usuario recibirá un informe de desempeño en la pantalla el cual muestra cuantas infracciones se cometieron por tipo los cuales se han dividido en cebra, intersección y cruce en carril indebido.

5.2.5 Control de interfaces.

Los scripts relacionados a éste requerimiento permiten el manejo de la interfaz de las escenas, controla las teclas que permiten pausar, visualizar el mapa, reiniciar los niveles y salir completamente de la aplicación. Dentro de las escenas se han dispuesto imágenes estáticas que de acuerdo a diversas causalidades se habilitan o deshabilitan, por ejemplo, existen flechas que se activan cierto tiempo y que sugieren al usuario tomar una ruta determinada para cumplir cierta actividad.

5.3 REFERENCIA DE LOS SCRIPTS.

Luego de conocer los requerimientos de la aplicación en términos de programación se procede a referenciar los scripts utilizados para la programación de los objetos, la tabla 4 hace el recuento de éstos scripts, a saber:

Tabla 4. Referencia de Scripts utilizados en la programación de la aplicación.

Nro.	Nombre del Script	Función
1	C_autonomo.cs	Este script controla el comportamiento de autonomía de los vehículos que contienen las dinámicas para transitar libremente por el escenario. Aquí se programan las condiciones que tienen los carros autónomos y que les permite asemejarse a los de la vida real.
2	camara.cs	Controla los movimientos de la cámara que sigue al automóvil controlado y a su vez enfoca el escenario por donde se desplaza. También se encarga del cambio de cámara para visualizar el mapa.
3	Carrerasindoblar.cs Semáforo_nuevo.cs Semáforo_nuevo_carrera.cs SemaforoC2.cs SemaforoNuevoSindoblar.cs	Este script controla la secuencia de visualización de los semáforos distribuidos por la ciudad. Permite el control de algunos objetos necesarios para las dinámicas de los vehículos autónomos.
4	carro_quieto.cs	Aquí se anima el avance de los carros hacia la cebra para luego dejarlos estáticos sobre la escena y que propician la consecución de los retos propuestos.
5	carroplayer.cs	Este script se encarga de las dinámicas de movimiento del auto controlado por el usuario. Posee atributos de aceleración, velocidad y desaceleración los cuales se orientan con las teclas de dirección.
6	Cebra.cs Cebra_verde.cs	Para detectar si el vehículo está parado sobre una cebra es necesario poner un cubo invisible sobre ella, este script activa y desactiva los cubos que están ubicados en calles o carreras respectivamente.

7	creditsin.cs	Una vez dentro de la opción créditos en el menú inicial de la aplicación este script permite el retorno al menú nuevamente.
8	guia.cs	Permite activar elementos dentro de las escenas como flechas y prohibidos, además contabiliza las infracciones que se cometen por pisar la cebra, frenar en la intersección y girar en carril indebido.
9	In_ayuda.cs	Una vez dentro de la opción ayuda en el menú inicial de la aplicación este script permite el retorno al menú nuevamente.
10	menu1.cs	Con este script se controlan las opciones del menú principal. Una vez cliqueada una de las opciones el algoritmo envía la aplicación a un escena que ha sido adaptada para cada opción.
11	metaguia.cs	Permite determinar la ubicación actual de la meta en cada nivel, una vez establecida, el script actualiza la nueva posición de la meta en el nivel y mueve el marcador a esa nueva posición para que pueda visualizarse en el mapa.
12	pausa.cs	Programa el proceso de pausa de la aplicación.

5.4 DESCRIPCIÓN DE LOS SCRIPTS PRINCIPALES

A continuación se hace una descripción sintética de las formas en que se elaboraron y programaron las dinámicas de los objetos en los scripts.

C_autonomo.cs: Este script se encarga del control de los carros que se mueven libremente por el escenario. La primera acción que puede ejecutar éste objeto es avanzar, para ello se generan una serie de atributos como aceleración y velocidad en el script los cuales se ejecutan una vez se reproduce la aplicación. Estos atributos se relacionan a unas variables que se inicializan en un valor determinado las cuales, a partir de un algoritmo, realizan la animación de desplazamiento del vehículo. La segunda acción que puede ejecutar el auto es frenar, para esto se trazan 3 rayos invisibles desde el centro del objeto hasta una distancia en el eje X

