



**ESPE**  
UNIVERSIDAD DE LAS FUERZAS ARMADAS  
INNOVACIÓN PARA LA EXCELENCIA

**DEPARTAMENTO DE CIENCIAS DE LA COMPUTACIÓN**

**CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA**

**TESIS PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN  
SISTEMAS E INFORMÁTICA**

**AUTOR: REYES ROSERO DAVID ESTEBAN**

**TEMA: “Desarrollo de un videojuego educativo 2D para plataformas  
Web con el motor de juegos HTML5 Turbulenz utilizando la  
metodología OOHDM.”**

**DIRECTORA: ING. ZAMBRANO, MARGARITA, Msc  
CODIRECTOR: ING. FUERTES, WALTER, PhD**

**SANGOLQUÍ, OCTUBRE 2015**

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE  
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

CERTIFICADO

Ing. Margarita Zambrano

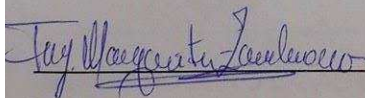
Ing. Walter Fuertes

Que el trabajo titulado “DESARROLLO DE UN VIDEOJUEGO EDUCATIVO 2D PARA PLATAFORMAS WEB CON EL MOTOR DE JUEGOS HTML5 TURBULENZ UTILIZANDO LA METODOLOGÍA OOHDM”, realizado por el Sr. Reyes Rosero David Esteban, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la Universidad de las Fuerzas Armadas – ESPE.

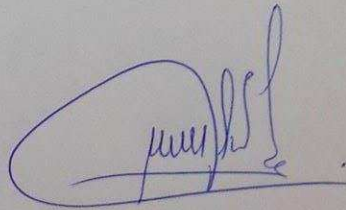
El mencionado trabajo consta de un documento empastado y un disco compacto el cual contiene los archivos en formato portátil de Acrobat (PDF).

Se autoriza al Sr. Reyes Rosero David Esteban, que el material se entregue al Ing. Mauricio Campaña, en su calidad de Director de la Carrera.

Sangolquí, Octubre de 2015



ING. MARGARITA ZAMBRANO, Msc.  
DIRECTORA DE TESIS



ING. WALTER FUERTES, PhD.  
CODIRECTOR DE TESIS

UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE  
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**DECLARACIÓN DE RESPONSABILIDAD**

Reyes Rosero David Esteban

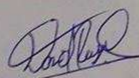
**DECLARO QUE:**

El proyecto de grado denominado "DESARROLLO DE UN VIDEOJUEGO EDUCATIVO 2D PARA PLATAFORMAS WEB CON EL MOTOR DE JUEGOS HTML5 TURBULENZ UTILIZANDO LA METODOLOGÍA OOHDM", ha sido desarrollada en base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan en la bibliografía.

Consecuentemente este trabajo es de autoría propia.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, Octubre de 2015



---

REYES ROSERO DAVID ESTEBAN

C.C: 1718098153

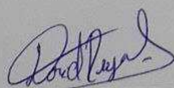
UNIVERSIDAD DE LAS FUERZAS ARMADAS – ESPE  
CARRERA DE INGENIERÍA EN SISTEMAS E INFORMÁTICA

**AUTORIZACIÓN PUBLICACIÓN**

Reyes Rosero David Esteban

Autorizo a la Universidad de las Fuerzas Armadas – ESPE la publicación en la Biblioteca Virtual de la Institución, del trabajo “DESARROLLO DE UN VIDEOJUEGO EDUCATIVO 2D PARA PLATAFORMAS WEB CON EL MOTOR DE JUEGOS HTML5 TURBULENZ UTILIZANDO LA METODOLOGÍA OOHDM”, cuyo contenido, ideas y criterios son de mi responsabilidad y autoría.

Sangolquí, Octubre de 2015



---

REYES ROSERO DAVID ESTEBAN

C.C: 1718098153

## **DEDICATORIA**

A mis padres Marco y Nicole, que durante todos estos años me han apoyado en todo; gracias a ellos es que hoy puedo alcanzar esta meta tan importante en mi vida.

A mis hermanos Sebastián y Carolina, por su apoyo y cariño demostrado.

A mis perros Coqui, Chiqui, Nena, Nomo, Nacho y Nube por acompañarme en mi vida.

**Reyes Rosero David Esteban**

## **AGRADECIMIENTOS**

A la Ingeniera Margarita Zambrano y al Ingeniero Walter Fuertes por su guía y apoyo para lograr este último peldaño de la carrera.

A mis padres por brindarme su apoyo, experiencia y valores.

A mi mejor amigo Daniel por apoyarme y ayudarme en lo que he necesitado, a las hermanitas luna Angie y Mafer, por ayudarme en este proceso.

A mis amigos, compañeros de universidad y a todos los que me olvidé y ayudaron a la elaboración de este proyecto. Gracias por todo.

