



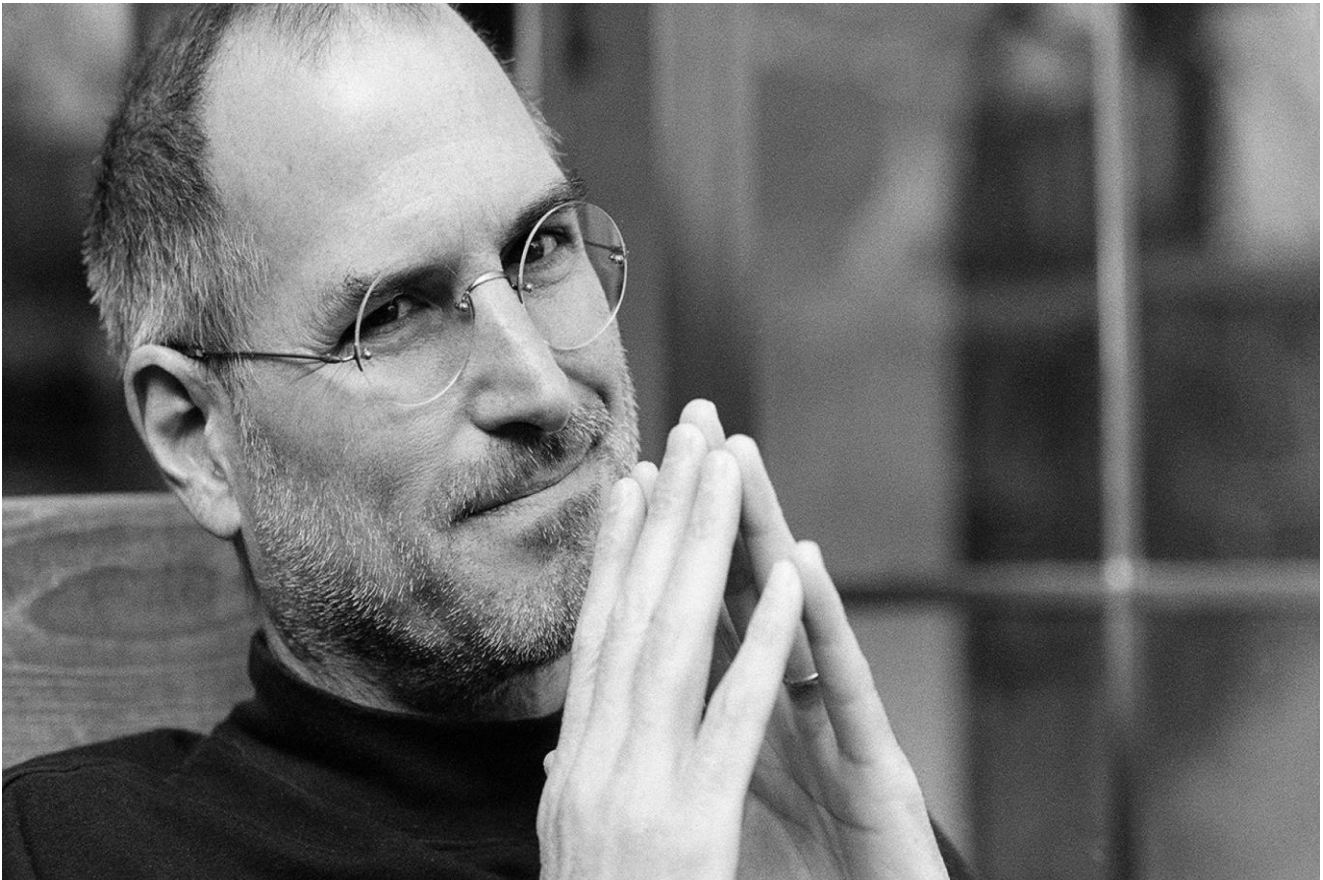
**Universidad de las Ciencias Informáticas
Centro de Entornos Interactivos 3D (VERTEX)
Facultad 5**

KeylaxyEyes: videojuego serio para el tratamiento de niños con deficiencias del campo visual.

***Trabajo de diploma para optar por el título de Ingeniero en
Ciencias Informáticas.***

Autora: Grethel Naranjo Rondón
Tutor: MSc. Omar Correa Madrigal

“Año 57 de la Revolución”
La Habana, Cuba
Junio 2015



“Tienes que encontrar lo que amas. Y esto es tan verdad para tu trabajo como lo es para tus seres queridos. Tu trabajo va a llenar una gran parte de tu vida y la única manera de estar verdaderamente satisfecho es hacer lo que creas que es un gran trabajo. Y la única manera de hacer un gran trabajo es amar lo que haces. Si no lo has encontrado aún, sigue buscando. No pares.”

Steve Jobs.

Declaración de autoría

Declaro ser el único autor de la presente tesis y reconozco a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Grethel Naranjo Rondón

Autora

MSc. Omar Correa Madrigal

Tutor

Datos de contacto

Tutor: MSc. Omar Correa Madrigal.

Institución: Universidad de las Ciencias Informáticas (UCI).

Título: Máster en Informática Aplicada.

Síntesis del Tutor:

- Director del Centro de Entornos Interactivos 3D VERTEX.
- 10 años de experiencia.

E-mail: ocorrea@uci.cu.

Dedicatoria

A Luis Enrique Castillo Villar, mi lucero.

A Keyla, mi tata.

Agradecimientos

A mi familia, que siempre han estado ahí para apoyarme a pesar de las decisiones erróneas que he tomado, en especial a mi mamá y mi abuela Idalia.

A Roberto Javier García Membrado por sembrar en mí la semillita de superación.

A Rey, mi amigo, mi hermano, mi apoyo, mi base, gracias por existir.

A Ernesto, gracias por soportarme todos estos años, por ser mi talismán, por convertirme en una persona mejor.

A la familia de Ernesto, gracias por tratarme con tanto cariño.

A mi tutor Omar por su paciencia, sus ejercicios y las horas que tomó de su doctorado para dedicárselas a mi tesis, dígan lo que dígan él es el mejor tutor en toda la UCI.

A la pequeña familia que logré tener aquí cuando creí que no podría soportar 5 años becada, mama Liz, papa Julio y tata Yamila

A dos muy buenos amigos Pedro (el fiscal) y Fredy (el europeo).

A todos los profes que han puesto su empeño directa e indirectamente para que yo sea ingeniera el día de hoy, Luis Eduardo, Leo, Susej, Zaida, Zory, Parra, Millet, Yaíma, Kire, Alain...

A mis tutores Yusmila, Gelson y Jaime por su confianza.

A mis colegas de estudio Valido y Victor.

A mi colega de cuando yo era programadora del COJ, Alcolea.

A Juan y su familia por el apoyo y por las buenas vibras han enviado para mí.

Al 5503 por acogerme con tanta comprensión, en especial a mimí, nené, Yersin, Yandry, tita, kiki, luísi, Arisney, Nesty, Marín, Esmaykel, Liset.

A todas la personas que he conocido aquí y con las que he podido contar en mis mejores y peores momentos y que siempre llevaré guardadas en mi corazón de ingeniera.

Resumen

La amplitud del campo visual es una variable oftalmológica muy importante en la detección de problemas visuales. Para su medición se utiliza la campimetría, que puede ser cinética, estática o por confrontación. En Cuba generalmente se utiliza la campimetría por confrontación para tratar a personas menores de edad. Esta prueba se complejiza al aplicarla a niños debido a que requiere mucho tiempo para su ejecución. Además de que, por razones propias de la edad y por no poseer elementos lo suficientemente motivadores, los pacientes no cooperan con el proceso. Esto, unido a la subjetividad del método produce una baja cuantificación de las mediciones del campo visual. Con el objetivo de ofrecer una alternativa de rehabilitación del campo visual en pacientes menores de edad se presenta el videojuego serio de navegador KeylaxyEyes. En el mismo se vincula la lógica del juego y el tratamiento sobre la base de la implementación de la campimetría estática y cinética. Para guiar el correcto ciclo de vida de la solución se escoge la metodología Huddle soportando el desarrollo con el motor de videojuegos Phaser y las tecnologías HTML5, JQuery y JSON. Intercambios con especialistas en rehabilitación de la escuela especial Abel Santamaría y del Instituto cubano de Oftalmología Ramón Pando Ferrer validaron como positiva la solución generada.

Palabras clave: campo visual, navegador, rehabilitación, videojuego.

Índice de contenidos:

INTRODUCCIÓN.....	1
CAPÍTULO 1: Fundamentación teórica	- 4 -
1.1 Definiciones fundamentales	- 4 -
Campo visual	- 4 -
Rehabilitación.....	- 5 -
Videojuegos Serios	- 6 -
1.2 Clasificación de videojuegos serios	- 10 -
1.3 Videojuegos serios de navegador.....	- 13 -
1.4 Elementos del diseño de un videojuego.....	- 14 -
1.5 Motores de videojuegos de navegador.....	- 14 -
1.6 Lenguajes y frameworks.....	- 16 -
1.7 Entorno de desarrollo integrado	- 18 -
1.8 Metodología de desarrollo	- 19 -
1.9 Consideraciones parciales	- 26 -
CAPÍTULO 2: Características y diseño del sistema.	- 27 -
2.1 Historia del videojuego	- 27 -
2.2 Descripción de la solución	- 27 -
2.3 Guión Técnico	- 29 -
2.4 Consideraciones parciales	- 38 -
CAPÍTULO 3: Implementación y prueba del sistema	- 39 -
3.1 Funcionalidades del sistema.....	- 39 -
Requisitos funcionales	- 39 -
Requisitos no funcionales del sistema.....	- 42 -
3.2 Implementación	- 43 -

Funciones de los scripts.....	- 44 -
3.3 Pruebas	- 45 -
Pruebas de compatibilidad	- 46 -
Pruebas de aceptación.....	- 48 -
3.4 Consideraciones parciales	- 48 -
Conclusiones generales	- 49 -
Recomendaciones.....	- 50 -
Referencias Bibliográficas.....	- 51 -
Glosario de términos.....	- 54 -

Índice de ilustraciones

Ilustración 1: Amplitud del campo visual.....	- 4 -
Ilustración 2: Videojuego EyeOk	- 8 -
Ilustración 3: Videojuego CitySalud.....	- 9 -
Ilustración 4: Videojuego Meteorix.....	- 9 -
Ilustración 5: Videojuego Montaña Rusa.....	- 10 -
Ilustración 6: Fases de la metodología Huddle	- 20 -
Ilustración 7: Implementación de campimetría estática.....	- 28 -
Ilustración 8: Implementación de campimetría estática.....	- 28 -
Ilustración 9: Secuencia de scripts.....	- 44 -
Ilustración 10: Vista en el navegador Mozilla Firefox	- 46 -
Ilustración 11: Vista en el navegador SeaMonkey	- 47 -
Ilustración 12: Vista en el navegador Opera	- 47 -
Ilustración 13: Vista en el navegador Google Chrome	- 48 -

Índice de tablas

Tabla 1: Comparación de motores de videojuegos - 16 -

Tabla 2: Comparación de IDEs - 18 -

Tabla 3 Plantilla del documento de diseño - 20 -

Tabla 4: Estructura del JSON..... - 29 -

Tabla 5: Documento de diseño - 30 -

Tabla 6: Representación de funcionalidades..... - 39 -

Tabla 7: RF1 - Activar modo pantalla completa..... - 39 -

Tabla 8: RF2 - Crear jugador - 40 -

Tabla 9: RF3- Crear item de campimetría estática - 40 -

Tabla 10: RF4- Visualizar item de campimetría estática..... - 40 -

Tabla 11: RF5- Crear item de campimetría cinética - 40 -

Tabla 12: RF6- Visualizar item de campimetría cinética..... - 41 -

Tabla 13: RF7- Controlar jugador - 41 -

Tabla 14: RF8- Recoger item..... - 41 -

Tabla 15: RF9-Aumentar la complejidad por niveles - 41 -

Tabla 16: RF10- Cambiar nivel - 42 -

Tabla 17: RF11- Salvar en fichero el perfil de usuario..... - 42 -

Tabla 18: RF12- Controlar audio..... - 42 -

Tabla 19: Requisitos no funcionales..... - 43 -

Tabla 20: Script Inicial..... - 44 -

Tabla 21 Script Carga..... - 44 -

Tabla 22 Script Menú principal - 45 -

Tabla 23 Script Nuevo héroe - 45 -

Tabla 24 Script Nivel 1 - 45 -

Tabla 25: Script Nivel 2 - 45 -

INTRODUCCIÓN

El juego es una actividad fundamental para el desarrollo humano, si bien ha sido sinónimo de diversión y entretenimiento, lleva inconsciente e implícitamente el aprendizaje. Es considerado como una actividad voluntaria, primordialmente social, en la que se comparten experiencias y se aprenden pautas de comportamiento, valores y cultura. Desde finales de la década de los 90 se comienza a examinar la utilidad de los juegos para nuevos propósitos; con la introducción de las tecnologías de la informática y las comunicaciones, el creciente interés por emplearlos con fines instructivos y su capacidad para contribuir a la formación se vieron ampliados con el desarrollo de los videojuegos.