establecida por el programador, una vez generados los rayos, por medio de un algoritmo se modifican los atributos de velocidad y aceleración del objeto cuando los rayos se encuentren con un obstáculo justo en frente de ellos. Los obstáculos se representan con planos invisibles u otros vehículos los cuales poseen un componente llamado Collider o colisionador que detectado por los rayos permiten que se ejecute el algoritmo. Una tercera acción que pueden realizar los vehículos autónomos es esquivar otros coches, esto se logra programando dos de los tres rayos que se han generado con antelación, así pues, si el rayo de frente y el rayo del lado derecho detectan un Collider, el carro esquivará por la izquierda y si el rayo de frente y el rayo izquierdo son quienes detectan el obstáculo esquivará por el lado derecho. El movimiento que realiza el auto para esquivar se programa a partir de los ángulos de orientación del objeto los cuales lo rotan en el eje Y cierta cantidad de grados de acuerdo al lado por donde quiere esquivar mientras sigue desplazándose, una vez lo haya hecho retoma la posición inicial en la que venía y continúa su marcha hacia el frente. Una última acción que pueden efectuar los vehículos autónomos es la de decidir si desea girar en las esquinas o seguir derecho. Para esto, se disponen cubos en todas las esquinas de calles y carreras a los cuales se desactiva el componente llamado Mesh Renderer el cual lo hace invisible manteniendo su componente Collider, para que el Collider del vehículo no colisione con el del cubo se activa la opción de Trigger en el carro, esto admite que cuando ambos Colliders choquen puedan atravesarse sin problema. Para animar el movimiento de giro, los rayos generan un número aleatorio cada vez que éstos detecten un obstáculo, si el número es superior a un valor el carro gira, gracias al cubo, aproximadamente 45 grados en relación al ángulo inicial en el que se encontraba, cuando termine de girar mientras se desplaza hacia delante, continuará su trayectoria en línea recta, si el número es inferior al valor el carro no gira y continúa su camino en línea recta.

carroplayer.cs: Este script configura las dinámicas de control del auto que maneja el usuario así como las mecánicas interacción de éste con la aplicación. En primera instancia, se programaron los botones de desplazamiento y dirección

del coche estableciendo las teclas W, A, S, D o los cursores de dirección para el control del vehículo. Una vez el usuario presiona las teclas W - S o flechas arriba - abajo se modifican los atributos de velocidad y aceleración programados en el algoritmo sumando o restando unidades según sea el caso, lo que produce el efecto de desplazamiento o frenado del automóvil. Algo similar ocurre con la dirección del coche, si el usuario presiona las teclas A, D o las flechas izquierda - derecha se modifican los ángulos locales del objeto lo que genera la sensación de rotación de éste. Si el usuario presiona dos teclas a la vez, una de desplazamiento y una de dirección, el objeto cumplirá ambos parámetros, es decir, se desplazará al mismo tiempo que irá rotando. El escenario ha sido diseñado para que el usuario que conduce el vehículo controlado esté propenso a cometer alguna infracción, es por ello que éste vehículo tiene un cubo invisible puesto sobre el techo que al colisionar con otros cubos distribuidos por la ciudad indican que se ha cometido una infracción y generan acciones programadas en el script como reiniciar el nivel, mostrar un mensaje de infracción, contabilizar la cantidad de infracciones entre otros.

Carrerasindoblar.cs, Semáforo_nuevo.cs, Semáforo_nuevo_carrera.cs, SemaforoC2.cs, SemaforoNuevoSindoblar.cs: Básicamente cada uno de estos scripts desarrolla la misma lógica de programación, lo que diferencia un script del otro es que debido a la distribución de los semáforos y los cubos que permiten el giro de los vehículos, además del diseño de las vías, fue necesaria la división de los scripts para que unos controlaran las calles y otros las carreras, también para que cuando un carro no decidiera girar en una esquina no se creara conflicto con otros cubos de giro que se encontrara a su paso. Para generar la secuencia del semáforo se crean 3 objetos de tipo Sphere que representan cada uno de los colores, se ubican respectivamente en el modelo exportado desde SketchUp y que fue arrastrado a la escena, posteriormente se cambia a color negro el material de las esferas y se agrega el componente Halo a cada una de ellas para darle un toque de luz a cada objeto. El algoritmo permite establecer el tiempo de duración de encendido de cada una de las luces del semáforo que se generan cambiando

el material de las esferas de color negro al color que corresponda en el semáforo y habilitando el Halo correspondiente, el mismo proceso se realiza para las luces del semáforo peatonal. Desde los semáforos es posible habilitar un plano invisible que se ha puesto al borde de las cebras para controlar los vehículos autónomos que estén transitando, de esta manera, cuando el semáforo está en rojo y rojo-amarillo se habilita el collider del plano para que pueda ser detectado por los rayos de los autos y frenen justo antes de pisar la cebra, una vez el semáforo cambia a verde el collider se deshabilita y los carros pueden avanzar

guia.cs: La realimentación al usuario de las infracciones cometidas es uno de los aspectos más importantes en la aplicación. Dentro de cada escena se han ubicado imágenes estáticas de flechas de sugerencia, símbolos de prohibido y notificaciones con texto alusivo al mal uso de las señales de tránsito. Con este script se habilitan estas herramientas visuales para guiar al usuario en la aplicación. Por otro lado, este algoritmo permite contabilizar la cantidad de infracciones que se han dividido por tipos, por un lado las que suceden al pisar la cebra, luego las que se producen al estacionar sobre una intersección y finalmente las que se generan por el giro sobre un carril indebido, cada vez que se comete una de éstas, un contador interno de infracciones por tipo las almacena y una vez es terminado el nivel se muestra un mensaje de notificación mostrando el desempeño del usuario en cada nivel.