**Reyes Rosero David Esteban**

## ÍNDICE DE CONTENIDOS

<b>ÍNDICE DE CONTENIDOS</b> .....	<b>vi</b>
<b>ÍNDICE DE FIGURAS</b> .....	<b>x</b>
<b>INDICE DE TABLAS</b> .....	<b>xiii</b>
<b>CAPÍTULO 1: INTRODUCCIÓN</b> .....	<b>1</b>
1.1. Tema.....	1
1.2. Antecedentes .....	1
1.3. Planteamiento del problema.....	1
1.4. Justificación.....	2
1.5. Objetivos .....	3
1.5.1. Objetivo general .....	3
1.5.2. Objetivos específicos.....	3
1.6. Alcance .....	4
1.7. Factibilidad.....	5
1.7.1. Factibilidad Técnica .....	5
1.7.2. Factibilidad Operativa.....	6
<b>CAPÍTULO 2: MARCO DE REFERENCIA</b> .....	<b>7</b>
2.1. Videojuegos .....	7
2.1.1. Historia de los videojuegos.....	7
2.1.2. Clasificación de los videojuegos.....	8

2.1.3. Motor de videojuegos Turbulenz .....	12
2.1.4. Metodología de desarrollo de videojuegos HUDDLE .....	18
2.1.4.1. Preproducción .....	19
2.1.4.2. Producción .....	20
2.1.4.3. Postmortem .....	22
2.1.5. Herramientas de desarrollo .....	23
2.1.5.1. HTML5.....	23
2.1.5.2. JavaScript .....	26
2.1.5.3.CSS .....	27
2.1.6. Educación y videojuegos .....	28
2.1.7. Videojuegos en la Educación .....	31
2.1.7.1. Aprendizaje basado en juegos Digitales (DGBL).....	32
2.2. Metodología OOHDM .....	32
2.2.1. Obtención de requerimientos.....	33
2.2.2. Diseño Conceptual .....	33
2.2.3. Diseño Navegacional .....	33
2.2.4. Diseño de Interfaz Abstracta.....	34
2.2.5. Implementación.....	34
<b>CAPÍTULO 3: ANÁLISIS Y DISEÑO .....</b>	<b>35</b>
3.1 Preproducción .....	35



3.1.1 Documento de Diseño .....	35
3.1.2 Especificación de Requerimientos .....	37
3.1.2.1 Identificación de Roles y Tareas .....	37
3.1.2.2 Especificación de Escenarios .....	37
3.1.2.3. Especificación de casos de uso por Actor .....	38
3.1.3. Diseño Conceptual.....	42
3.1.3. Diseño Navegacional.....	42
3.1.3.1. Objetos Navegacionales.....	42
3.1.3.2. Contextos Navegacionales.....	43
3.1.3.3. Esquema Navegacional .....	43
<b>CAPÍTULO 4:IMPLEMENTACIÓN DEL VIDEOJUEGO .....</b>	<b>48</b>
4.1 MOTOR DE VIDEOJUEGOS TURBULENZ .....	48
4.1.1 INSTALACIÓN.....	48
4.1.1.1 Enviroment .....	48
4.1.1.2 Servidor Local .....	49
4.1.2 Turbulenz Hub .....	50
4.2 Producción del Videojuego .....	52
<b>CAPÍTULO 5:PRUEBAS Y EVALUACIÓN DE RESULTADOS .....</b>	<b>61</b>
5.1 Pruebas funcionales .....	61

5.1.1. Duración de la evaluación.....	61
5.1.2. Infraestructura utilizada .....	61
5.1.3. Variables a medir .....	62
5.1.4. Resultados obtenidos.....	62
5.1.5. Evaluación pedagógica .....	66
5.2. Pruebas de Rendimiento.....	68
5.3. Etapa Postmortem.....	69
<b>6. CONCLUSIONES Y RECOMENDACIONES .....</b>	<b>71</b>
6.1. Conclusiones.....	71
6.2. Recomendaciones.....	72
<b>ANEXO A1: MANUAL DE USUARIO.....</b>	<b>Error! Bookmark not defined.</b>
<b>Bibliografía</b>	<b>73</b>

## ÍNDICE DE FIGURAS

Figura 1 Videojuego "Save the Day", creado por Denki.....	13
Figura 2. Ejemplo de Cuerpos Rígidos .....	15
Figura 3. Ejemplo de Constraints.....	16
Figura 4. Etapas del proceso de desarrollo de videojuegos HUDDLE.....	19
Figura 5. Fases de la etapa de Preproducción .....	20
Figura 6. Fases de la etapa de Producción .....	22
Figura 7. Fases de la etapa Postmortem .....	23
Figura 8. Ejemplo de Botones y cajas de texto sin CSS .....	27
Figura 9. Ejemplo de botones y cajas de texto con CSS. ....	27
Figura 10. Casos de uso del Videojuego "Recoge letras" .....	38
Figura 11. Diagrama de Clases del videojuego "Recoge Letras".....	42
Figura 12. Esquema navegacional del videojuego "Recoge Letras" .....	43
Figura 13. ADV Pantalla Inicial .....	44
Figura 14. ADV Pantalla Acerca de .....	44
Figura 15. ADV Pantalla Pausa .....	45
Figura 16. ADV Pantalla Controles .....	45
Figura 17. ADV Pantalla de Juego.....	46
Figura 18. ADV Pantalla Logro Completado .....	46
Figura 19. ADV Pantalla Fin de Juego.....	47
Figura 20. Consola de comandos Turbulenz .....	49
Figura 21. Servidor local de Turbulenz activado .....	49
Figura 22. Página principal Servidor Local Turbulenz.....	50