Los videojuegos proveen a los jugadores de habilidades y destrezas. A través de sus diseños transmiten la cultura digital y desarrollan actitudes propias de personas a quienes les gusta aprender nuevas acciones y manejarse en mundos virtuales. Participar en ese ambiente invita a los videojugadores a relacionarse con la comunidad que se forma a través de las redes sociales del contexto de juego, añadiendo, por tanto, pautas de relación y convivencia. Además aprenden del reto, de acciones propias y de las que ejecutan los demás jugadores o la inteligencia artificial, en el caso de que se juegue en forma individual. Esas experiencias que proveen los videojuegos han sido aprovechados en ámbitos interesados en promover habilidades y estimular el desarrollo y adquisición de conocimientos, en el entrenamiento de profesionales de la milicia, los bomberos, la policía, áreas empresariales y en la medicina; destacando los videojuegos terapéuticos o de rehabilitación.

Cuba, en los últimos años ha potenciado el desarrollo de *software*, incursionando recientemente en el ámbito de la producción de sistemas de realidad virtual como videojuegos serios con fines terapéuticos para entornos de escritorio, siendo centro esencial de este desarrollo la Universidad de las Ciencias Informáticas (UCI) y dentro de la ella el Centro de Entornos Interactivos 3D (VERTEX) de la Facultad 5. VERTEX en los últimos años, de conjunto con el Instituto Cubano de Oftalmología “Ramón Pando Ferrer” (ICO), ha desarrollado soluciones informáticas para el tratamiento de distintos padecimientos oftalmológicos, obteniendo resultados satisfactorios en el área de la agudeza visual. Ejemplo de ello son los videojuegos Meteorix y Montaña rusa dinámica. Otra de las áreas que recientemente explora es el campo visual.

La amplitud del campo visual (CV) es una variable oftalmológica muy importante en la detección de problemas visuales ocasionados por daños externos o de forma congénita. Algunos de los padecimientos oftalmológicos que afectan el campo visual, como degeneración macular, cataratas,

glaucoma o lesiones oculares, que se identifican en edades tempranas pueden ser corregidos con un tratamiento adecuado.

De las pruebas de medición del campo visual, en Cuba, la más usada para tratar a personas menores de edad es la campimetría por confrontación. Esta prueba complejiza al ser aplicada a niños debido a que requiere mucho tiempo para su ejecución. Además de que por razones propias de la edad y por no poseer elementos lo suficientemente motivadores los pacientes no cooperan con el proceso. Esto, unido a que las aproximaciones se hacen a vista de los especialistas, propicia que se produzca una baja cuantificación de la información adquirida.

Teniendo en cuenta la situación antes descrita se plantea como problema de la investigación: ¿Cómo contribuir a la rehabilitación del campo visual haciendo uso de aplicaciones informáticas?

Para dar solución a dicho problema se sigue como objetivo: desarrollar un videojuego serio de navegador como alternativa de rehabilitación del campo visual de niños.

Como objeto de estudio se consultan los videojuegos para la rehabilitación de pacientes con deficiencias visuales.

El campo de acción se enmarca en los videojuegos para la rehabilitación de niños con deficiencias del campo visual.

Tareas a cumplir:

- Definición de las bases conceptuales referentes al tema de desarrollo de videojuegos serios de navegador con fines terapéuticos.
- Caracterización de los videojuegos serios de navegador con fines terapéuticos para identificar los principales elementos que se puedan incluir en la solución propuesta.
- Identificación y caracterización de las tecnologías y herramientas *web* que se utilizan para la creación de videojuegos de navegador.
- Selección de las herramientas y tecnologías para la creación del videojuego serio de navegador.
- Realización del levantamiento de requisitos funcionales y no funcionales.
- Implementación de un prototipo funcional de videojuego serio que brinde la solución al problema planteado.
- Ejecución de las pruebas para validar el cumplimiento de los requerimientos.
- Integración del videojuego a la plataforma para la gestión de videojuegos.

Para llegar a un resultado concreto de la investigación, se emplearon los siguientes métodos investigativos:

Métodos teóricos:

- **Análisis histórico-lógico:** utilizado para el estudio de trabajos similares e investigaciones que abordan el tema de la creación de videojuegos serios a nivel nacional e internacional que sirvan como punto de partida para el desarrollo de la solución en su conjunto.
- **Método analítico-sintético:** utilizado para analizar desde diferentes aristas los conceptos asociados a los videojuegos serios. La síntesis de la información recopilada sobre los distintos videojuegos serios, permitió describir sus características generales.
- **Método de la modelación:** utilizado para crear abstracciones con el objetivo de explicar la realidad, se utiliza para la modelación de los diagramas necesarios según la metodología seleccionada, que guían todo el proceso de desarrollo de la solución.

Método empírico:

- El método **observación** se utilizó con el objetivo de observar el funcionamiento de videojuegos ya existentes, para obtener un registro visual de todos los elementos que se pudieran mejorar en su lógica y que puedan formar parte de la solución.

CAPÍTULO 1: Fundamentación teórica

En el capítulo se expone el marco teórico de la investigación. Se analizan los conceptos fundamentales relacionados con el campo visual, la rehabilitación de la visión, los videojuegos serios terapéuticos, entre otros términos de interés. Además se identifica la metodología de desarrollo de *software* que guiará el ciclo de vida de la solución así como las tecnologías y herramientas necesarias para su correcta aplicación.

1.1 Definiciones fundamentales

Para una mayor comprensión de la utilización de los videojuegos serios en tareas de rehabilitación, se hace necesario el conocimiento de varios conceptos que a continuación se exponen.

Campo visual

La amplitud del campo visual responde a la porción del espacio que el ojo es capaz de ver sin mover o rotar los órganos de visión. En estado normal, el CV se extiende aproximadamente desde 60° hacia dentro de la nariz hasta 100° hacia afuera en cada ojo, y unos 60° por encima y 70° por debajo de la horizontal (CUBBIDGE et al., 2006). [Ilustración 1]

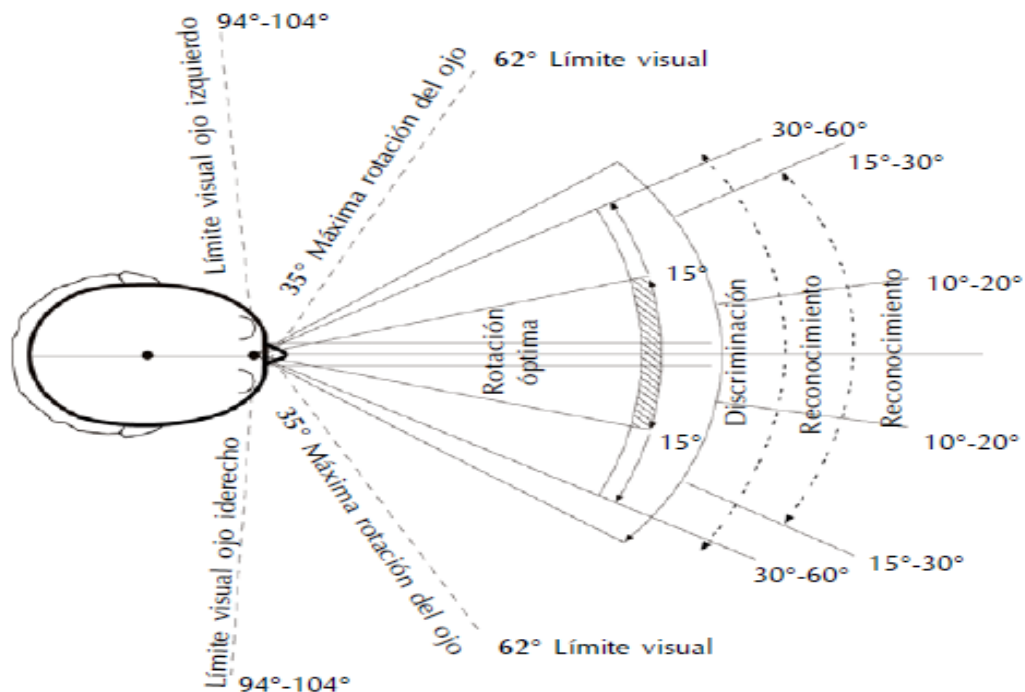


Ilustración 1: Amplitud del campo visual

Para la medición del CV se utiliza una técnica llamada campimetría, que es el estudio del campo visual y la detección de sus defectos (zonas de pérdida parcial o total de visión) mediante el empleo de un campímetro o perímetro (DE MEDINA Y BONO, 1944). Esta técnica puede ser aplicada de varias formas, cinética, estática o por confrontación.

Campimetría cinética: se manifiesta como puntos de luz que se mueven desde los límites del campo visual hacia dentro hasta que el observador puede verlos.

Campimetría estática: los puntos de luz se encienden aleatoriamente sobre una pantalla blanca, de manera que el paciente debe presionar un botón cuando lo visualiza.

Campimetría por confrontación: el paciente permanece sentado frente al examinador, cierra uno de sus ojos y mira fijamente al ojo contrario de su examinador, mientras se le pide que indique cuando ve y/o cuando deja de ver un objeto (bolígrafo, dedo del examinador, u otro) que se irá desplazando a lo largo del campo visual del ojo examinado.

Algunas de las enfermedades que afectan el campo visual, si son identificadas en edades tempranas pueden ser corregidas con un tratamiento adecuado. En ocasiones no es necesaria una operación quirúrgica si con tratamientos de rehabilitación puede evitarse el aumento de la afección.

Para la técnica de campimetría por confrontación es imprescindible la presencia física de un especialista, puesto que él es quien da las valoraciones de la prueba; mientras que las demás técnicas se realizan con un equipo llamado Octopus, que luego analiza las estadísticas y permite al especialista obtener una valoración. Por tanto las más factibles para utilizar en la solución son la campimetría cinética y la estática, ya que pueden ser implementadas, siguiendo el mismo principio del tratamiento, en otros medios informáticos.

Rehabilitación

Rafael Gonzales Mas en su artículo “Rehabilitación médica” describe dos definiciones de rehabilitación, una de apreciación puramente médica que se considera como el empleo de todas las armas médicas que puedan favorecer la recuperación. Estas medidas se emplean como un suplemento de los tratamientos médicos y quirúrgicos utilizados por el médico. Y la otra es la restauración del paciente hasta sus máximos límites posibles (físico, mental, social, vocacional y económico) (MAS, 1997).

La Dra. Solangel en su artículo “Medicina de rehabilitación” apunta que la rehabilitación es un proceso complejo que resulta de la aplicación integrada de muchos procedimientos. En este se logra que el

individuo recupere su estado funcional óptimo, tanto en el hogar como en la comunidad, en la medida que lo permita la utilización apropiada de todas sus capacidades residuales (TÁPANES, 2012).

Las Dras. Dora, Mayra y Susana en el libro “Rehabilitación óptica en la degeneración macular relativa a la edad” definen rehabilitación visual como un conjunto de procesos encaminados a obtener el máximo aprovechamiento visual de un paciente portador de baja visión (BARADA *et al.*, 2002).

Para los médicos, rehabilitación es sinónimo de curación. Consideran que los síntomas del enfermo serán vencidos y que este será capaz de volver a sus anteriores actividades, combinando las técnicas de rehabilitación física con un tratamiento psicológico implícito.

De acuerdo con los autores antes mencionados y prestando especial atención a Tápanes y Barada, se puede definir rehabilitación como: un proceso global y continuo de duración limitada. Dicho proceso presenta objetivos definidos que están encaminados a lograr niveles óptimos de independencia física en el paciente. Además permite a las personas con discapacidades adquirir habilidades funcionales así como un ajuste psicológico, social, vocacional y económico, necesario para llevar de forma libre e independiente sus propias vidas.

Con la aparición de las tecnologías de la informática y las comunicaciones surge un nuevo modo de rehabilitación. Esta nueva etapa aprovecha las potencialidades de las tecnologías y las pone a disposición de la sociedad. Ejemplo de ello son los videojuegos serios terapéuticos o de rehabilitación. Para analizar este concepto es necesario hacerlo por partes, conociendo ya el concepto de rehabilitación resta analizar los videojuegos serios.

Videojuegos Serios

El término juego serio (del inglés *serious games*) o juegos con fines terapéuticos fue definido en 1970 por Clark Abt incluso antes de la aparición de los primeros videojuegos. Abt habla principalmente de los juegos de mesa y de los juegos de cartas, pero proporciona una definición general que puede aplicarse con facilidad a los juegos de la era informática.

“Un juego es un contexto con reglas entre adversarios que intentan conseguir objetivos. Interesan los juegos serios porque tienen un propósito educativo explícito y cuidadosamente planeado, y porque no están pensados para ser jugados únicamente por diversión” (ABT, 1970).

El término juego serio se considera contradictorio, puesto que el vocablo "juego" representa diversión, alegría, fantasía y relajación, se concibe como una acción que se aleja de las cosas “serias” de la vida; mientras que el término “serios” alude a responsabilidad, sensatez, realidad y acciones con consecuencias a considerar (MARCANO, 2008). Mike Zyda en su artículo “*From Visual Simulation to*

Virtual Reality to Games” publicado en la revista *Computer* de la *IEEE Computer Society*, define juego serio como una prueba mental, de acuerdo con unas reglas específicas, que utiliza la diversión como modo de formación, con objetivos en el ámbito de la educación, sanidad, política pública y comunicación estratégica (ZYDA, 2005).

Beatriz Marcano describe los videojuegos como herramientas a tener en cuenta en la estimulación cognitivo-afectiva, ya que favorecen el aprendizaje, la autoestima, potencian la creatividad y las habilidades digitales, al mismo tiempo que generan motivación y entretenimiento. Suponen una modalidad de enseñanza que debe ser utilizada por la comunidad educativa, aprovechando la cantidad de elementos emocionales que integran, su estimulación sensorial y la posibilidad de inmersión a través de los ambientes virtuales en los que se desenvuelven (MARCANO, 2008).