Capítulo 6.

6 DISEÑO E IMPLEMENTACIÓN DE LAS ACTIVIDADES O SITUACIONES DE MOVILIDAD.

6.1 DESCRIPCIÓN DE LAS ACTIVIDADES.

En este capítulo se detallan los retos que han sido propuestos y desarrollados en la aplicación, para ello se hace una breve descripción del desafío y posteriormente como ha sido implementado dentro del respectivo escenario.



Figura 28. Actividad 1. Giro en esquinas por el carril correcto.



Figura 29. Imagen relacionada a la actividad de giro.

En la actividad que se muestra en las figuras 28 y 29, el usuario debe llegar de un punto a otro, para esto debe girar por una de las esquinas del escenario, aquí deberá tomar el carril correcto para el giro, con antelación se muestra una flecha que indica la dirección que debe tomar. El semáforo que se encuentra en el cruce debe ser respetado y de la misma manera el giro debe ser hecho correctamente, de no ser así, el contador de multas mostrado en la parte superior izquierda empezará a incrementarse de acuerdo a la cantidad de infracciones que se cometan, cuando dicho contador acumule 3 multas el nivel se reiniciará y el usuario deberá realizar de nuevo y completamente el nivel. Antes de comenzar cada una de las actividades el usuario podrá observar en un mapa el trazo del camino que debe seguir, es por ello que el nivel también se reiniciará si éste toma un camino diferente al que se muestra en el mapa. Una vez finalizada la actividad se muestra en pantalla, en modo de retroalimentación, el desempeño del usuario durante el nivel evidenciando cuantos tipos de infracciones ha cometido dentro del mismo.

Para propiciar que el usuario cometa la infracción de han dispuesto vehículos que transiten junto con el auto controlado, éstos sirven como guías de camino que posiblemente sean referencia para conducir, de la misma manera la demarcación de las vías sugiere los caminos por los que deben transitar los automóviles. Una situación ha sido generada adrede con el fin de apoyar la consecución de las infracciones, es el caso de ubicar vehículos lo suficientemente lentos y cercanos a los semáforos en vías de cruce que inciten a que el usuario adopte una maniobra de esquite y deba retomar nuevamente el carril para poder girar, de no ser así, acumulará una infracción.

El próximo reto a cumplir se muestra en la figura 30, el usuario debe realizar nuevamente giros en las esquinas sin incumplir la norma. De la misma manera que el nivel 1, este nivel tiene los mismos requerimientos relacionados con los semáforos y los carriles para el giro, así mismo se cumplen las mismas condiciones de no ser realizada correctamente la actividad.



Figura 30. Actividad 2. Imagen cambio de carril en línea continua.

Se ha añadido una nueva condición para éste reto la cual consiste en realizar cruces de carril sobre demarcación de línea continua en el suelo que según la norma está prohibido hacer.

Para posibilitar la consecución de la infracción relacionada al cambio de carril sobre línea continua, adicional a las condiciones que se nombraron en el primer nivel, se ha dispuesto una fila de autos lentos que estarán transitando unos metros antes de llegar al semáforo con el fin de que el usuario deba evadirlos y retomar el carril teniendo en cuenta y observando que dicho cruce exige hacerse únicamente en línea discontinua pues de no ser así estará incumpliendo la norma de tránsito que sumará una infracción.

El siguiente reto correspondiente a la figura 31 representa una de las infracciones más recurrentes en las vías y se trata del estacionamiento sobre la señal de intersección de color amarillo que se muestra en la imagen sobre la vía. Según la normatividad de tránsito vigente ningún vehículo automotor puede ubicarse encima de ésta demarcación cuando haya congestión en la vía así el semáforo se encuentre en verde pues es posible generar accidentes o un atasco mayor.



Figura 31. Actividad 4. Imagen de atasco y vehículos sobre la intersección.

Para ésta actividad se han dispuesto una serie de vehículos que generan trancón lo que produce que los carros que requieren seguir derecho se ubiquen sobre la demarcación de intersección haciendo propenso al usuario a cometer ésta infracción. En éste reto es posible que el usuario realice dos maniobras, primero que se tome un tiempo de espera mientras los autos atascados delante de él reanuden su marcha y la zona demarcada quede libre para poder transitar o que el usuario tome una ruta alternativa para llegar al lugar de destino.