Figura 23. Plataforma Turbulenz.....	51
Figura 24. Plataforma de Turbulenz Hub (online) .....	52
Figura 25. Pantalla principal.....	54
Figura 26. Pantalla Como Jugar .....	55
Figura 27. Pantalla Créditos .....	55
Figura 28. Pantalla Inicio del Nivel.....	56
Figura 29. Pantalla del videojuego en Ejecución .....	56
Figura 30. Pantalla Juego Pausado .....	57
Figura 31. Pantalla Fin del Juego .....	57
Figura 32. Pantalla de logro completado nivel 1 .....	58
Figura 33. Pantalla de logro completado nivel 2 .....	58
Figura 34. Pantalla de logro completado nivel 3 .....	59
Figura 35. Diseño del Título del Juego .....	59
Figura 36. Diseño de la animación del personaje principal.....	60
Figura 37. Tiempo promedio por nivel para estudiantes entre 7 a 9 años ..	62
Figura 38. Promedio de palabras acertadas por nivel para estudiantes entre 7 a 9 años .....	63
Figura 39. Tiempo promedio de cada nivel dependiendo el género .....	63
Figura 40. Promedio de palabras acertadas en cada nivel dependiendo el género.....	64
Figura 41. Cantidad de estudiantes por número de palabras acertadas en cada nivel.....	64
Figura 42. Porcentaje de estudiantes por cantidad de aciertos nivel 1 .....	65

Figura 43. Porcentaje de estudiantes por cantidad de aciertos nivel 2 .....	65
Figura 44. Porcentaje de estudiantes por cantidad de aciertos nivel 3 .....	66
Figura 45. Promedio general de la valoración del videojuego .....	67

**ÍNDICE DE TABLAS**

Tabla 1. Estadios del desarrollo según Piaget.....	29
Tabla 2. Documento de Diseño.....	35
Tabla 3. Caso de uso Iniciar videojuego .....	39
Tabla 4. Caso de uso Pausar videojuego .....	39
Tabla 5. Caso de uso Leer Acerca de.....	40
Tabla 6. Caso de uso Consultar controles .....	40
Tabla 7. Caso de uso Silenciar Videojuego .....	41
Tabla 8. Caso de uso Salir del Videojuego .....	41
Tabla 9. Documento de Producción.....	52
Tabla 10. Escala utilizada para la valoración de resultados .....	67
Tabla 11. Resultados pruebas de rendimiento 1.....	68
Tabla 12. Resultados pruebas de rendimiento 2.....	69
Tabla 13. Reporte Postmortem .....	69

## RESUMEN

La aparición de nuevas tecnologías así como la extinción de otras hace que el desarrollo de aplicaciones y videojuegos Web evolucionen a la par de estos. El estándar HTML5 busca solventar ciertas limitaciones que se han presentado con complementos como el de Flash que hoy en día está en juego su existencia. El objetivo del presente proyecto fue el de desarrollar un videojuego educativo para niños de 7 a 11 años utilizando un motor de videojuegos llamado Turbulenz el cual utiliza las tecnologías HTML5 y JavaScript para el desarrollo de videojuegos que puedan ser ejecutados sin la necesidad de complementos y dentro de cualquier navegador. Para el análisis y desarrollo del videojuego se utilizó el proceso de desarrollo de videojuegos Huddle complementando en la fase de diseño con la metodología OOHDM, lo que permitió la creación exitosa del videojuego mediante un proceso ágil. El videojuego tiene como propósito ayudar a la coordinación óculo-manual, concentración y también sirve como refuerzo del vocabulario aprendido dentro de la escuela. El resultado del presente proyecto es un videojuego de tipo educativo denominado “Recoge Letras”, el cual fue evaluado en un grupo de 66 estudiantes conformados por 33 niños y 33 niñas de la escuela “Marqués de Selva Alegre”, ubicada en el valle de los Chillos. En el análisis de resultados los profesores realizaron una validación pedagógica del videojuego en donde se obtuvo una calificación de 3.81 sobre 4.