Los videojuegos serios poseen atributos y técnicas particulares que ayudan a comprender mejor y más rápidamente materiales y procesos complejos. Aumentan el compromiso de los entrenados en su formación, por la motivación que sienten de querer lograr el éxito en el juego. En cuanto al incremento de habilidades y destrezas destacan el desarrollo de la coordinación ojo-mano, mayor agudeza visual, rapidez de reacción, capacidad de atención a múltiples estímulos, facilidad para relacionarse con otros, alta motivación al logro, mayor tolerancia a la frustración, capacidad para tomar riesgos, resolver problemas y tomar decisiones (MARCANO, 2008).

Mark y Bernand afirman en su libro “Introducción a los videojuegos” que estos representan un nuevo arte, tan adecuado a la era digital como lo fueron los anteriores medios. Facilitan el acceso a nuevas experiencias estéticas y convierten la pantalla del ordenador en un reino de experimentación e innovación ampliamente accesible (WOLF y PERRON, 2005).

El poder de enseñanza que tienen los juegos y, por extensión, los que se basan en las nuevas tecnologías, como los videojuegos, es algo que nadie pone en duda. Algunos equipos de investigación han querido ir más allá y explorar el posible potencial terapéutico del videojuego para analizar la posibilidad de su empleo en grupos con necesidades especiales. Así, se desarrolla una nueva generación de juegos, denominados, juegos serios (INFOCOP, 2009).

De acuerdo con los autores antes mencionados atendiendo principalmente a Zyda y Abt, se puede definir videojuego serio como una actividad recreativa–formativa que involucra a una o varias personas. La interacción entre ellas debe ser por medio de un dispositivo tecnológico; dígase PC (computadora personal), consola, dispositivo móvil u otra. Dicha actividad requiere seguir reglas específicas y tomar decisiones hasta lograr un objetivo, que más allá de vencer un adversario se centra en la adquisición de conocimientos o en la rehabilitación.

Ejemplos de Videojuegos Serios

EyeOK: videojuego info-educativo sobre salud visual, tiene por finalidades el entretenimiento del usuario, el aprendizaje de buenas prácticas sobre salud visual y la realización de *tests* útiles tanto para el profesional como para el educador. Sus actividades permiten la detección y posibilitan la prevención de problemas visuales mientras que enseña buenas prácticas de salud visual. Aunque los jugadores serán en un principio la población infantil y juvenil, el videojuego va dirigido a padres, educadores y centros de formación. También está pensado para que los profesionales de la visión tengan disponibles contenidos digitales para la realización de *tests* ópticos (SERRANO, 2010).

[Ilustración 2]



Ilustración 2: Videojuego EyeOk

City Salud: trata la educación para la salud desde múltiples puntos de vista y permite aprender valores sobre alimentación saludable, actividad física, higiene, descanso. Previene hábitos nocivos como el consumo de tabaco y alcohol. Los contenidos educativos, que los participantes pueden descargarse en forma de fichas, coinciden con el temario de Conocimiento del Medio y Educación Física a la vez que se potencia el manejo de las nuevas tecnologías para menores de 12 años. Para completar el juego, los participantes tendrán que recorrer diferentes escenarios y resolver pruebas recopilando información relacionada con la salud (SERRANO, 2010). [Ilustración 3]



Ilustración 3: Videojuego CitySalud

Meteorix: Es un videojuego de acción en primera persona, de corte galáctico. Fusiona su lógica con la de un tratamiento, de manera que el paciente no se percata de que está siendo tratado. Este equilibrio constituye uno de sus principales aportes. También es capaz de personalizar los contenidos del juego avanzando a mayores niveles de complejidad, según sea el nivel de respuesta del niño (habilidad óculo-manual). La aplicación ha sido de gran utilidad, pues con sesiones diarias de 30 minutos se le brinda al paciente ejercicios de fijación central con parche de una manera entretenida. Luego de 12 sesiones ya se pueden lograr incrementos de la visión desde 1 hasta 3 niveles, según el *test* de agudeza visual *Snellen* (ARIAS-DÍAZ *et al.*, 2013). [Ilustración 4]



Ilustración 4: Videojuego Meteorix

Montaña Rusa Dinámica: videojuego 3D¹ para la rehabilitación de la agudeza visual, usando el tiempo de reacción y la orientación del campo visual como factores de presión, para pacientes

¹ tres dimensiones: alto, ancho y profundidad

jóvenes. El videojuego muestra las vías de una montaña rusa hecha de azúcar y hormigas que intentarán comérsela. El niño cuando observa las hormigas debe tratar de atraparlas haciendo uso del *mouse* (RAVELO, 2014). [Ilustración 5]



Ilustración 5: Videojuego Montaña Rusa

El análisis de estos ejemplos permitió obtener un registro visual de sus funcionalidades y características distintivas. Los dos primeros tratan temas de salud en pos de orientar a los niños, padres y profesores hacia una vida saludable. Emplean la tecnología Flash para su desarrollo, solo pueden ser jugados en línea y no tienen como objetivo realizar un tratamiento específico. Mientras que los dos restantes sirven como medio de rehabilitación de la agudeza visual y proveen una serie de variables de salida que permiten a especialistas identificar la evolución de sus pacientes. Además utilizan tecnologías de escritorio en su desarrollo por lo que solo pueden ser jugados en la PC donde se instalan y esta debe cumplir con ciertos requerimientos tecnológicos para una correcta ejecución de la aplicación.

1.2 Clasificación de videojuegos serios

Los videojuegos serios pueden adquirir varias clasificaciones atendiendo a diferentes características propias de cada uno. Estas características se recogen en tres grandes grupos: las relacionadas con el contenido tratado en el videojuego (ALVAREZ y RAMPNOUX, 2007); las relacionadas con la lógica del videojuego (GAVALDÀ *et al.*, 2011) y las relacionadas con el entorno de ejecución del videojuego (VALLEJO, 2014). Estas clasificaciones no son excluyentes entre si y se detallan a continuación.

Clasificación atendiendo al contenido tratado en el videojuego

Advergaming: (del inglés *advertising* y *game*, es decir, publicidad y juego), es la práctica de utilizar videojuegos para publicitar una marca, producto, organización o idea.

Edutainment: término que resulta de la unión de (*education* y *entertainment*, o sea, educación y entretenimiento), se aplica a los programas que enseñan mediante recursos lúdicos².

Edumarket Games: término que resulta de la unión de *education* (educación) y *marketing*, combina varios aspectos como los propios del *advergaming* y del *edutainment* u otros relacionados con la prensa y la persuasión. Un ejemplo es FoodForce, un juego con objetivos en el ámbito de las noticias y la persuasión.

Simulation Games: videojuegos que se emplean para adquirir o ejercitar distintas habilidades o para enseñar comportamientos eficaces en el contexto de situaciones o condiciones simuladas. En la práctica, son más utilizados los simuladores de conducción de vehículos por ejemplo FlightGear, los simuladores de gestión de compañías como TransportTycoon y los simuladores sobre negocios en general, que ayudan a desarrollar el pensamiento estratégico y enseñan a los usuarios los principios de la micro y macroeconomía además de la administración de empresas, una muestra de ello es Virtonomics.

Games for health: (en español juegos para la salud), diseñados como terapia psicológica, para el entrenamiento cognitivo o la rehabilitación física. Se utilizan como estrategia adicional a la terapia habitual que se utiliza para tratar algún tipo de patología.

Alternate reality game: videojuegos serios de realidad alternativa, están basados en el mundo real que, utilizando recursos audiovisuales, desarrollan una historia que se verá constantemente afectada por la intervención de sus participantes. Los objetivos se centran en favorecer las actitudes de superación personal, aprovechar la dinámica de juego para intensificar los procesos de aprendizaje, fomentar los entornos colaborativos y la comunicación, para solucionar los problemas y acertijos planteados. Ejemplo World without oil donde los videojugadores son los responsables de gestionar el problema que la sed desenfrenada de petróleo representa para la economía, el clima y la calidad de vida, además de dar soluciones trabajando de forma colaborativa y con creatividad.

² La ludificación o gamificación es la utilización de técnicas y dinámicas propias de los juegos y el ocio en actividades no recreativas.

Clasificación atendiendo a la lógica del videojuego

Acción: requieren que el jugador emplee sus reflejos, puntería y habilidad, a menudo en un contexto de combate o de superación de obstáculos y peligros. Dentro de este amplio género se encuadran varios subgéneros de gran popularidad, como los juegos de lucha y los de plataformas, entre otros.

Estrategia: se caracterizan por la necesidad de manipular a un numeroso grupo de personajes, objetos o datos, utilizando la inteligencia y la planificación, para lograr los objetivos. Aunque la mayoría de estos juegos son fundamentalmente de temática bélica, los hay también de estrategia económica, empresarial o social. Dos grandes subgéneros son los juegos de estrategia en tiempo real (también llamados «RTS», siglas del inglés de *real-time strategy*), y los que se ejecutan por turnos («TBS», siglas del inglés *turn based strategy*).

Simulación: este género se caracteriza por recrear situaciones o actividades del mundo real, dejando al jugador tomar el control de lo que ocurre. En ocasiones la simulación pretende un alto grado de verosimilitud, lo que le otorga una componente didáctica. Los tipos de simulación más populares son los de manejo de vehículos (pilotar un coche, un avión, un tren), los de construcción (construir una ciudad, un parque de atracciones o un imperio), o los de vida (dirigir la vida de una persona o un animal virtual).

Deporte: los videojuegos de deportes son aquellos que simulan deportes del mundo real. Entre ellos se encuentran golf, tenis, fútbol, hockey, juegos olímpicos, entre otros. La mecánica del juego es la misma que en el deporte original, aunque a veces incorpora elementos adicionales.

Aventura: son videojuegos en los que el protagonista debe avanzar en la trama interactuando con diversos personajes y objetos. El jugador encarna a un protagonista que por lo general debe resolver incógnitas y rompecabezas con objetos diversos.

Sandbox: aquellos en los cuales se comienza prácticamente desde cero, creando prácticamente todo lo necesario para avanzar y transformar un mundo virtual propio. Son videojuegos no lineales, porque no tienen una línea de juego apenas definida, el orden de las acciones permite la mayor libertad. Eso no impide que puedan definirse unos objetivos para guiar al jugador, quien puede optar por ignorarlos. Una de las características esenciales de un juego *sandbox* es que al jugador se le dan muy pocas limitaciones, y este es capaz de alterar y/o modificar el entorno del juego.

Agilidad mental: los juegos de lógica o de agilidad mental, también llamados de puzzles o rompecabezas, ponen a prueba la inteligencia del jugador para la resolución de problemas, que pueden ser de índole matemático, espacial o lógico. Algunos se enfocan en la rapidez mental del

jugador (juegos de lógica-acción), otros en la capacidad de razonamiento (juegos de física, o de acertijos), otros en la capacidad de observación (juegos de objetos ocultos).

Clasificación atendiendo al entorno de ejecución

Para escritorio: Videojuegos en los que se utilizaron tecnologías de escritorio para su desarrollo y que solo pueden ser ejecutados por medio de una PC.

Para consola: Videojuegos que para su ejecución necesitan de un dispositivo como un X-Box u otro tipo de consola, y que pueden ser visualizados en una computadora o en un televisor.

Para móvil: Videojuegos diseñados para ser ejecutados en plataformas móviles dígame *tablet*, celulares.

En línea: Videojuegos jugados vía internet independientemente de la plataforma. Puede tratarse de videojuegos multijugador, en los que se juega con otras personas o videojuegos de navegador que se descargan desde la *web* y se ejecutan en el navegador.

Web o de navegador: Videojuegos desarrollados usando tecnologías *web*, y que solo pueden ser visualizados a través de un navegador *web*.

Consideración del estudio de clasificaciones de videojuegos serios

Atendiendo a lo explicado anteriormente y teniendo en cuenta el contenido tratado; KeylaxyEyes será un videojuego para la salud. Actualmente en las escuelas el tratamiento se desarrolla mientras el niño juega libremente, por lo que es conveniente, de acuerdo a la lógica del videojuego, que sea de tipo aventura. Además dada la tecnología a emplear para su desarrollo y posterior ejecución podrá ser clasificado como un videojuego de navegador.

1.3 Videojuegos serios de navegador

Las técnicas de realidad virtual y videojuegos para la salud han sido empleadas en la enseñanza, el tratamiento y la rehabilitación de distintos padecimientos, beneficiándose de estos avances tanto doctores como pacientes. Dentro de los videojuegos para la salud se encuentran los videojuegos de navegador que son una modalidad desarrollada en los últimos años con la evolución de las tecnologías. Esta combinación ha obtenido resultados satisfactorios en el tratamiento de las funciones visuales del paciente (SERRANO, 2010).

Los videojuegos de navegador brindan la posibilidad de que el paciente se conecte de manera remota desde cualquier lugar sin la necesidad de tener un dispositivo de altas prestaciones, permitiéndoles llevar el tratamiento al hogar. El usuario no necesita realizar instalaciones extra para jugar. Los

archivos se ejecutan en el navegador ya que están basados exclusivamente en las tecnologías usadas por este, estableciendo la gran diferencia con los videojuegos de escritorio. Estas características hacen posible que la solución se pueda integrar fácilmente a una plataforma de gestión de videojuegos o si el usuario así lo desea llevarlo a su casa, es decir, utilizarlo de manera portable.

1.4 Elementos del diseño de un videojuego

El diseño de un videojuego tiene un número considerable de elementos a tener en cuenta (ROUSE, 2010). Los más significativos a los efectos del presente trabajo son:

Sistema de reglas: define el modo de actuar de los jugadores en el ámbito del videojuego.