Como se realizó en las actividades anteriores las condiciones para la actividad continúan cumpliéndose de la misma forma, por ello el usuario debe tener en cuenta los semáforos, girar por los carriles adecuados, evitar a toda costa embestir a otros vehículos, subirse a los andenes, tomar rutas equivocadas, no transitar en contravía y todos aquellos requerimientos de tránsito que se atienden en la vida real.



Figura 32. Imagen de parqueo indebido sobre vías principales.

La figura 32 presenta una imagen de una situación en donde el auto controlado debe esquivar varios carros parqueados en la vía. Éste contexto fue planteado con el fin de que el usuario pudiera evidenciar y comprender los riesgos que implica parquear en vías principales lo que en muchas ocasiones genera accidentes fatales, atascos en las vías y en el mayor de los casos, comportamientos de intolerancia y violencia por parte de los conductores involucrados.

Con respecto al diseño e implementación de las actividades descritas anteriormente, cabe la pena aclarar varios aspectos, en primer lugar, los niveles que se plantean en la aplicación no influyen con la consecución de las actividades pero si puede contener varias de ellas, es decir, en cualquiera de los niveles es posible encontrar varias de las actividades que se han detallado anteriormente sin que se tenga un orden específico entre ellas.

Aunque es posible cometer infracciones es preciso aclarar que la aplicación está pensada y desarrollada con el fin de emular las dinámicas de tránsito propias de la vida real, es por ello que dentro de los diferentes niveles pueden encontrarse situaciones como pasarse los semáforos en rojo, realizar los giros en carriles indebidos, hacer cruce de carril en línea continua, pararse sobre las intersecciones, embestir otros vehículos, transitar en contravía, subirse a los

andenes, entre otras; sin embargo, uno de los objetivos de la aplicación es evidenciar infracciones, que si bien son muy comunes, la mayoría de conductores comete regularmente es por ello que se determina que el usuario que ejecute la aplicación lo haga con la mayor seriedad teniendo en cuenta estas situaciones que se presentan y que por simple uso de razón deben sortearse de la mejor manera.

6.2 REFERENCIA DE NIVELES Y ACTIVIDADES.

La tabla 5 permite hacer una representación de los niveles de juego, las posibles infracciones que se pueden presentar y las actividades que se encuentran dentro de cada uno.

Tabla 5. Resumen de la descripción de actividades.

NIVEL	INFRACCIONES POSIBLES	ACTIVIDADES
1	<ul style="list-style-type: none"> - Pasarse los semáforos en rojo. - Girar en los carriles indebidos. - Transitar en contravía. - Embestir otros vehículos. - Intentar subirse a los andenes. - Tomar una ruta diferente a la sugerida. - Girar por esquinas prohibidas. 	<p>El objetivo de este nivel es llegar de un punto a otro sin cometer infracciones. El auto debe realizar un giro a la izquierda, por ello debe ubicarse en el carril correcto para girar, además debe estar pendiente del semáforo que da paso para poder girar.</p>
2	<ul style="list-style-type: none"> - Pasarse los semáforos en rojo. - Intentar subirse a los andenes. - Arrollar o ser arrollado por otros vehículos.- - Tomar una ruta diferente a la sugerida - Girar en los carriles indebidos. - Transitar en contravía. - Embestir otros vehículos. 	<p>En este nivel el usuario deberá transitar de un lugar a otro cruzando por dos intersecciones para luego girar a mano izquierda, para lo cual el propósito es que mientras el usuario avanza por el escenario lo recomendable es ir buscando el carril propicio para el giro.</p>
3	<ul style="list-style-type: none"> - Girar en los carriles indebidos. - Transitar en contravía. - Embestir otros vehículos. - Pasarse los semáforos en rojo. - Arrollar o ser arrollado por otros vehículos.- - Tomar una ruta diferente a la sugerida.- - Intentar subirse a los andenes. - Girar por esquinas prohibidas. - Mantenerse estacionado en intersecciones por tiempos prolongados. 	<p>El propósito en este reto, al igual que los anteriores, es transitar de un punto a otro, para ello se presenta una situación particular en donde el usuario puede optar por dos decisiones, como el camino más corto para llegar al punto de destino sugiere un giro donde se encuentra una congestión vehicular el auto puede esperar a que se descongestione la vía para poder hacer el giro en el carril adecuado o la otra solución es realizar el recorrido alternativo que propone el mapa, esta última con el fin de evitar más atasco y permanecer tiempos prolongados sobre la intersección.</p>