### **PALABRAS CLAVE:**

- **HTML5**
- **VIDEOJUEGO EDUCATIVO**
- **TURBULENZ**
- **JAVASCRIPT**
- **OOHDM**

## **ABSTRACT**

The emergence of new technologies and the extinction of others makes Web application development and video games evolve simultaneously. The HTML5 standard seeks to resolve certain limitations that have arisen with plugins like Flash that now have their existence at stake. The main goal of this project was to develop an educational game for children between 7 and 11 years using a game engine called Turbulenz which uses HTML5 and JavaScript technologies to develop games that can be executed in any browser without the need for add-ons. The development process for games named Huddle and the OOHDM methodology were used in the development process of the game which allowed the successful creation of the game through an agile process. The game aims to help hand-eye coordination, concentration and also serves like reinforcement for the vocabulary learned at school. As a result of this project, the educational video game "Recoge Letras" was created and was tested in a school, located in Chillos Valley. In the analysis of results a score of 3.81 out of 4 was obtained by a teacher's validation of the video game.

### **KEYWORDS:**

- **HTML5**
- **EDUCATIONAL VIDEO GAME**
- **TURBULENZ**
- **JAVASCRIPT**
- **OOHDM**



# **CAPÍTULO 1: INTRODUCCIÓN**

## **1.1. Tema**

“Desarrollo de un videojuego educativo 2D para plataformas Web con el motor de juegos HTML5 Turbulenz utilizando la metodología OOHDM”

## **1.2. Antecedentes**

Los videojuegos son una variación de juegos electrónicos los cuales pueden ser ejecutados en cualquier dispositivo (computador, consola, teléfono celular). Dentro del campo educacional los videojuegos se presentan como una opción para aumentar la motivación por parte de los estudiantes, ayudándoles a mejorar ciertas habilidades dependiendo del objetivo del juego.

Los videojuegos han evolucionado desde su creación. Tanto el hardware como el software han tenido grandes avances, pasaron de ser juegos simples a juegos extremadamente complejos, de poseer gráficas simples a crear mundos virtuales; cada vez se busca llegar a alcanzar una similitud con el mundo real. Gracias a la Internet muchos juegos permitieron conectar a millones de usuarios en el mundo, y el desarrollo de videojuegos orientados a la Web permitió que el acceso hacia los videojuegos sea mucho más fácil. (Squire, 2003)

## **1.3. Planteamiento del problema**

Actualmente la tecnología en el mundo crece a pasos agigantados y cada vez las nuevas generaciones se adaptan a ella fácilmente y a edades

más tempranas. Sin embargo se tiene que considerar que la tecnología juegue un papel importante dentro de la educación. Una de estas áreas son los videojuegos, donde se puede observar que la industria de estos mismos ha crecido gracias a grandes avances tecnológicos tanto de hardware y de software, pero ese crecimiento va enfocado en su mayoría al entretenimiento más que en la educación.

En muchas ocasiones se considera que los videojuegos únicamente provocan una adicción y no tienen un efecto positivo dentro de la sociedad. No obstante, el desarrollo de videojuegos que ayuden a mejorar habilidades de los niños puede ser un complemento importante para la educación. (Griffiths, 2002)

Lastimosamente la mayor parte de videojuegos se los desarrolla orientados a una plataforma específica y en el caso de los videojuegos orientados a la Web algunos de ellos requieren de ciertos componentes para su correcto funcionamiento en las diferentes plataformas. Esto puede ocasionar la pérdida del interés por parte de los usuarios ya que se ven obligados a instalar componentes que les faltan o simplemente no poder jugarlos.

A su vez existen videojuegos que requieren ser descargados para luego ser instalados, lo cual puede ser bastante demoroso dependiendo del juego y en el caso de que se cambie de equipo se deberá repetir este procedimiento.

## **1.4. Justificación**

Según lo planteado dentro de la identificación del problema se ha visto necesario desarrollar un videojuego orientado a la web que sea capaz de ser ejecutado dentro de cualquier navegador. Por este motivo se pretende

desarrollar un videojuego educativo que tenga como principal objetivo desarrollar la percepción, la memoria visual, la coordinación óculo-manual y que sirva como refuerzo al vocabulario de los niños, dándoles otra opción de aprendizaje.

Existen varios beneficios de los videojuegos dentro de la educación ya como la mejora de la coordinación óculo-manual, la confianza en uno mismo, etc. (Griffiths, 2002)

Por otra parte se está planteando la utilización de un motor de juegos HTML5 ya que es un Lenguaje orientado a la Web que posee nuevas características que permiten un mejor desempeño dentro de los diferentes navegadores. Así al realizarse el videojuego utilizando este tipo de lenguaje se logra que el usuario no necesite descargarse ningún tipo de componente extra para que el juego pueda ser ejecutado lo cual es una ventaja para que los niños entre las edades de 7 y 11 años puedan acceder al videojuego sin impedimento alguno.

## **1.5. Objetivos**

### **1.5.1. Objetivo general**

Desarrollar un videojuego educativo 2D para plataformas Web con el motor de juegos HTML5 Turbulenz utilizando la metodología OOHDM para niños entre los 7 y 11 años de edad.