Objetivo o reto: tiene una relación directa con el fin de ganar el juego, de acuerdo a su cumplimiento el jugador pierde o gana.

Sistema de recompensas: establece las formas de retroalimentación visual o sonora que el juego tiene para estimular al jugador cuando ha logrado vencer los retos definidos.

Argumento narrativo: se refiere a la historia donde quedan definidos los retos y escenarios del juego. Su dimensión está dada por el tipo de juego y por lo general se recoge en el guión de contenidos del videojuego.

A diferencia de los juegos para el ocio, la lógica de diseño de los videojuegos para la rehabilitación implica ajustes en los elementos expuestos anteriormente. Por ejemplo en los referentes para el tratamiento de la visión; no tienen asociados un argumento narrativo complejo ya que los tiempos por sesión oscilan sobre los 30 minutos (WADDINGHAM *et al.*, 2006).

El reto debe someter a prueba la capacidad del paciente sin llevarlo al límite, esto quiere decir que necesita de un mecanismo regulador que varíe los saltos de complejidad de paciente a paciente según establecen los protocolos de tratamiento. Por otra parte en la rehabilitación se trabaja sobre mecanismos de retroalimentación que evitan frustrar al jugador a pesar de que haya fallado el reto; un manejo adecuado de los fallos resulta fundamental para lograr una rehabilitación exitosa (BURKE *et al.*, 2010).

1.5 Motores de videojuegos de navegador

Un motor de videojuego es un término que hace referencia a una serie de rutinas de programación que permiten el diseño, la creación y la representación de un videojuego. La funcionalidad básica de un motor es proveer al videojuego de un motor de renderizado para los gráficos 2D (dos dimensiones:

alto y ancho) y 3D (tres dimensiones: alto, ancho y profundidad), motor físico o detector de colisiones, sonidos, animaciones, inteligencia artificial y un escenario gráfico (GREGORY, 2009).

Existen varias herramientas que pueden ser usadas como motor de videojuego de navegador, a continuación se detallan características de algunas de ellas. Tomando en consideración características como visualización basada en *sprites*, manejo de leyes físicas, tratamiento de sonidos y licencia. Serán evaluadas de bien (B), regular (R) en dependencia de las potencialidades de cada característica.

Impact: es una de las bibliotecas más populares para crear videojuegos usando JavaScript. Esto se debe a su facilidad de uso y al conjunto de herramientas que pone a disposición del programador. Maneja *sprites*, animaciones, mapas con patrones, colisiones, sonidos y provee un editor de niveles llamado Weltmeister, que permite crear casi cualquier objeto en 2D. Además cuenta con un sistema de *plugins* con el que se pueden extender aún más sus funcionalidades. Pero trabaja bajo un esquema de código cerrado y la licencia tiene un costo de US\$ 99 (ANDREARRS, 2014).

Gamejs: está inspirado en una antigua biblioteca para crear juegos en Python³ llamada PyGame; su sintaxis y la facilidad de uso de PyGame ha inspirado a los creadores de Gamejs para crear algo similar pero esta vez para JavaScript. Permite manejar *sprites*, animaciones, sonidos, mapas de patrones y colisiones pero además posee métodos muy útiles para búsqueda de rutas (necesario para la inteligencia artificial) y comunicaciones vía HTTP. Es además *OpenSource* (ANDREARRS, 2014).

MelonJS: es una sencilla pero poderosa biblioteca JavaScript para desarrollar juegos 2D. Aún está en desarrollo pero se puede utilizar para crear buenos juegos. Es *OpenSource*, compatible con Google Chrome, Safari, Firefox, Opera e incluso Internet Explorer en versiones igual o superiores a la 9. Posee mecanismos básicos de física y colisión. Integra el popular formato de mapas *tiled*, lo que permite diseñar fácilmente niveles usando el editor de mapas, así el programador puede centrarse en las características propias del juego. Este motor 2D basado en *sprites*, también cuenta con soporte a múltiples canales de audio, matemática básica de vectores y efectos de transición (ANDREARRS, 2014).

Phaser: es uno de los motores de juegos para JavaScript más recientes. Está diseñado para que los juegos se puedan ejecutar tanto en ordenadores como en dispositivos móviles. No sólo soporta canvas sino también WebGL y puede pasarse de uno a otro automáticamente según la compatibilidad

³ Lenguaje de programación.

del navegador; lo que le da un punto de ventaja en cuanto al renderizado y la velocidad de respuesta. Además contribuye a una mejor experiencia de usuario. Maneja física, colisiones, animaciones, sistema de partículas, mapas de patrones, sonidos y permite escalar el juego para que se ajuste a la resolución de cualquier dispositivo sin alterar la relación de aspecto. Phaser es *OpenSource*, se puede descargar y contribuir con su código fuente desde Github (ANDREARRS, 2014).

Tabla 1: Comparación de motores de videojuegos

	Impact	GameJS	MelonJs	Phaser
Gestión de interfaces	B	B	B	B
Motor de física	B	B	R	B
Manejo de sonido	B	R	B	B
OpenSource	No	Si	Si	Si

Luego de una comparación de las potencialidades de cada uno de los motores de videojuego de navegador analizados, se concluye que: Impact demuestra tener grandes potencialidades pero su licencia no es libre, GameJS carece de funcionalidades para el tratamiento de sonidos, MelonJS atrae con sus prestaciones pero está aún en desarrollo. Por otra parte está Phaser una potente herramienta que además de tener licencia *OpenSource* tiene todos los elementos necesarios para el desarrollo de videojuegos 2D avanzados, dígame motor de física, manejo de colisiones, tratamiento de mapas, visualización basada en *sprites*, sonidos. Además permite escalar el videojuego sin alterar la relación de aspecto por lo que es el motor de videojuegos de navegador escogido para el desarrollo de KeylaxyEyes.

1.6 Lenguajes y frameworks

A continuación se describen los lenguajes que serán empleados en el desarrollo de la solución. Además del *framework* que será utilizado como vía de optimización del lenguaje de programación JavaScript. También se caracteriza la tecnología que permitirá lograr la persistencia de los datos generados.

HTML5: HTML siglas de *Hypertext Markup Language* (lenguaje de marcado de hipertexto) es el lenguaje de marcado para las páginas *web*. Representa el lenguaje de la publicación de la *World*

Wide Web (LAWSON y SHARP, 2011a). Es un lenguaje de marcación diseñado para crear páginas *web* y otros documentos que sean posibles visualizar en el navegador *web*, siendo esencial para la creación de páginas y mostrar el contenido en ellas. HTML5 es la actualización de HTML, versión que trae nuevas facilidades para los desarrolladores y que serán utilizadas para la creación de *KeylaxyEyes*. Ejemplo de ello es la etiqueta `<canvas>`, es un elemento complejo que permite que se generen gráficos al hacer dibujos en su interior. Es definido como un rectángulo en la página, donde se puede utilizar JavaScript para la representación de gráficos. También determina un grupo de funciones (*canvas API*) para dibujar formas, crear gradientes y aplicar transformaciones (LAWSON y SHARP, 2011b).

JavaScript: JavaScript es un lenguaje de script multiplataforma. No es útil como un lenguaje independiente, más bien está diseñado para una fácil incrustación en otros productos y aplicaciones, tales como los navegadores *web*. Dentro de un entorno anfitrión, JavaScript puede ser conectado a los objetos de su entorno para proveer un control programable sobre estos (ZELDMAN, 2010). Para el desarrollo del videojuego será utilizado el *framework* JQuery para JavaScript.

JQuery implementa una serie de clases que permiten programar sin preocuparnos del navegador que utiliza el usuario, ya que funcionan de igual forma en todas las plataformas más habituales. Ofrece una infraestructura con la que se obtiene mayor facilidad para la creación de aplicaciones complejas del lado del cliente; por ejemplo ofrece ayuda en la creación de interfaces de usuario, efectos dinámicos. Además tiene licencia para uso en cualquier tipo de plataforma, personal o comercial (VAN LANCKER, 2012).

JSON

Del inglés (*JavaScript Object Notation*) es un formato ligero empleado para el intercambio de datos, siendo un subconjunto de la notación para objetos empleado en JavaScript (VEGA, 2013). Es sencillo y fácil de aprender. Mejora la representación estructural de los datos; requiere una baja codificación y procesamiento.

JSON soporta dos tipos de estructuras: los objetos que contienen una colección de pares llave-valor y los arreglos de valores. JSON no tiene espacios de nombres, cada objeto es un conjunto de claves independientes de cualquier otro objeto. Puede representar cualquier estructura de datos permitiendo añadir nuevos campos con total facilidad, característica que lo hace ser flexible. JSON es empleado con frecuencia en entornos *web* donde el flujo de los datos es alto entre el cliente y el servidor, también donde los tiempos de procesamiento de los datos son de vital importancia, porque optimiza los tiempos de respuesta debido a su leve peso (VEGA, 2013).

Además en el equipo de desarrollo *web* del departamento de construcción de componentes del centro VERTEX, se tomó como acuerdo que todos los productos del tipo videojuego de navegador que fueran desarrollados deben utilizar JSON para la persistencia de los datos.

1.7 Entorno de desarrollo integrado

Un Entorno de Desarrollo Integrado (IDE, por sus siglas del inglés *Integrated Development Environment*) es un entorno de programación que ha sido empaquetado como un programa de aplicación. Consiste en un editor de código, un compilador, un depurador y en ocasiones un constructor de interfaz gráfica de usuario (FERGARCIA, 2015). Los IDEs pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes.

Cada IDE tiene sus características únicas que deben proporcionar las funcionalidades básicas que un desarrollador espera si quiere desarrollar una aplicación *web*. Se consideraron las siguientes características en aras de hacer una comparación imparcial (DAMM, 2013). Se evaluarán de bien (B), regular (R), o no tiene (-) de acuerdo a las potencialidades de cada característica.

- Estructura en general de la interfaz de usuario, ya que afecta la facilidad el manejo del IDE.
- La creación de proyectos con intuitividad.
- Auto completado de HTML5 y JavaScript.
- Validación HTML5 y JavaScript.
- Navegación por código.

Tabla 2: Comparación de IDEs

Criterio\IDE	Netbeans	WebStorm	Visual Studio Express for Web	Eclipse EE
Interfaz de usuario	B	B	R	B
Creación de proyectos	B	B	R	R
Auto completado HTML5	B	B	B	R
Auto completado JavaScript	B	B	B	R

Auto completado JQuery	B	B	-	-
Validación HTML	B	B	B	B
Validación JavaScript/JQuery	B	B	R	B
Navegación por código	B	B	B	B
Licencia	Pública	Privada	Pública	Pública

La tabla muestra que para el desarrollo de aplicaciones *web* el IDE idóneo es WebStorm, pero atendiendo a que en Cuba se lleva a cabo la política de migración a *software* libre se hace necesario utilizar NetBeans, que además de poseer una licencia GPL, posee las características necesarias para realización de un aplicación *web*.

1.8 Metodología de desarrollo

Un videojuego es considerado una aplicación informática, por lo que es necesario una metodología o proceso de desarrollo que guíe su ciclo de vida. No existen muchas metodologías creadas específicamente para videojuegos, por lo general la adaptación de metodologías ágiles como Programación Extrema (en inglés *eXtreme Programming*, XP) o SCRUM son las opciones más utilizada por los equipos de desarrollo pequeños, inmersos en la creación de videojuegos (URRUTIA *et al.*, 2010).

Para el desarrollo de la presente solución se escoge la metodología “Huddle” la cual surge con el propósito de dirigir el ciclo de vida de un videojuego. Para su creación fueron tomadas como base elementos de varias metodologías entre ellas *WaterfallProcess* y SCRUM. Huddle es un proceso específico para desarrollo de videojuegos con las siguientes características: ágil, iterativo, incremental y evolutivo, óptimo para equipos pequeños multidisciplinarios (URRUTIA *et al.*, 2010).

Esta metodología al ser creada específicamente para guiar el ciclo de vida de un videojuego provee todos los artefactos necesarios para su creación. Al ser ágil permite un desarrollo rápido y a la vez con un nivel avanzado de madurez. Ha sido probada en el centro Vertex, arrojando resultados satisfactorios en el desarrollo de los trabajos “Videojuego serio para la enseñanza del esqueleto de la cabeza humana” (GÓMEZ y OZETE, 2013) y “Videojuego Serio Montaña Rusa Dinámica para el tratamiento de la agudeza visual” (RAVELO, 2014). La metodología plantea tres momentos principales Preproducción, Producción y Postmortem las cuales se describen a continuación de forma general. [Ilustración 6]



Ilustración 6: Fases de la metodología Huddle

Etapa de Preproducción

Se caracteriza sobre todo por realizar la concepción de la idea del videojuego. Se definen los aspectos fundamentales que conformarán el videojuego a lo largo de su construcción. Elementos o tareas que se abarcan en esta etapa son la definición de género, historia, bocetos y jugabilidad. Esta fase tiene como salida una primera versión del documento de diseño el cual es la base fundamental para comenzar la fase de Producción (URRUTIA *et al.*, 2010). A continuación se detalla el contenido del documento de diseño.

Tabla 3 Plantilla del documento de diseño

Conceptos	
Título	El título del juego, debe ser un nombre que capte la atención del jugador y del lector del documento. A grandes rasgos, debe incluir el concepto del juego. El título debe ser algo memorable.
Diseñador	El nombre del estudio y/o del diseñador o diseñadores del documento.
Género	El género abarca, qué tipo de juego será. Simulación, FPS, Rol, etc.
Plataforma	Qué <i>hardware</i> se requiere para jugarlo. Computadora Personal, Xbox 360, PS3, etc.
Versión	La versión del documento. Debe ser un número y no una fecha.