4	<ul style="list-style-type: none"> - Mantenerse estacionado en intersecciones por tiempos prolongados. -Girar en los carriles indebidos. - Pasarse los semáforos en rojo. - Arrollar o ser arrollado por otros vehículos. -Transitar en contravía - Tomar una ruta diferente a la sugerida.- Intentar subirse a los andenes. - Girar por esquinas prohibidas. - Embestir otros vehículos. 	<p>El objetivo del cuarto nivel es que el usuario llegue a punto designado, para ello se ha diseñado una situación en la que se presenta un atasco en la vía justo en la señalización de intersección, para ello se propones dos alternativas, por un lado que el usuario aguarde a que se descongestione la vía y así evitar que el auto se estacione sobre la intersección y se generen infracciones o accidentes, o por otro lado, tomar el recorrido alternativo propuesto en el inicio de la actividad.</p>
5	<ul style="list-style-type: none"> -Girar en los carriles indebidos. - Mantenerse estacionado en intersecciones por tiempos prolongados -Transitar en contravía - Tomar una ruta diferente a la sugerida. - Intentar subirse a los andenes. -Pasarse los semáforos en rojo. - Arrollar o ser arrollado por otros vehículos. - Girar por esquinas prohibidas. - Embestir otros vehículos. - Esquivar carros sin tener precauciones de direccionales para prevenir los carros que vienen detrás. 	<p>En este nivel se quiere hacer énfasis en las complicaciones que pueden presentarse al parquear en vías principales. Para esto el usuario debe transitar por el escenario evitando los autos que están parqueados en medio de la vía y teniendo precaución con aquellos vehículos que también esquivan delante o los que vienen atrás para no sufrir accidentes o cometer infracciones.</p>

Capítulo 7.

7 RESULTADOS Y OBSERVACIONES

7.1 PRUEBAS DE FUNCIONAMIENTO.

Para las pruebas de funcionamiento se delimitó fueran de forma cerrada exponiendo la aplicación a una población limitada y no superior a 5 personas con el fin de determinar si cumple los objetivos planteados desde la propuesta de anteproyecto. La intención de realizar estas pruebas es examinar el funcionamiento de la aplicación en temas referentes a las mecánicas de interacción y las correspondientes respuestas a acciones que ejecuta el usuario a la misma, evadiendo criterios personales relacionados a temas como el arte, la animación, la música, las actividades, entre otros, puesto que para profundización en estos aspectos, junto con un estudio más concienzudo de la población a la cual estará dirigida la implementación de la herramienta, demandaría más tiempo del estipulado para la realización de este proyecto.

7.2 PRESENTACIÓN DE RESULTADOS.

Conforme a los objetivos planteados para el desarrollo de esta aplicación a continuación se presentan los resultados obtenidos:

1. Programar las mecánicas de interacción del usuario con la aplicación a través de comandos del teclado y movimientos de la cabeza que controlen el vehículo.

Luego de consultar diversas formas de interacción del usuario con la aplicación y la forma de visualización de la misma, pudiendo utilizar diversas herramientas para ello como controles externos o Gamepads, Joysticks Kinect, cascos de realidad virtual entre otros, se optó por el uso de un teclado convencional de computador para la interacción con la

aplicación gracias a fácil manejo y posibilidad de obtención, además porque no se requería de accesorios externos al computador que requirieran sincronización con los demás dispositivos, lo que disminuye los percances que se pueden presentar a la hora de la ejecución. En cuanto a la visualización, se eligió el uso de gafas de realidad virtual ya que en el mercado pueden conseguirse fácilmente de diversos tipos y costes lo que permite que sean asequibles a cualquier usuario.

2. Diseñar las situaciones de movilidad para que el conductor este propenso a cometer infracciones.

Las situaciones que se tuvieron en cuenta para la representación de las dinámicas que permitían el desarrollo y la ejecución de la aplicación se diseñaron en base a contextos y ambientes reales en donde se pudieran observar, de esta manera, para cada una de las actividades se plantearon condiciones que exigían al usuario a realizar determinadas maniobras que terminaban en la consecución de una o varias infracciones y que generaban el reinicio del nivel. De la misma forma en cada uno de los niveles se propusieron una serie de alternativas para que el usuario, en forma de ayuda, pudiera lograr su cometido de la mejor manera posible.

3. Diseñar un modelo de ciudad que permita representar las situaciones de movilidad y posibles infracciones de tránsito.

El estudio de las actividades planteadas en la delimitación posibilitó la proyección y el desarrollo de un escenario que respondiera a las dinámicas creadas, conjuntamente, la consecución de dichas situaciones correspondió a la ubicación de los diferentes elementos diseñados, modelados, implementados y programados en cada uno de los niveles que contiene la aplicación, lo que permitió simular un ambiente real adecuado y conveniente para este objetivo.