### **1.5.2. Objetivos específicos**

1. Analizar el motor de juegos HTML5 Turbulenz para el desarrollo del videojuego.

2. Revisar los principios didácticos para el desarrollo del videojuego educativo.
3. Diseñar e Implementar un videojuego educativo siguiendo los lineamientos de la metodología OOHDm.
4. Realizar pruebas de funcionalidad del videojuego, procesamiento estadístico y la evaluación de resultados.

## **1.6. Alcance**

Se desarrollará un videojuego educativo 2D orientado para plataformas Web para niños entre los 7 y 11 años de edad. Las actividades del videojuego están orientadas a desarrollar la percepción, la memoria visual, la coordinación óculo-manual y servirá como refuerzo para el vocabulario de los niños.

Junto al videojuego se entregará un manual de usuario que permitirá entender el manejo del mismo. Éste manual se lo puede encontrar en la sección de anexos Anexo A1.

Se realizará las pruebas de funcionalidad en una escuela, el procesamiento estadístico y se evaluará los resultados obtenidos.

El videojuego tendrá las siguientes funciones:

- Iniciar el videojuego.
- Pausar el videojuego.
- Leer Acera del videojuego (Créditos).
- Consultar Controles y modo de juego.
- Elegir nivel de dificultad
- Silenciar videojuego.
- Salir del Videojuego.

## 1.7. Factibilidad

Los principales aspectos para este estudio fueron clasificados en dos áreas, las cuales se describen a continuación:

### 1.7.1. Factibilidad Técnica

Técnicamente el desarrollo del videojuego es viable, ya que la plataforma de desarrollo se lo puede realizar en cualquier editor de texto y el motor de videojuegos existe para las plataformas Windows, Mac y Linux.

El proyecto es factible ya que el tesista cuenta con los elementos de hardware y de Software requeridos para su desarrollo.

- **Hardware**
  - 1 Computador ASUS G55V
  - Intel Core i7 2.3GHz
  - Memoria DDR3 de 8GB
  - Disco Duro de 500BG
  
- **Software**
  - Python 2.7.6
  - Turbulenz engine 0.28.0
  - Turbulenz SDK 0.28.0
  - Sublime Text editor

Posteriormente, le implantación del proyecto también es factible ya que Turbulenz proporciona un portal para la publicación del videojuego.

### **1.7.2. Factibilidad Operativa**

El proyecto propuesto es factible en el ámbito operativo, ya que se tiene acceso tanto a la información, herramientas necesarias para el desarrollo del videojuego educativo y el know-How de los profesores de la Universidad de las Fuerzas Armadas-ESPE.

## **CAPÍTULO 2:**

### **MARCO DE REFERENCIA**

#### **2.1. Videojuegos**

##### **2.1.1. Historia de los videojuegos**

Existen diferentes opiniones acerca del primer videojuego creado por el ser humano debido a que no existía una definición de lo que era un videojuego.

Por este motivo se puede decir que los videojuegos tienen su origen en la misma época del desarrollo informático, esto quiere decir entre los años 1950-1960. En estos años se observa la aparición de los primeros prototipos de juegos de ordenador a modo de experimentos por diversas universidades. (Gil Juárez & Vida Mombiela, 2007).

En 1972 la empresa Atari lanza al mercado su primer videojuego llamado Pong, un juego sencillo que consistía de dos paletas y una pelota y se simulaba una partida de tenis de mesa (ping pong), en el cual los jugadores tenían que obtener más puntos que su contrincante, estos puntos se obtenían cuando uno de los dos jugadores no lograba devolver la pelota. Este videojuego se lo podía jugar en las máquinas que funcionaban con monedas (juegos de Arcade). Otros juegos muy populares de esta época fueron Space Invaders, Asteroids entre otros.

Con los avances tecnológicos, el acceso y la inserción de los ordenadores dentro de los hogares fue un tema mucho más fácil y así muchas empresas empezaron a crear mejores consolas de videojuegos, así como videojuegos con una mayor calidad gráfica.

Dentro de los videojuegos más famosos y exitosos de la época de los años 80 se puede encontrar a PacMan (1980) creado por Namco, Mario Bros de Nintendo (1985) y Tetris creado por Alexey Pajitnov (1985).

En la década de los 90 las videoconsolas evolucionan rápidamente, mejorando sus entornos gráficos, sonoros; permitiendo generar ambientes en 3 dimensiones, dentro de esta época nacen consolas como la de PlayStation y Nintendo 64, consolas que han seguido vigentes hasta estos años gracias a su constante evolución. Aquí se puede encontrar videojuegos como Monkey Island, Sonic, La Leyenda de Zelda, Mortal Kombat, Doom, FIFA, World of Warcraft, Final Fantasy, etc. (Gil Juárez & Vida Mombiela, 2007)

Hoy en día existen 3 empresas que dominan en el ámbito de las consolas de videojuegos y son:

- PlayStation con su más reciente consola la PlayStation 4.
- Microsoft con la XBOX ONE.
- Nintendo con su Wii U.