Sinopsis de jugabilidad y contenido	En uno o dos párrafos, describir la esencia de jugar el juego. Incluir un poco del contenido que tendrá, historia, personajes, objetivo, etc.
Categoría	Comparar con uno o varios juegos existentes y enfatizar en las diferencias y características principales de este juego.
Licencia	Describir si el juego está basado en un libro o en una película. Si es original, se puede omitir este campo o describir el por qué puede convertirse en una franquicia.
Mecánica	Describir la jugabilidad y el control del juego. ¿Qué hace el jugador? ¿Qué usa para lograr sus objetivos?
Tecnología	Enlistar qué <i>hardware</i> y <i>software</i> se requiere para producir el juego. Desde lenguaje de programación hasta editor de sonidos.
Público	¿A quién va dirigido el juego? ¿Quién lo jugará? Se puede describir una demografía como niños o adolescentes, sin embargo, es más sencillo describir un tipo de jugador, ya sea casual, competitivo o veterano.
Visión general del juego	
Debe establecer la visión y el enfoque del juego que guiará al proyecto hasta el final del proceso. El resumen debe mencionar lo más interesante, las ventajas y lo original del juego. ¿Por qué las personas jugarían este juego? La estructura de los párrafos es similar a un ensayo, una introducción debe abarcar todos los aspectos importantes mientras que los párrafos subsecuentes deben detallar lo mencionado en la introducción. Al final, la conclusión debe dejar al lector entusiasmado y emocionado por jugar el juego.	
Mecánica del juego	
Esta sección esencialmente describe lo que el jugador puede hacer y cómo puede hacerlo. Describir las acciones del jugador, de preferencia en secuencia a cómo será en el juego.	
Cámara	Describir el tipo de cámara que se utilizará. Es decir, qué perspectiva tiene el jugador ante lo que está viendo en el juego, si es 3D o 2D, vista

	isométrica, en primera persona, etc.
Periféricos	¿Qué periféricos utilizará el jugador para lograr los objetivos mencionados? Incluir todos los que apliquen: teclado, <i>mouse</i> , <i>gamepad</i> , micrófono, etc.
Controles	Describir los botones y teclas que invoquen las acciones mencionadas en la sección de Mecánica del Juego.
Puntaje	Explicar de qué manera el juego se mantiene al tanto de los logros del jugador. Incluir también si existe una tabla de puntajes que compare los logros entre los jugadores, ya sea de manera local o en línea.
Guardar	Describir cómo el jugador guarda su progreso de los objetivos logrados y cómo puede continuar los objetivos pendientes. De igual manera, describir los dispositivos de almacenamiento que se usarán o si el juego tiene un sistema de contraseñas.
Estados del juego	
Un estado del juego se refiere al lugar donde se encuentra el jugador durante el juego, es decir, si el jugador está en el Menú Principal, está jugando un Juego Multijugador, está en el Menú de Pausa, etc. Los diagramas deben representar visualmente las relaciones entre los estados, si del Menú Principal se puede ir al Menú de Opciones, ¿Cómo lo hace? ¿Qué se ejecuta? ¿Qué interfaz muestra?	
Interfaces	
Las interfaces dan la pauta a la interactividad que tiene el jugador con el juego, en esta sección se debe describir la apariencia del juego, es decir, colores y temática. Es importante dejar una impresión visual en el jugador y debe estar relacionada con el concepto del juego.	
Nombre de la pantalla	El nombre de la pantalla, si es el Menú Principal u otra.
Descripción	¿Para qué sirve esta interfaz?
Estados del juego	Enlistar todos los estados de juego que invoquen esta pantalla así como

	también los estados que se puedan invocar en ella.
Imagen	Una imagen que muestre en concepto cómo se vería la pantalla.
Niveles	
<p>Los juegos comúnmente se dividen en niveles o en mapas secuenciales dentro de los cuales se debe cumplir con ciertos objetivos para progresar en el juego. Existen juegos en los cuales los niveles solo cambian a razón de la dificultad y los objetivos siguen siendo los mismos, de igual manera se deben describir esos cambios en esta sección.</p>	
Título del nivel	El nombre del nivel.
Encuentro	Describir si es el primer nivel, un tutorial o un <i>bonus</i> , en otras palabras, ¿Cuándo es que el jugador llega a este nivel?
Descripción	Una descripción detallada del nivel.
Objetivos	¿Qué debe hacer el jugador para terminar el nivel? Este campo también debe incluir si el jugador tiene que resolver ciertos acertijos o derrotar a cierto enemigo para progresar.
Progreso	Describir qué ocurre cuando el jugador termina el nivel.
Enemigos	Si el nivel tiene enemigos que el jugador debe enfrentar, estos se enlistan en este campo, de lo contrario este campo puede ser omitido.
<i>Items</i>	Enlistar los objetos que el jugador o los enemigos pueden usar y que aparecen en este nivel, este campo se puede omitir si no existen dichos objetos.
Personajes	Los personajes que aparecen en el nivel. Este campo puede ser omitido si no existen personajes en el juego.
Personajes	
<p>Los personajes principales y secundarios que aparecerán en el juego. Esta sección se puede omitir si</p>	

el juego no tiene personajes.	
Nombre	El nombre del personaje.
Descripción	Describir detalladamente el físico del personaje, si es humano o extraterrestre, su vestimenta, etc.
Imagen	Fotografía o dibujo conceptual del personaje.
Concepto	Describir la conducta y comportamiento, al igual que los motivos del personaje. Mencionar también si es el enemigo principal o el protagonista. El concepto también puede relatarse como una historia del personaje, detallando en las relaciones con otros personajes del juego.
Encuentro	¿Cuándo aparece este personaje en el juego?
Habilidades	Enlistar las habilidades del personaje.
Armas	Enlistar las armas del personaje.
Enemigo	
Los enemigos obstaculizan el progreso del jugador, pueden ser máquinas, otros personajes, monstruos, etc.	
Nombre	El nombre del enemigo.
Descripción	Describir detalladamente el físico del enemigo así como también su comportamiento.
Encuentro	¿Cuándo aparece este enemigo en el juego?
Imagen	Fotografía o dibujo conceptual del enemigo.
Habilidades	Enlistar las habilidades del enemigo.
<i>Items</i>	
Todos los objetos especiales que ayudan al jugador a realizar los objetivos y progresar en el juego se	

mencionan aquí.	
Logros	
Describir los varios logros o hitos que el jugador obtiene mientras progresa en el juego. Estos pueden otorgar medallas, personajes secretos o puntos extra.	
Música y sonidos	
La música y/o sonidos que se usarán en el juego, nombre, descripción junto con un número de referencia. Si es música de fondo, la referencia debe empezar con una 'M' seguida de un número en secuencia. Si es un efecto de sonido, empezar con 'S'.	
Miembros del equipo	
Información de las personas que trabajarán en el proyecto, incluye su nombre, el rol o roles que desempeñan y medios por los cuales se les puede contactar.	
Detalles de producción	
Antes de entrar a la etapa de Producción, se definen en el documento algunos detalles del proyecto.	
Fecha inicio	¿Cuándo empieza la etapa de Producción del proyecto?
Fecha terminación	¿Cuándo termina la etapa de Producción del proyecto?

Etapa de Producción

Es la fase que más tiempo requiere, donde los mayores esfuerzos se invierten en las etapas de diseño artístico y diseño técnico. Se desglosa en las etapas: diseño técnico, diseño artístico, sonido, interfaz, gráficos, diseño mecánico, motor del juego, implementación, y pruebas. Es en estas etapas donde al equipo de diseño del videojuego inicial se le incorporará el resto de la plantilla asociada a la producción del juego. Una vez concluida se tiene el *gold master*, que es una primera versión del producto final y una versión actualizada del documento de diseño (URRUTIA *et al.*, 2010).

Etapa Postmortem

El punto de partida de esta fase es el lanzamiento del videojuego. La etapa Postmortem genera un reporte cuyo propósito es escribir en detalle las actividades específicas que fueron más efectivas para el proyecto desde el inicio del proceso hasta la entrega del producto; de igual manera, describe las actividades que llegaron a perjudicar el desarrollo, junto con sugerencias para corregir dichos problemas, con la finalidad de no acarrearlos al siguiente proyecto (URRUTIA *et al.*, 2010).

1.9 Consideraciones parciales

Después de analizar algunos videojuegos serios de navegador, las tecnologías asociadas a estos, algunos motores de videojuegos de navegador, los lenguajes e IDEs así como la metodología recomendada para desarrollarlos, se concluye que:

- La utilización de videojuegos para la rehabilitación ha obtenido resultados satisfactorios tanto a nivel nacional como internacional. El hecho de llevar implícitamente el factor motivación, resuelve el principal problema existente en la rehabilitación de niños con problemas del campo visual.
- La combinación de tecnologías como HTML5, JavaScript y JSON permite desarrollar una aplicación de fácil integración y portabilidad; facilitando que el paciente pueda llevar el tratamiento al hogar. Además la metodología Huddle proporciona una guía viable para el desarrollo de videojuegos con un nivel avanzado de madurez.
- El motor de videojuegos de navegador, Phaser, provee los elementos necesarios para el desarrollo del videojuego deseado. Además el IDE seleccionado, Netbeans, es una potente herramienta que agiliza el proceso de desarrollo al facilitar la generación de proyectos HTML correctamente estructurados.

CAPÍTULO 2: Características y diseño del sistema.

En este capítulo se describe el prototipo de videojuego, que será la solución de la presente investigación. Se confecciona el guión técnico, artefacto rector del desarrollo en la metodología Huddle, encargado de representar la descripción detallada de cada aspecto que guía el desarrollo del videojuego.

2.1 Historia del videojuego

Prixy es un planeta donde cada 5 años se realiza la ceremonia del renacimiento, que consiste en que en la plaza se debe completar al rey del tridente, para lograrlo los sirvientes deben viajar a los distintos planetas y encontrar las piezas mágicas. Pero este año algo terrible ha pasado, el malvado SEÑOR M destruyó la nave real transportadora de objetos, se robó las piezas fundamentales y huyó. Por suerte mientras escapaba tropezó con un asteroide y las piezas fundamentales se esparcieron por el campo gravitatorio. Ahora necesitan a alguien muy valiente para que enfrente al SEÑOR M y salve al rey.

2.2 Descripción de la solución

EL videojuego implementa en un primer nivel la técnica de medición del campo visual llamada campimetría estática. En el videojuego los puntos de luz son representados por objetos que se visualizan en pantalla de manera aleatoria y en posiciones predefinidas en un intervalo de 3 segundos, tiempo de reacción medio para niños de 5-7 años (MARTINTO *et al.*, 2014). Las posiciones son calculadas teniendo en cuenta la resolución de pantalla con el objetivo de que queden uniformemente distribuidas, dividiéndola en una matriz de 11 filas y 19 columnas para un total de 209 posiciones. El niño, sin mover o rotar la cabeza, al verlos debe tratar de atraparlos haciendo uso del *mouse*. Al mismo tiempo se almacena en un arreglo la posición del objeto junto a un valor 1 (si logra cogerlo), 0 en caso contrario. Estos valores son guardados en un fichero JSON que es utilizado por la plataforma de gestión de videojuegos serios Medicando para graficar el mapa propuesto. El resultado de este proceso le permite al doctor obtener una imagen de las zonas de la visión central del paciente [Ilustración 1] interpretando cada grupo de 0 como una zona afectada. [Ilustración 7]



Ilustración 7: Implementación de campimetría estática

Los restantes 4 niveles utilizan la información generada del nivel 1 para distribuir una serie de objetos en la escena localizándolos específicamente en las áreas periféricas a las zonas identificadas como afectadas, con el objetivo de estimular las células presentes en esta zona. Las áreas a tratar son señaladas a partir de la variación de valores consecutivos de 0 a 1 y viceversa, en el mapa del CV.

Luego de identificadas las áreas que serán estimuladas se emplea la campimetría cinética como vía de rehabilitación. Para el videojuego se muestran estrellas que hacen un desplazamiento constante a través de las áreas señaladas. El área de desplazamiento en cada nivel disminuye, buscando una mayor aproximación al borde de la zona afectada. [Ilustración 8]

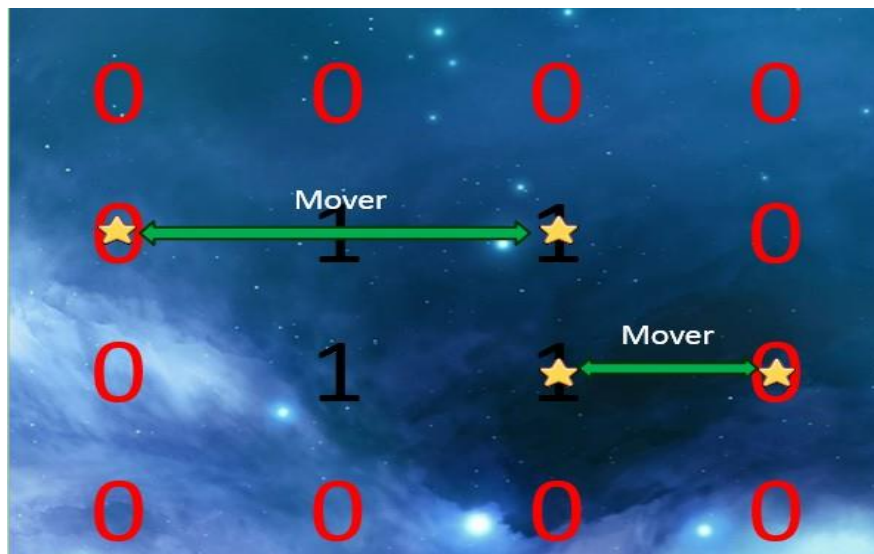


Ilustración 8: Implementación de campimetría estática

En cada nivel aumenta la complejidad, determinada por la cantidad de obstáculos a evadir y la cantidad de *item* a recoger para lograr el objetivo. Los obstáculos se definen como bombas que aparecen de manera aleatoria y que deben ser esquivadas porque disminuyen la puntuación. Y los *items* son definidos por estrellas y monedas.