4. Programar los comportamientos de autonomía de los vehículos que se mueven dentro de la ciudad.

Una de las metas más relevantes y que requirió un trabajo más concienzudo en el desarrollo de la aplicación fue la consecución de las dinámicas de autonomía de los vehículos y los objetos u elementos que componían el modelo de ciudad. La implementación del modelo POO permitió la ejecución de éste objetivo facilitando la forma de programación de todos y cada uno de los componentes que hacían parte de los diferentes niveles concebidos en la herramienta. De la misma manera, el uso de Scripts para cada uno de los elementos disminuyó el índice de error que se pudiera generar al realizar pocos Scripts más elaborados y con un desarrollo más robusto. Al finalizar la aplicación y luego de realizar varias pruebas de funcionamiento se determina que el escenario junto con los objetos que los componen aún contiene pequeñas fallas de programación también conocidos como *bugs* o *glitches* que se dan al no obtener resultados esperados, sin embargo, éstas no interfieren en ningún momento ni bajo ninguna circunstancia en la ejecución, rendimiento u estabilidad de la aplicación, por el contrario son fallas habituales que se dan en el desarrollo de este tipo de herramientas y que son insumo para trabajos futuros.

Capítulo 8.

8 CONCLUSIONES Y TRABAJO FUTURO.

1. Se desarrolló una aplicación basada en realidad virtual inmersiva denominada “TUTRÁNSITO” que permite evidenciar infracciones típicas que se comenten diariamente al conducir y que desencadenan diferentes situaciones que se reflejan en los agentes que intervienen. La herramienta cumple con los objetivos propuestos logrando describir y evidenciar algunos comportamientos de personas que diariamente conducen algún tipo de automotor y que no acatan las normas y señales de tránsito estipuladas por la ley.
2. Al ser la seguridad vial una problemática latente en la actualidad, tanto que sea una asignatura obligatoria dentro de las instituciones educativas, hace de esta aplicación una herramienta necesaria y atrayente para los agentes que intervienen en dichas dinámicas, que posibilita espacios de meditación, reflexión y concientización de las consecuencias de sus actos. La culminación de la aplicación “TUTRÁNSITO” demuestra la viabilidad de narración de problemáticas de índole social a partir de la implementación de herramientas alternativas e innovadoras como ésta.
3. De acuerdo al contexto y al de desarrollo de la aplicación es viable el desarrollo de aplicaciones para licenciados en electrónica gracias a los diferentes énfasis y la transversalidad de asignaturas que se registran en el pensum de la licenciatura.
4. La selección del motor de videojuegos Unity 3D así como su uso permitió el desarrollo de la aplicación en un lapso de tiempo corto sin demeritar su calidad y eficacia en la ejecución gracias a la variedad de herramientas que éste brinda en relación al diseño, programación, visualización e integración de

los elementos. De igual manera, la elección del lenguaje de programación C# o CSharp fue la más conveniente gracias a la existencia de grandes cantidades de documentación, videotutoriales y a la gran cantidad de desarrolladores que día tras día crece lo que posibilita la solución ágil de diversos inconvenientes que se presentaron durante el desarrollo de la aplicación y en cuanto a programación se refiere.

5. La implementación del modelo POO facilitó el desarrollo de la aplicación en términos de la programación permitiendo crear scripts específicos para cada una de las funciones de los elementos lo que posibilita un aumento en el índice de eficiencia de la aplicación y corrigiendo la gran mayoría de errores que pudieran presentarse si se codificara en otros lenguajes de programación.
6. Debido al tiempo estipulado para la entrega de la aplicación no fue posible realizar un estudio de la población a la cual podría ser dirigida así como los resultados que arrojaría en dicha población, además para poder observar la aceptación de este tipo de herramientas en diferentes contextos, sin embargo, La aplicación "TUTRÁNSITO" está dispuesta a cambios y modificaciones a futuro en todas sus etapas que contribuyan a un mejoramiento de la misma en cuanto a desempeño, eficacia, rendimiento, etc.
7. Para trabajos futuros en cuanto a la aplicación, es posible pensar en nuevas actividades que permitan incluir otras situaciones diferentes a las contempladas en éste proyecto, de la misma manera, el trabajo queda abierto para aquellos estudiantes y personas en general que quieran realizar una implementación realizando un estudio más concienzudo de la población a la cual quiere ser dirigida la herramienta y que permita generar estudios potencialmente más elaborados sobre este tipo de problemáticas.

REFERENCIAS

REFERENCIAS DE LIBROS Y ARTÍCULOS.