En la actualidad, se pueden encontrar videojuegos para plataformas, distintas de las anteriormente mencionadas; tales como computadoras de escritorio, laptops, dispositivos móviles, celulares, etc.

### **2.1.2. Clasificación de los videojuegos**

Existen varias formas de clasificación para los videojuegos, dependiendo del contexto del juego, su ambientación, por consolas, etc.



Uno de estos autores los clasifica en 6 grupos dentro de los cuales pueden existir diferentes subgéneros. (Gil Juárez & Vida Mombiela, 2007).

- **Videojuegos de Acción.** Caracterizados por necesitar una gran percepción visual o precisión de los controles, dentro de este se pueden encontrar los siguientes subgéneros.
  - Shooters (juegos de disparo). Ej. Call of Duty, Counter Strike.
  - Pelea o Lucha. Ej. Tekken, Mortal Kombat.
  - Plataformas. El personaje debe avanzar saltando las plataformas que existen dentro de los diferentes escenarios. Ej. Super Mario Bros.
  - Arcade. Son los videojuegos clásicos como PacMan.
  
- **Videojuegos de Aventura.** Caracterizados por contar una historia la cual el personaje debe llevar a cabo, y cumplir ciertos objetivos para llegar al desenlace de la historia. Como subgéneros se encuentran los siguientes:
  - Aventuras conversacionales: para continuar la trama de la historia del videojuego es necesario introducir una serie de comandos, se basa en textos.
  - Aventuras gráficas: Busca resolver rompecabezas dependiendo de la historia, es necesario interactuar con los objetos del videojuego a través de una interfaz.
  - Survival Horror: Su principal temática es la de supervivencia en un ambiente de terror.
  - Hit n' Run (golpea y corre): dentro de este juego existen misiones las cuales permiten avanzar en la historia. El jugador tiene total libertad para interactuar con todo lo que este dentro del juego.

- Juegos de rol: se caracteriza porque el jugador crea a su propio personaje dentro de un escenario ambientado y ficticio, el personaje tiene la capacidad de evolucionar. Ej. World of Warcraft, Dota.
- **Videojuegos Deportivos.** Estos videojuegos llevan al mundo digital todo tipo de deporte existente en la vida real. Aquí se puede encontrar videojuegos de fútbol, básquetbol, tenis, carreras de autos, etc.  
Dentro de estos videojuegos se tiene la oportunidad de jugar contra la máquina o con otros jugadores, también tienen la opción de realizar campeonatos, competencias online o de administrar un equipo como si el jugador representara al director técnico. Ej. FIFA, Top Spin, Gran Turismo.
- **Videojuegos de estrategia.** Es necesario llevar a cabo una buena planificación así como seguir un orden para llegar al objetivo final. Poseen una gran cantidad de interfaces para detallar los recursos y sus acciones. Estos se dividen en:
  - Dependiendo de la temática que proponen se puede encontrar videojuegos de Estrategia social, económica o bélica, etc. Ej. Age of Empires.
  - Dependiendo la mecánica se encuentran los juegos en tiempo real RTS (Real Time Strategy) Ej. Warcraft o los juegos por turnos Ej. Civilization.
- **Videojuegos de Simulación.** Son videojuegos que buscan recrear ciertas actividades o algún funcionamiento específico. Algunos son utilizados como herramientas de aprendizaje en el sentido profesional. Dentro de este se encuentran los siguientes subgéneros:
  - **Simuladores de naves:** Poseen los comandos de control de los aviones, barcos, submarinos, etc. Ej. Silent Hunter.

- **Simuladores de sistemas.** Recreación de una ciudad, hospital, zoológico y más. Ej. ZooTycoon.
- **Simuladores sociales.** Son entornos sociales donde su principal objetivo son las relaciones personales uno de los videojuegos más populares de este tipo es Los Sims.
- **Videojuegos Sociales.** Estos videojuegos traen como controles adicionales por ejemplo raquetas, cámaras de video, instrumentos musicales entre otros. Su principal objetivo es de reunir a un grupo de personas, amigos a que lo jueguen dentro de un mismo espacio físico. Dentro de este género se puede encontrar los siguientes subgéneros:
  - **Juegos Concurso:** Estos videojuegos reproducen a los concursos de preguntas y respuestas de la televisión Ej. Buzz, Quien quiere ser millonario.
  - **Juegos Musicales.** Juegos donde se canta o se toca algún instrumento musical. Ej. Guitar Hero, Rock Band.
  - **Juegos Deportivos.** Reproducen los movimientos de algunos deportes como el tenis, el box, los bolos. Ej. Wii Sports.
  - **Juegos de movimiento Corporal.** Estos tipos de juego necesitan de un periférico el cual permite el reconocimiento de los movimientos del cuerpo, no es necesario del uso de un control para jugarlos. Ej. Just Dance utilizando el Kinect.

Existen otros autores los cuales proponen el aumento de más categorías como los siguientes: (Wolf, 2008)

- **Juegos de Mesa (Board Games).** Se trata de juegos de mesa que han sido adaptados al mundo digital, así como monopolio, scrabble, ajedrez, entre otros.