El fichero JSON almacena toda la información que genera el videojuego dígase nombre del jugador, fecha de juego incluyendo día y hora, además de total de objetivos propuestos y total de objetivos cumplidos. También recoge la puntuación total alcanzada y el arreglo de puntos que permite trazar el mapa del campo visual. Toda esta información se estructura como muestra la siguiente tabla:

Tabla 4: Estructura del JSON

Clave	Valor
"name_player": " ernesto"	Nombre del jugador
"date": "2015-09-26 "	Fecha en que juega
"time_ini": "09:30 "	Hora de inicio de juego
"time_end": "11:00 "	Hora de fin de juego
"works": "266"	Cantidad total de objetivos
"hits": "178"	Cantidad de objetivos cumplidos
"score":5826	Puntuación total alcanzada
"map": [[[200, 348], 1],[[200, 520], 1],[[480, 621], 0],[[548, 762], 0],[[1011, 751], 0],[[1264, 751], 0],]	Mapa del campo visual

2.3 Guión Técnico


Es el artefacto más importante del proceso. El diseñador del juego asienta la idea a este documento con información detallada del proyecto, desde el título del juego, el género, una visión general y mecánica, aspectos de jugabilidad, modos de juego, plataforma, *software* que se utilizará. Estará sujeto a cambios a lo largo de toda la fase de diseño y será consultado en las restantes fases del

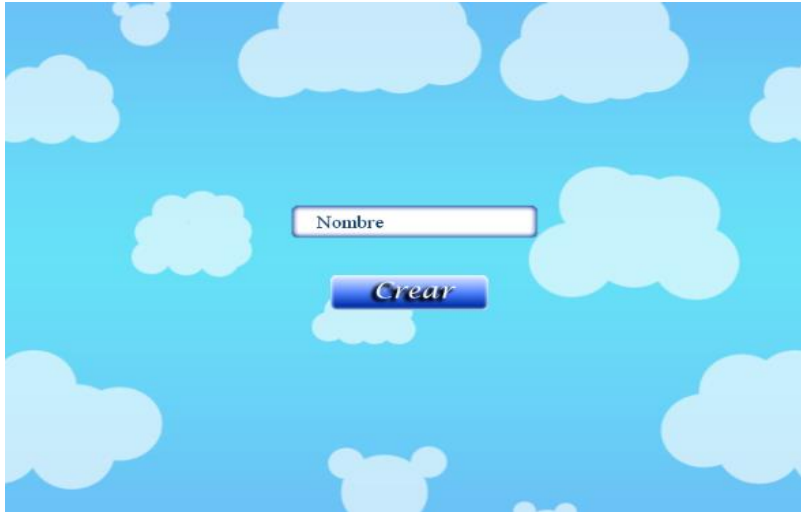
proceso de desarrollo del videojuego. Contiene las especificaciones necesarias para comenzar el proyecto (URRUTIA *et al.*, 2010).

Tabla 5: Documento de diseño

Conceptos	
Título	KeylaxyEyes.
Diseñador	Grethel Naranjo Rondón.
Género	Videojuego serio.
Plataforma	Navegador <i>web</i> .
Versión	1.0
Sinopsis de jugabilidad y contenido	El jugador deberá guiar la nave con el <i>mouse</i> para capturar o evadir los objetos de la escena. Cada uno de los niveles siguientes irá aumentando la complejidad determinada por la cantidad de obstáculos a evadir y también aparecerán más <i>items</i> que darán mayor puntuación.
Categoría	Aventura.
Licencia	Original.
Mecánica	El control del personaje es a través del <i>mouse</i> , click izquierdo para disparar el arma y para recoger los <i>items</i> solo debe pasar por encima de ellos.
Tecnología	IDE Netbeans 8.0, motor de videojuego Phaser, lenguajes HTML5, JavaScript.
Público	Niños con trastornos del campo visual.
Visión general del juego	
En cada nivel el jugador deberá alcanzar varios objetivos, dados por la recolección de <i>items</i> que aumentan la puntuación, evadir obstáculos o derrotar al enemigo, para así completar el reto del nivel, que estará dado por el elemento perteneciente al rey que debe ser rescatado en cada uno. Al	


<p>final será bonificado por su desempeño, mostrando la puntuación alcanzada y el objeto que define el cumplimiento del reto. En cada nivel aumenta la complejidad de los objetivos para así mantener al jugador motivado.</p>	
<p>Mecánica del juego</p>	
Cámara	2D, tercera persona.
Periféricos	<i>Mouse.</i>
Controles	<ul style="list-style-type: none"> -El jugador sigue la trayectoria del puntero. -Click izquierdo para disparar el arma. -Colisionar con el <i>item</i> para recogerlo. -Tecla espacio para acceder al menú pausa.
Puntaje	Cada <i>item</i> , así como los enemigos vencidos, le proporcionarán una puntuación acumulativa que se mostrará en el borde superior izquierdo de la pantalla de juego. Así mismo, cada vez que sea golpeado por un enemigo u obstáculo la puntuación es disminuida.
Guardar	Al final del juego se descarga un fichero JSON con la información generada a partir del desempeño del jugador. Este fichero es guardado localmente dentro del directorio json en la carpeta del videojuego.
<p>Estados del juego</p>	
<pre> graph LR Start(()) --> Boot[Boot] Boot --> Preloader[Preloader] Preloader --> MainMenu[MainMenu] MainMenu --> Level1[Level1] Level1 --> Level2[Level2] Level2 --> Level3[Level3] Level3 --> Level4[Level4] Level4 --> Level5[Level5] Level5 --> End(()) MainMenu <--> NewPlayer[NewPlayer] </pre>	
<p>Interfaces</p>	


Nombre de la pantalla	Menú principal.
Descripción	<p>Contiene las opciones:</p> <ul style="list-style-type: none"> -Jugar: inicia la partida. -Crear jugador: crea un nuevo usuario. -Música si/no: activa/desactiva la música de fondo.
Estado MainMenu	Aparece al inicio.
Imagen	
Nombre de la pantalla	Menú Nuevo Héroe.
Descripción	<p>Presenta un campo de entrada de datos en el que el jugador podrá introducir su nombre usando el teclado, además del botón crear que retorna al menú principal y actualiza el texto del saludo (cambia “pequeño” por el nombre introducido por el usuario, que para el ejemplo es “ernesto”).</p>
Estado NewPlayer	Es invocado desde el menú principal.

<p>Imagen</p>	
<p>Niveles</p>	
<p>Título del nivel</p>	<p>Armando la nave.</p>
<p>Encuentro</p>	<p>Este es el primer nivel.</p>
<p>Descripción</p>	<p>El jugador debe encontrar las piezas de la nave.</p>
<p>Objetivos</p>	<p>Obtener las posiciones tanto de los objetos que el niño logra ver y/o recoger así como los que le quedan ocultos. Estos puntos serán utilizados para trazar un mapa del campo visual, identificando las zonas afectadas.</p>
<p>Progreso</p>	<p>Se muestra una imagen de la nave y seguidamente la introducción al siguiente nivel.</p>
<p>Enemigos</p>	<p>-</p>
<p>Items</p>	<p>Diamantes (solo en el prototipo).</p>
<p>Personajes</p>	<p>Jugador.</p>
<p>Título del nivel</p>	<p>En busca de las botas.</p>
<p>Encuentro</p>	<p>Este es el segundo nivel, para llegar hasta aquí el jugador debe haber superado</p>

	el nivel 1.
Descripción	El jugador debe sobrevivir a los obstáculos en el nivel y recoger la mayor cantidad de <i>items</i> hasta que encuentre las botas del rey. Los <i>items</i> tipo estrella presentarán un movimiento cinético correspondiente a 4 posiciones entre la variación 0-1 anteriormente calculada desde el nivel 1.
Objetivos	Encontrar las botas del rey.
Progreso	Aparece un portal que será la salida al siguiente nivel. Se muestra una imagen de las botas del Rey y un botón que permite pasar al siguiente nivel.
Enemigos	Señor M.
<i>Item</i>	Estrellas.
Personajes	Jugador.
Título del nivel	En busca de la capa.
Encuentro	Este es el tercer nivel, para llegar hasta aquí el jugador debe haber superado los niveles anteriores.
Descripción	El jugador debe sobrevivir a los obstáculos en el nivel y recoger la mayor cantidad de <i>items</i> hasta que encuentre la capa del rey. Los <i>items</i> tipo estrella presentarán un movimiento cinético correspondiente a 3 posiciones entre la variación 0-1 anteriormente calculadas desde el nivel 1.
Objetivos	Encontrar la capa del rey.
Progreso	Aparece un portal que será la salida al siguiente nivel. Se muestra una imagen de la capa del rey y un botón que permite pasar al siguiente nivel.
Enemigos	Señor M.

<i>Items</i>	Bonos de puntos, estrellas, bonos de tiro.
Personajes	Jugador.
Título del nivel	En busca del cetro.
Encuentro	Este es el cuarto nivel, para llegar hasta aquí el jugador debe haber superado los niveles anteriores.
Descripción	El jugador debe sobrevivir a los obstáculos en el nivel y recoger la mayor cantidad de <i>items</i> hasta que encuentre el cetro del rey. Los <i>items</i> tipo estrella presentarán un movimiento cinético correspondiente a 2 posiciones entre la variación 0-1 anteriormente calculada desde el nivel 1.
Objetivos	Encontrar el cetro del rey.
Progreso	Aparece un portal que será la salida al siguiente nivel. Se muestra una imagen del cetro y un botón que permite pasar al siguiente nivel.
Enemigos	Súbditos del Señor M.
<i>Item</i>	Estrellas.
Personajes	Jugador.
Título del nivel	En busca de la corona.
Encuentro	Este es el quinto nivel, para llegar hasta aquí el jugador debe haber superado los niveles anteriores.
Descripción	El jugador debe sobrevivir a los obstáculos en el nivel y recoger la mayor cantidad de <i>items</i> hasta que encuentre la corona del rey. Los <i>items</i> tipo estrella presentarán un movimiento cinético correspondiente a una posición entre la

	variación 0-1 anteriormente calculada desde el nivel 1.
Objetivos	Encontrar la corona del rey.
Progreso	Aparece un portal que será el fin del juego. Se muestra una imagen del rey con las botas, la capa, la corona y el cetro.
Enemigos	Señor M.
Item	Estrellas.
Personajes	Jugador, rey.
Personajes	
Nombre	Lo escoge el usuario.
Descripción	El jugador es humano, pero la imagen que se muestra corresponde a una nave espacial con los colores de la bandera cubana donde la punta es de color rojo con una estrella blanca y hacia abajo salen las franjas azules y blancas. Esta nave es simétrica (forma semi-ovalada) tiene dos cañones en la parte inferior y se puede observar una escotilla de cristal que permite ver la cara del jugador.
Imagen	
Concepto	Protagoniza esta historia, viene desde la tierra para ayudar a los amigos de Prixy a salvar a su rey.
Encuentro	Desde el inicio.
Habilidades	Tomar <i>item</i> , disparar, volar.
Armas	Cañones.

Enemigo	
Nombre	Señor M.
Descripción	Es un extraterrestre dentro de un robot espacial.
Encuentro	Aparece una vez en cada nivel, eliminarlo representa el objetivo más importante en cada nivel, pues es el que finaliza el reto, o sea, permite adquirir el elemento del rey que se busca en cada nivel.
Imagen	
Habilidades	Disparar, llevarse <i>item</i> y volar.
<i>Items</i>	
Estrellas, monedas y diamantes.	
Logros	
<ol style="list-style-type: none"> 1- Recoger todos los <i>items</i> del nivel otorga la condición de cadete buscador. 2- Vencer al enemigo sin recibir ningún disparo otorga la condición de cadete veloz. 3- Pasar todos los niveles sin morir otorga la condición de comandante espacial. 	
Música y sonidos	
Música de fondo -----M0001.mp3 Alcanzar monedas ---S0001.wav Alcanzar estrellas ----S0002.wav Vencer al enemigo ---S0003.wav Fin del nivel -----S0004.wav	

Miembros del equipo	
Grethel Naranjo Rondón.	
Detalles de producción	
Fecha inicio	5-10-2014
Fecha terminación	15-6-2015

2.4 Consideraciones parciales

- La etapa de Preproducción, estableció un punto de partida para el correcto desarrollo de la solución; dándole contenido concreto al equipo de desarrollo, para la implementación del comportamiento de cada elemento que interviene en el videojuego.
- El pre cálculo de los puntos de representación del campo visual teniendo en cuenta la resolución de pantalla, permitió simular coherentemente la zona central el campo visual del paciente; este proceso trajo consigo que fueran identificadas las zonas afectadas con una mejor precisión.
- La utilización de un mapa de la visión central para guiar el uso de la campimetría cinética posibilita estimular las zonas del campo visual afectadas con una mejor precisión.

CAPÍTULO 3: Implementación y prueba del sistema

El desarrollo de este capítulo comienza con la extracción de las funcionalidades del videojuego propuesto como solución, los requerimientos necesarios para su correcta implementación. Posteriormente continúa con el análisis del diagrama de estados y cómo fue organizada su implementación, así como los resultados alcanzados durante las pruebas realizadas al *software*.