[1] Jorge, A.N.; Godoy Del Sol, H.; Ortis, M., "Caracterización de la mortalidad por accidentes del tránsito con participación de ciclos: un problema sociomédico," en *MediSur*. 2010, vol. 8, no. 4, pp. 57-62. Jul.-Ago. 2010. ISSN 1727-897X.

[2] Pérez, F.J., "Presente y Futuro de la Tecnología de la Realidad Virtual," en *Creatividad y Sociedad*. 2011, vol., no.16, pp.1-39. Mar. 2011.

[3] Schultheis, M; Mourant, R, "Virtual Reality and Driving: The Road to Better Assessment for Cognitively Impaired Populations," in *Presence*, vol.10, no.4, pp.431-439, Aug. 2001 doi: 10.1162/1054746011470271

[4] Fouladinejad, N.; Fouladinejad, N.; Abd Jalil, M.K.; Taib, J.M., "Modeling virtual driving environment for a driving simulator," in *Control System, Computing and Engineering (ICCSCE), 2011 IEEE International Conference on* , vol., no., pp.27-32, 25-27 Nov. 2011

[5] Nehaoua, L.; Mohellebi, H.; Amouri, A.; Arioui, H.; Espie, S.; Kheddar, A., "Design and Control of a Small-Clearance Driving Simulator," in *Vehicular Technology, IEEE Transactions on* , vol.57, no.2, pp.736-746, March 2008 doi: 10.1109/TVT.2007.905336

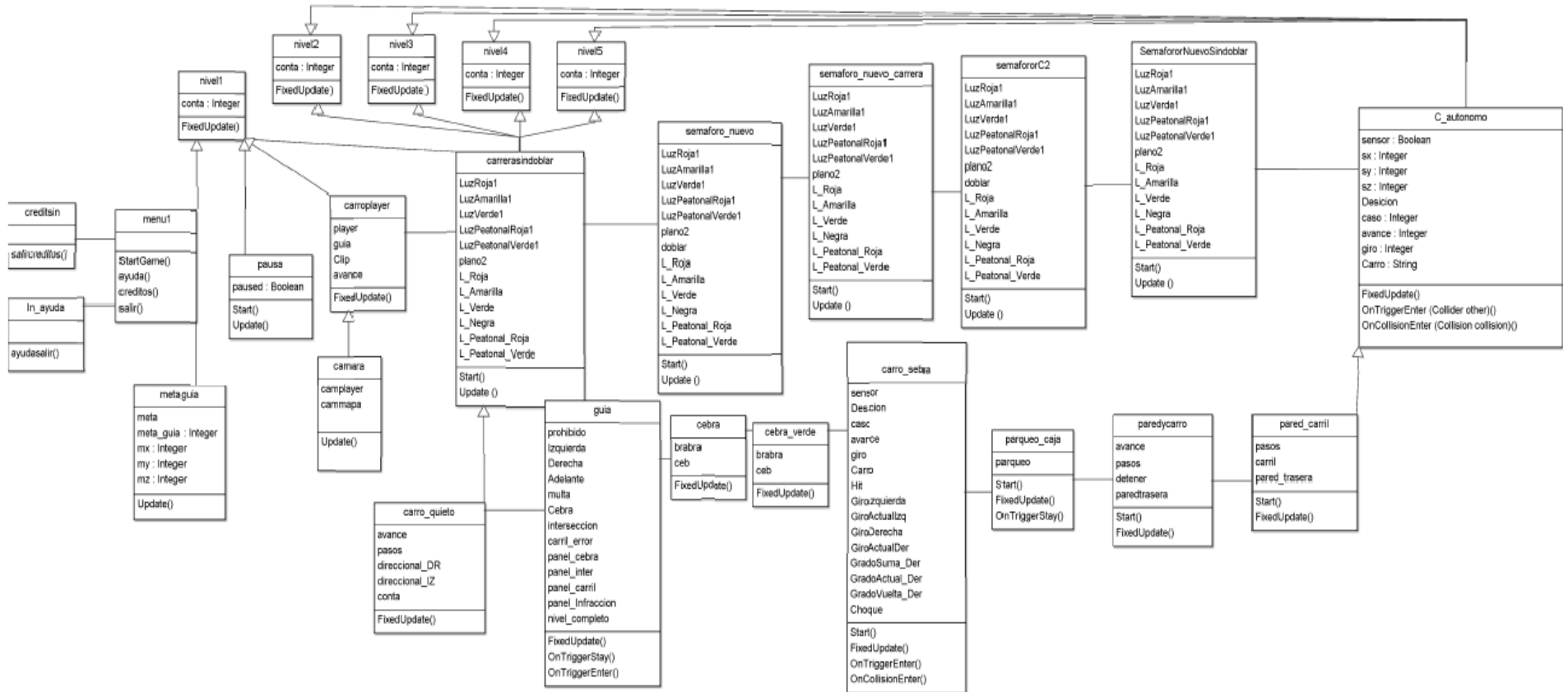
[6] Sekizawa, S.; Inagaki, S.; Suzuki, T.; Hayakawa, S.; Tsuchida, N.; Tsuda, T.; Fujinami, H., "Modeling and Recognition of Driving Behavior Based on Stochastic Switched ARX Model," in *Intelligent Transportation Systems, IEEE Transactions on* , vol.8, no.4, pp.593-606, Dec. 2007 doi: 10.1109/TITS.2007.903441