- **Juegos de Captura (Capturing).** En estos videojuegos su principal objetivo es el de capturar objetos los cuales buscan evadir o huir del personaje principal. Ej. Keystone Kapers.
- **Juegos de Evasión (Dodging).** EL objetivo principal de estos videojuegos es el de evitar proyectiles o cualquier otro objeto que se mueva hacia el personaje. Ej. Frogger.
- **Juegos Educativos (Educational).** Videojuegos que han sido diseñados para la educación dentro de los cuales su principal objetivo es el enseñar algún tema en específico. Ej. Juegos de Deletrear, juegos de palabras, matemática básica, etc.

### 2.1.3. Motor de videojuegos Turbulenz

Es un motor de videojuegos HTML5 desarrollado por la empresa Turbulenz Limited, que en el año 2013 libera su código bajo la licencia MIT.

Posee librerías en el lado del servidor las cuales están desarrolladas en JavaScript y TypeScript. Lo cual permite crear y publicar juegos monetizados o gratis ya sean 2D o 3D que puedan ser ejecutados en plataformas que soporten HTML5 sin la necesidad de instalar complementos, así como los navegadores modernos y posee las siguientes características generales:

- Carga de recursos asíncrona.
- Múltiples hilos de evaluación y ejecución.
- Librerías para el desarrollo de prototipos.
- Herramientas para la generación de código HTML, TypeScript.
- Motor de física.
- Motor de colisiones
- Integración con las redes sociales Facebook, google+, twitter.
- Plataforma Web para pruebas del videojuego.
- Plataforma gratuita para la publicación del videojuego final.

Cuando se desarrolla un videojuego se tiene que tomar en cuenta aspectos como colisiones, movimiento, velocidad, tiempo depende de la temática del mismo se puede utilizar la gravedad, rotaciones, etc.

En este caso Turbulenz posee funciones ya preestablecidas lo cual facilita el desarrollo de un videojuego, a continuación se describirá estas funciones. La Figura 1. Muestra un ejemplo de un videojuego desarrollado con Turbulenz.



Figura 1 Videojuego "Save the Day", creado por Denki  
 Imagen tomada de: <https://ga.me/games/save-the-day>

### 2.1.3.1. APIs de Bajo Nivel

Estas proveen acceso a las funcionalidades de nivel bajo, son las encargadas de controlar funcionalidades como el sonido, teclado, mouse; son parecidas a las APIs como OpenGL, OpenAL, etc.

Esta librería es formada por los siguientes módulos llamados Devices (dispositivos):

- GraphicsDevice.
- MathDevice.
- PhysicsDevice.
- SoundDevice
- NetworkDevice
- InputDevice

- **Turbulenz Engine Object**

Es uno de los objetos más importantes de este motor de videojuegos, ya que mediante este objeto se puede utilizar todas las funcionalidades nativas de Turbulenz ya sea mediante métodos o por la creación de los dispositivos previamente mencionados.

### 2.1.3.2. APIs de física 2D de Bajo Nivel

Son los encargados específicamente de la física que poseen los cuerpos dentro del juego así como la interacción entre ellos y su comportamiento dentro del mundo de juego. Los más importantes son:

- **Physics2DDevice Object**

Este objeto es muy importante ya que provee la física a los cuerpos creados dentro del mundo. Este objeto utiliza como unidades de medida los metros (m), radianes (rad), segundos (s), kilogramos (kg) y Newtons (N); el eje de coordenadas se representa en X y -Y. Si se trabaja con pixeles se deberá usar una escala. Este objeto también es utilizado para la creación de los siguientes objetos:

- **World Object (Mundo)**

Esta librería es la que simula un mundo con sus propiedades como gravedad representada por un arreglo [X,Y].

- **Rigid bodies (Cuerpos Rígidos)**

Son cuerpos que son creados y puestos dentro del mundo, tienen propiedades como masa, inercia, rotación, velocidad, torque, forma, etc. lo más parecido a las propiedades de un cuerpo en la vida real. Existen 3 tipos de Cuerpos Rígidos:(Ver Figura 2.)

- **Dinámicos**

Este tipo de cuerpos están bajo la influencia de la gravedad, colisiones y ataduras con otros cuerpos.

- **Estáticos**

Estos tipos de objetos no pueden cambiar ninguna de sus propiedades.

- **Cinemáticos**

Estos tipos de objetos son muy parecidos a los estáticos, con la diferencia que su posición puede ser modificada cada vez que se lo diga, para esto se cambia su velocidad o posición.