3.1 Funcionalidades del sistema

Las funcionalidades se han concretado en las diferentes reuniones establecidas según la metodología. Para facilitar el entendimiento de los elementos descritos se explica la estructura en la cual se presentan los requisitos. Los cuales serán representados por una tabla con el siguiente formato:

Tabla 6: Representación de funcionalidades

Identificación	Código que identifica de forma única los requisitos.
Título	Título descriptivo del requisito.
Descripción	Explicación del requisito especificado.

Requisitos funcionales

Tabla 7: RF1 - Activar modo pantalla completa

Identificación	RF1
Título	Activar modo pantalla completa.
Descripción	Al presionar el botón de jugar, el área de juego se ajusta a la pantalla de la PC (<i>fullscreen</i>).

Tabla 8: RF2 - Crear jugador

Identificación	RF2
Título	Crear jugador.
Descripción	El sistema debe permitir que el usuario pueda cambiar el nombre del jugador.

Tabla 9: RF3- Crear item de campimetría estática

Identificación	RF3
Título	Crear <i>item</i> de campimetría estática.
Descripción	Se generan un conjunto de puntos de acuerdo con la resolución de pantalla de forma que se distribuyan uniformemente en toda su área.

Tabla 10: RF4- Visualizar item de campimetría estática

Identificación	RF4
Título	Visualizar <i>item</i> de campimetría estática.
Descripción	Los objetos aparecen en pantalla de forma aleatoria en las posiciones anteriormente calculadas.

Tabla 11: RF5- Crear item de campimetría cinética

Identificación	RF5
Título	Crear <i>item</i> de campimetría cinética.
Descripción	Se obtienen pares de puntos en la variación de valores de 0-1 en el mapa del campo visual.

Tabla 12: RF6- Visualizar item de campimetría cinética

Identificación	RF6
Título	Visualizar <i>item</i> de campimetría cinética.
Descripción	Los objetos deben aparecer en pantalla de forma aleatoria con una trayectoria que se taza entre los pares de puntos anteriormente calculados.

Tabla 13: RF7- Controlar jugador

Identificación	RF7
Título	Controlar jugador.
Descripción	El sistema debe habilitar el movimiento del jugador con el <i>mouse</i> .

Tabla 14: RF8- Recoger item

Identificación	RF8
Título	Recoger <i>item</i> .
Descripción	El usuario puede recoger el <i>item</i> generado en la escena al pasar el jugador por encima de estos.

Tabla 15: RF9-Aumentar la complejidad por niveles

Identificación	RF9
Título	Aumentar la complejidad por niveles.
Descripción	La complejidad en cada nivel debe aumentar variando la cantidad de objetivos a alcanzar, la cantidad de obstáculos a evadir y disminuyendo la distancia de la trayectoria que deben recorrer los objetos definidos por la campimetría cinética.

Tabla 16: RF10- Cambiar nivel

Identificación	RF10
Título	Cambiar nivel.
Descripción	El usuario debe cambiar de nivel una vez vencidos los objetivos de este.

Tabla 17: RF11- Salvar en fichero el perfil de usuario

Identificación	RF11
Título	Salvar progreso de juego.
Descripción	El sistema debe almacenar en un fichero JSON los datos necesarios (mapa de campo visual y perfil de juego).

Tabla 18: RF12- Controlar audio

Identificación	RF12
Título	Controlar audio.
Descripción	Desde las opciones del menú principal el usuario podrá activar/desactivar la música de fondo.

Requisitos no funcionales del sistema

Los requisitos no funcionales especifican los criterios para la operación de un sistema en lugar de sus comportamientos. La siguiente tabla corresponde a los requisitos no funcionales del videojuego:

Tabla 19: Requisitos no funcionales

Identificación	Categoría
	Compatibilidad
RNF1	Estará optimizado para ejecutarse en los navegadores Mozilla Firefox y SeaMonkey directamente, pero podrá ser accedido desde cualquier otro que soporte HTML5 siempre que esté alojado en un servidor <i>web</i> .
	Portabilidad
RNF2	El usuario podrá ejecutar la aplicación en cualquier sistema operativo donde se ejecuten los navegadores con los que es compatible.
	Usabilidad
RNF3	El sistema poseerá un dominio de aplicación <i>web</i> .
RNF4	El videojuego poseerá una navegación sencilla, guiada por botones cuyos nombres son fáciles de asociar con la función que realizan.

3.2 Implementación

Para la implementación del videojuego se utilizó HTML5 como lenguaje primario, además de un conjunto de *scripts*. Estos *scripts* se clasifican atendiendo a la función que cumplen en el videojuego, dando comportamiento y funcionalidades a los objetos que conforman parte de los escenarios. A continuación se expondrán los *scripts* usados en la implementación del videojuego de cada estado del videojuego.

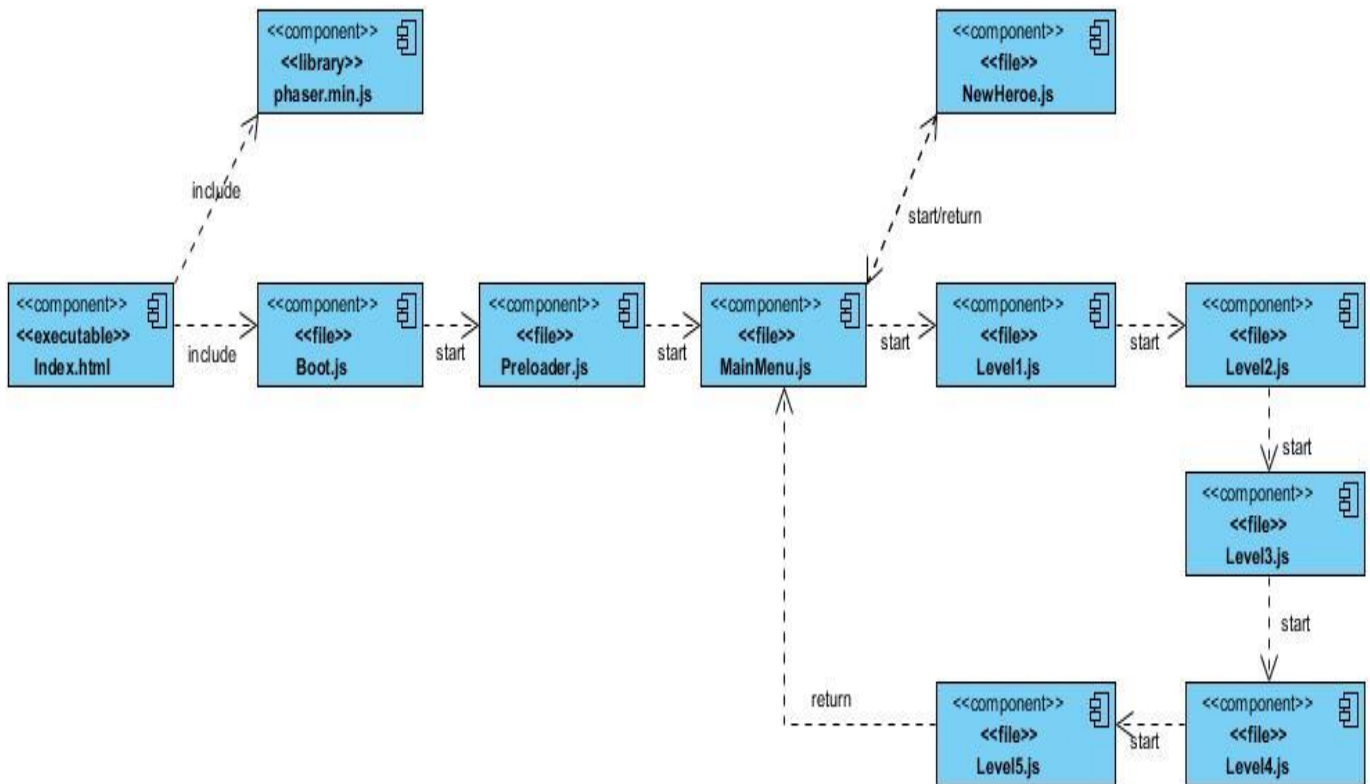


Ilustración 9: Secuencia de scripts

Funciones de los scripts

Tabla 20: Script Inicial

Nombre	<i>Boot</i>
Descripción	Es el script que da inicio al proceso de ejecución del videojuego. Inicializa la clase <i>Play</i> , clase principal que sostiene el cuerpo de la aplicación.

Tabla 21 Script Carga

Nombre	<i>Preloader</i>
Descripción	Carga en memoria todos los <i>sprites</i> ⁴ necesarios para la ejecución de la aplicación.

⁴ Imagen que contiene varios momentos de una animación. Cada uno de esos momentos es llamado *frame*.

Tabla 22 Script Menú principal

Nombre	<i>MainMenu</i>
Descripción	Presenta el menú principal, que contiene las opciones de crear un nuevo jugador, jugar y activar/desactivar el audio en las opciones.

Tabla 23 Script Nuevo héroe

Nombre	<i>NewPlayer</i>
Descripción	Permite crear un nuevo perfil de jugador.

Tabla 24 Script Nivel 1

Nombre	<i>Level1</i>
Descripción	Contiene la implementación de la técnica de medición del campo visual campimetría estática.

Tabla 25: Script Nivel 2

Nombre	<i>Level2-5</i>
Descripción	Representa un área de juego. Contiene la implementación progresiva de la campimetría cinética.

3.3 Pruebas

No existe un método estándar para las pruebas de juego, y la mayoría de las metodologías son desarrolladas por desarrolladores de videojuegos individuales. Las metodologías se refinan continuamente y las pruebas se llevan a cabo según el tipo de juego. El sub-proceso de prueba o el *testing* es uno de los más importantes dentro del desarrollo de videojuegos, ya que a través de este se controla la calidad del videojuego. La función principal de las pruebas es el descubrimiento y documentación de los defectos de *software* (JUEGALIBRE, 2013).

Después de concluido el proceso de desarrollo y la aplicación se encuentra en estado funcional es probada por el equipo de desarrollo, que ha estado involucrado en el diseño e implementación del juego, en busca de errores para su refinamiento. El principal aspecto a probar es la compatibilidad.

Pruebas de compatibilidad

Para evaluar la compatibilidad del videojuego fue necesario ejecutarlo en diferentes navegadores. A continuación se muestran las observaciones obtenidas de la ejecución en los navegadores seleccionados.

El videojuego fue probado en los navegadores Mozilla Firefox [Ilustración 10], SeaMonkey [Ilustración 11], Opera [Ilustración 12] y Google Chrome [Ilustración 13], en los dos primeros se puede realizar la ejecución directa del videojuego, mientras que para los restantes es necesario que se encuentre alojado en un servidor *web*. También podrá ser ejecutado por cualquier sistema operativo donde sean ejecutados dichos navegadores en las versiones que soporten HTML5.