[7] Tateyama, Y.; Ogi, T.; Nishimura, H.; Kitamura, N.; Yashiro, H., "Development of Immersive Virtual Driving Environment Using OpenCABIN Library," in *Advanced Information Networking and Applications Workshops, 2009. WAINA '09. International Conference on*, vol., no., pp.550-553, 26-29 May 2009 doi: 10.1109/WAINA.2009.208

- [8] Han Jiaqi; Santoso, M.; Lee Byoung Gook, "Immersive Driving Car Simulation for Children Using Natural User Interface Controller," in *Ubiquitous Virtual Reality (ISUVR), 2013 International Symposium on*, vol., no., pp.23-25, 10-13 July 2013 doi: 10.1109/ISUVR.2013.17
- [9] Gregoriades, A.; Florides, C.; Lesta, V.P.; Pampaka, M., "Driver Behaviour Analysis through Simulation," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, vol., no., pp.3681-3686, 13-16 Oct. 2013 doi: 10.1109/SMC.2013.627
- [10] Pérez, F.J., "Presente y Futuro de la Tecnología de la Realidad Virtual," en *Creatividad y Sociedad*. 2011, vol., no.16, pp.1-39. Mar. 2011.
- [11] JONES, H. (1995): *Virtual reality applications*. Londres, Academic Press.
- [12] Gálvez Mozo, A. (2004) *Posicionamientos y puestas en pantalla. Un análisis de la producción de sociabilidad en los entornos virtuales*. Barcelona: UAB.
- [13] urkle, S. (1997) *La vida en la pantalla. La construcción de la identidad en la era de internet*. Barcelona: Paidós.
- [14] Bakar, N.A.A.; Zulkifli, A.N.; Mohamed, N.F.F., "The use of multimedia, Augmented Reality (AR) and Virtual Environment (VE) in enhancing children's understanding of road safety," in *Open Systems (ICOS), 2011 IEEE Conference on*, vol., no., pp.149-154, 25-28 Sept. 2011 doi: 10.1109/ICOS.2011.6079288
- [15] mamura, T.; Ogi, T.; Lun, E.T.C.; Zhong Zhang; Miyake, T., "Trial Study of Traffic Safety Education for High School Students Using Driving Simulator," in *Systems, Man, and Cybernetics (SMC), 2013 IEEE International Conference on*, vol., no., pp.4606-4611, 13-16 Oct. 2013 doi: 10.1109/SMC.2013.784
- [16] «Manifiesto por el Desarrollo Ágil de Software». [En línea]. Disponible en: <http://www.agilemanifesto.org/iso/es/>. [Accedido: 14 ene 2016].

- [17] Laínez, J (2015). Desarrollo de Software Ágil: *Extremme Programming y Scrum*. Estados Unidos: Createspace Independent Publishing Platform. Recuperado de <https://books.google.com.co/books?id=TxRpCwAAQBAJ&pg=PA104&dq=desarrollo+iterativo+e+incremental+libros&hl=es-419&sa=X&ved=0ahUKEwj5crp5svKAhWBWSYKHQ14C54Q6AEIGjAA#v=onepage&q=desarrollo%20iterativo%20e%20incremental%20libros&f=false>
- [18] L. Rising y N. S. Janoff, «The Scrum software development process for small teams», IEEE Software, vol. 17, n.o 4, pp. 26-32, ago. 2000.
- [19] H. Takeuchi y I. Nonaka, «The new new product development game», Harvard Business Review, 1986.
- [20] Singh, M., "U-SCRUM: An Agile Methodology for Promoting Usability," in *Agile, 2008. AGILE '08. Conference*, vol., no., pp.555-560, 4-8 Aug. 2008 doi: 10.1109/Agile.2008.33
- [21] Club de Desarrolladores Stuff. Metodología SCRUM. [en línea]. 2008. [Consultado 15 de Enero de 2016]. Disponible en: <http://www.clubdesarrolladores.com/articulos/mostrar/63-metodologia-scrum/2>

ANEXOS



Anexo 1. Diagrama de clases.