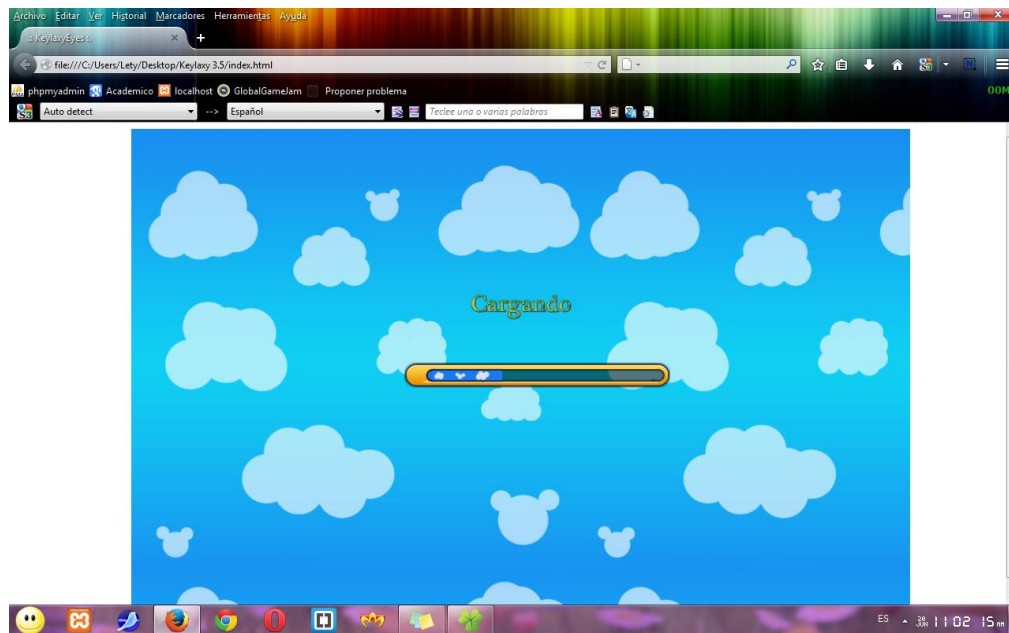


Ilustración 10: Vista en el navegador Mozilla Firefox



Ilustración 11: Vista en el navegador SeaMonkey

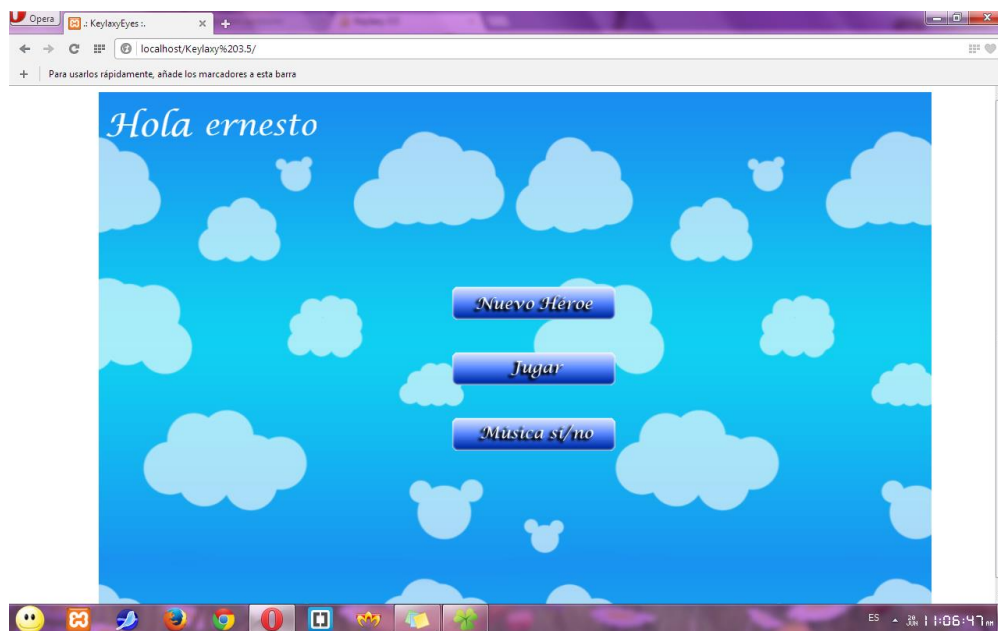


Ilustración 12: Vista en el navegador Opera

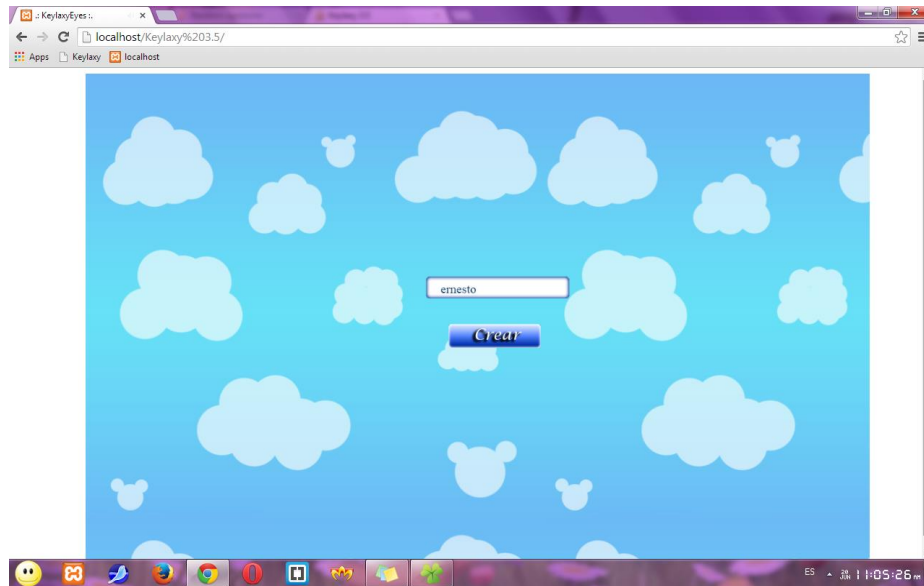


Ilustración 13: Vista en el navegador Google Chrome

Pruebas de aceptación

Durante la realización de las pruebas de aceptación fueron consultados 5 especialistas pertenecientes al Instituto Cubano de Oftalmología Ramón Pando Ferrer, la escuela especial Abel Santamaría de la Lisa y un especialista en temas de rehabilitación con sistemas de realidad virtual. A las personas consultadas les fue presentado el prototipo funcional del videojuego y se describió la solución desde el punto de vista de rehabilitación, haciendo énfasis en el procedimiento de las técnicas de campimetría utilizadas. Como resultado del encuentro se pudo constatar que son válidas las vías escogidas para la implementación de las técnicas de campimetría en la solución. Además de que el prototipo obtenido constituye un paso importante en la construcción de una alternativa de rehabilitación del campo visual en niños.

3.4 Consideraciones parciales

- Con la realización de las pruebas de compatibilidad se demostró que la solución desarrollada puede ser ejecutada en cualquier navegador *web* que soporte HTML5, característica que responde también al cumplimiento del requisito de portabilidad. Además durante las pruebas de aceptación se pudo constatar que el prototipo presentado puede ser una vía de solución al problema planteado.

Conclusiones generales

Teniendo en cuenta la opinión de los especialistas el videojuego KeylaxyEyes, a partir de la implementación de la campimetría estática como vía para analizar las zonas afectadas del campo visual así como la campimetría cinética como medio de rehabilitación, constituye un paso importante en la rehabilitación del campo visual en los niños.

El aprovechamiento de las facilidades de las tecnologías *web* para el desarrollo de videojuegos serios permitió la creación una aplicación con un alto grado de compatibilidad y portabilidad, ya que pueden ser ejecutados en cualquier navegador *web* que soporte HTML5. Además la utilización de la metodología Huddle y el empleo del motor de videojuegos de navegador Phaser, facilitó un ciclo de desarrollo rápido y efectivo de la solución en función del cumplimiento de los requisitos.

Recomendaciones

Al concluir el presente trabajo y en vista de continuar con su desarrollo se recomienda:

- Aumentar la cantidad de información almacenada en el JSON en vista de que el doctor pueda obtener la mayor cantidad de información sobre sus pacientes.
- Implementar un método de guardado automático al finalizar cada nivel.
- Incorporar técnicas de interacción más naturales como el uso del Kinect.
- Validar la solución con niños que presenten afecciones del campo visual.

Referencias Bibliográficas

ABT, C. *Serious Games*. New York: Viking Press, 1970.

ALVAREZ, J. y RAMPNOUX, O. "Serious Game: Just a question of posture?". 2007. Artificial & Ambient Intelligence.

ANDREARRS. *5 librerías en JavaScript para hacer juegos en HTML5* Hipertextual, de 2015]. Disponible en: <http://hipertextual.com/archivo/2014/08/librerias-javascript-para-hacer-juegos/>.

ARIAS-DÍAZ, A.; BERNAL-REYES, N., et al. *Medición de agudeza visual estereoscópica en una población infantil sana*. 2013, Disponible en: http://revista.smo.org.mx/pdf/8704/2013-87-4-05%20AO_215-219.pdf.

BARADA, D. D. H.; ARMAS, D. M. M. D., et al. Rehabilitación óptica en la degeneración macular relativa a la edad *Revista Cubana de Oftalmología*, 2002, vol. 15, nº Disponible en: http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S0864-21762002000200005.

BURKE, J. W.; MCNEILL, M., et al. Augmented reality games for upper-limb stroke rehabilitation. En *Games and Virtual Worlds for Serious Applications (VS-GAMES), 2010 Second International Conference on*. 2010. p. 75-78.

CUBBIDGE, R. P.; DOSHI, S., et al. *Campos visuales*. 2006. 120 p. ISBN 84-458-1597-0.

DAMM, S. *A comparison of tools for the development of HTML 5 Applications*. [Consultado el: 1 de noviembre de 2014]. Disponible en: <http://www.oio.de/public/opensource/comparison-IDE-for-HTML5-CSS3-JavaScript-shootout.htm>.

DE MEDINA Y BONO, F. A. *Valor clínico de la campimetría cuantitativa en el diagnóstico de la esclerosis múltiple, sin manifestaciones clínicas oculares aparentes, y sus relaciones con otras formas clínicas*. Publicaciones de "Juste", 1944.

FERGARCIA. *Entorno de Desarrollo Integrado (IDE)*. 2015, vol. 2015, Disponible en: <https://fergarcia.wordpress.com/2013/01/25/entorno-de-desarrollo-integrado-ide/>.

GAVALDÀ, J. D.; NAVARRO, H. T., et al. *Introducción a los videojuegos*. Universitat Oberta de Catalunya, 2011.

GÓMEZ, A. P. y OZETE, R. E. P. *Videojuego serio para la enseñanza del esqueleto de la cabeza humana*. Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas., Universidad de las Ciencias Informáticas., 2013.

GREGORY, J. *Game Engine Architecture*. CRC Press, 2009. ISBN 9781439865262.

INFOCOP. *Videojuegos con fines terapéuticos. Entrevista a los psicólogos encargados del proyecto Playmancer en España*. de 2015]. Disponible en: http://www.infocop.es/view_article.asp?id=2235.

JUEGALIBRE. *Prueba de juego* Disponible en: <http://juegalibre.virtual.uniandes.edu.co/index.php/2013/03/07/testing-prueba-de-juego/>.

LAWSON, B. y SHARP, R. *Introducing HTML5*. Berkeley Peachpit, 2011a.

---. *Introducing HTML5*. 2011b, 240 p. Disponible en: <http://sunshine.prod.uci.edu/gridfs/sunshine/books/Introducing-HTML5---New-Riders.pdf>. ISBN 978-0-321-68729-6.

MARCANO, B. *Juegos serios y entrenamiento en la sociedad digital*. 2008, nº Disponible en: http://www.usal.es/~teoriaeducacion/rev_numero_09_03/n9_03_marcano.pdf

MARTINTO, M. P. C. P.; DÍAZ, D. A. A., *et al. A new approach for self adaptive video game for rehabilitation. Experiences in the amblyopia's treatment*. IEEE, 2014,

MAS, R. G. *Rehabilitación médica*. Masson, 1997. ISBN 9788445804834.

RAVELO, J. C. A. *Videojuego Serio "Montaña Rusa Dinámica" para el tratamiento de la agudeza visual*. Trabajo de Diploma para optar por el título de Ingeniero en Ciencias Informáticas, Universidad de las Ciencias Informáticas, 2014.

ROUSE, R. *Game Design: Theory and Practice, Second Edition*. Jones & Bartlett Learning, 2010. ISBN 9781449633455.

SERRANO, E. P. *Videojuegos para la salud*. En *VI Curso de Comunicación y Salud*. Madrid. 24 y 25 de Noviembre 2010.

TÁPANES, D. S. H. *Medicina de Rehabilitación*. [Online] [Consultado el: 13 de octubre de 2014]. Disponible en: <http://www.sld.cu/sitios/rehabilitacion/temas.php?idv=615>.

URRUTIA, G. A. M.; LÓPEZ, C. E. N., *et al. PROCESOS DE DESARROLLO PARA VIDEOJUEGOS Instituto de Ingeniería y Tecnología. Universidad Autónoma de Ciudad Juárez*, 2010, nº 36/37,

VALLEJO, J. P. *¿Para qué Plataforma Desarrollar mi Videojuego?* de 2015]. Disponible en: <http://www.pivotstudio.co/blog/para-que-plataforma-desarrollar-mi-videojuego/>.

VAN LANCKER, L. *jQuery: El framework JavaScript de la Web 2.0*. Ed. ENI, 2012. ISBN 9782746072589.

VEGA, A. A. *5 razones por las que JSON vence a XML en proyectos web*. de 2015]. Disponible en: <http://www.analyticaweb.com/desarrollo-web/json-versus-xml-en-proyectos-web>.

WADDINGHAM, P.; BUTLER, T., *et al*. Preliminary results from the use of the novel Interactive Binocular Treatment (I-BiT™) system, in the treatment of strabismic and anisometropic amblyopia. *Eye*, 2006, vol. 20, nº 3, p. 375-378. ISSN 0950-222X.

WOLF, M. J. P. y PERRON, B. INTRODUCCIÓN A LA TEORÍA DEL VIDEOJUEGO *Revista de Comunicación Audiovisual. Universidad Pompeu Fabra*, 2005, nº

ZELDMAN, J. *Diseño con estándares web*. 2010.

ZYDA, M. From Visual Simulation to Virtual Reality to Games. *Computer*, 2005, nº p. 38.

Glosario de términos

Arreglo: en programación se conoce arreglo como una secuencia de valores.

Flightgear: simulador de conducción de vehículos.

Input: elemento de formularios HTML que permite la entrada de información.

Item: elementos, objetos que tienen un valor significativo en el videojuego por ejemplo las estrellas.

Jugabilidad: es el conjunto de propiedades que describen la experiencia del jugador ante un sistema de juego determinado, cuyo principal objetivo es entretener de forma satisfactoria y creíble ya sea solo o en compañía de otros jugadores.

Label: elemento de formularios HTML, que permite mostrar información visual pero no puede ser modificado por el usuario.

Mouse: dispositivo de entrada.

Octopus: equipo oftalmológico.

Refactorización: se usa a menudo para describir la modificación del código fuente sin cambiar su comportamiento, lo que se conoce informalmente por limpiar el código.

Scripts: es un programa usualmente simple, que por lo regular se almacena en un archivo de texto plano. El uso habitual de los *scripts* es realizar diversas tareas como combinar componentes, interactuar con el sistema operativo o con el usuario.

Snellen: es una prueba diseñada para evaluar la agudeza visual. La prueba consiste en identificar correctamente las letras en una gráfica conocida como gráfica de *Snellen*. Las letras tienen un tamaño decreciente dependiendo del nivel en que se encuentran. Solo se utilizan diez letras que son B, C, D, E, F, L, O, P, T y la Z.

Sprites: Imagen que contiene varios momentos de una animación. Cada uno de esos momentos es llamado *frame*.

Testing: pruebas.

Transporttycoon: simulador de gestión de compañías.

Virtonomics: simulador sobre negocios en general.

Waterfallprocess: Metodología de desarrollo de *software*.