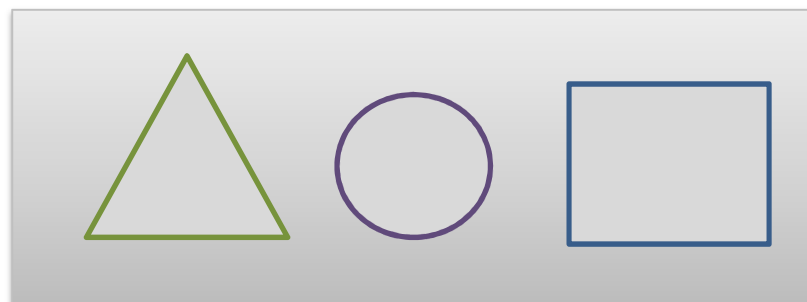


Figura 2. Ejemplo de Cuerpos Rígidos

- **Shape Object (Objeto Forma)**

Este objeto puede ser usado como propiedad de un cuerpo rígido y crea una forma poligonal con coordenadas como parámetros o una forma circular utilizando como parámetros centro y radio.

- **Constraints (restricciones)**

Este objeto tiene la capacidad de restringir el comportamiento de un cuerpo rígido dependiendo del tipo de restricción, por ejemplo se puede crear una atadura entre dos cuerpos rígidos, la cual puede ser elástica o rígida.(Ver Figura 3.)

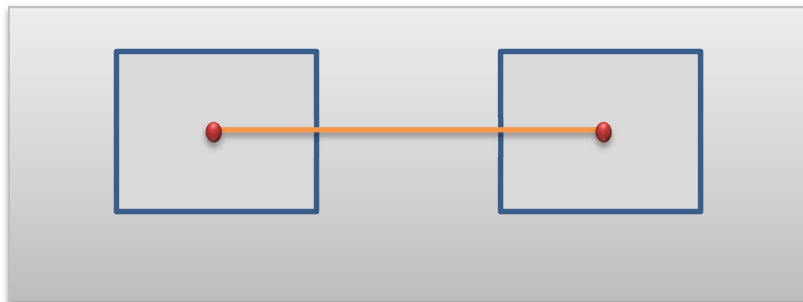


Figura 3. Ejemplo de Constraints

- **Debug Draw (Dibujo de Depuración)**

Este objeto permite dibujar los cuerpos así como las constraints existentes dentro del mundo con propósitos de prueba así como la visualización de puntos de colisión entre dos objetos.

### 2.1.3.3. APIs de Alto Nivel

Estas son bibliotecas desarrolladas en JavaScript las cuales están diseñadas para ser fáciles de usar, son las encargadas de la realización de cálculos como transformaciones a coordenadas en un plano, la reproducción de texturas 2D, de pisos, animaciones, etc. algunas de estas librerías son:

- The Animation Objects
- The AssetTracker Object
- The BoxTree Object
- The Camera Object



- The Canvas Object
- The CharacterController Object
- The DebuggingTools Object
- The DefaultRendering Object
- The Draw2D object
- The Effect Object
- The Floor Object
- The Font Object
- The Light Object

#### **2.1.3.4. Protolib Object**

Es una librería la cual está compuesta por funciones simples para gráficos, sonido, control del teclado y mouse para la realización de prototipos de videojuegos de una manera más sencilla y rápida. Esta librería controla lo siguiente:

- Color de fondo.
- Carga y dibujo de imágenes 2D, 3D.
- Renderización de Texto.
- Sonido.

#### **2.1.3.5. Ventajas**

- Al ser una herramienta de código abierto permite explotar todo su potencial, ya que se puede modificar las librerías existentes para que estas se adapten de una mejor forma a las necesidades requeridas. (Nagle, 2014)

- Posee un portal en el cual se puede publicar los videojuegos desarrollados, así se puede llegar tener una audiencia mayor, además posee características como métricas, datos de usuario, la posibilidad de monetizar el juego, etc. (Turbulenz Limited, 2015)
- Tiene algunas otras herramientas como plantillas y librerías simples para que las personas que estén comenzando puedan realizar un prototipo de un juego de una manera rápida y sencilla. (Turbulenz Limited, 2015)

#### **2.1.3.6. Desventajas**

Pese a que el motor de videojuegos posee un gran potencial y cuenta con excelentes características una de las grandes desventajas por el momento de este motor es que su comunidad de desarrolladores aún es muy pequeña por lo no existe gran cantidad de tutoriales o ejemplos de uso de esta herramienta. (Nagle, 2014)

#### **2.1.4. Metodología de desarrollo de videojuegos HUDDLE**

Es un proceso para el desarrollo específicamente de videojuegos el cual toma características de SCRUM, se caracteriza por ser ágil, iterativo, incremental, evolutivo y óptimo para grupos de trabajo. Huddle propone 2 roles adicionales los cuales son Game Designer y Project Manager. Este proceso de desarrollo se divide en Preproducción, producción y postmortem. (Urrutia, López, Martínez, & Corral, 2010). La Figura 4. Muestra las etapas del proceso de desarrollo de videojuegos HUDDLE.



Figura 4. Etapas del proceso de desarrollo de videojuegos HUDDLE

#### 2.1.4.1. Preproducción

En esta fase se revisa y se acepta la idea inicial, posterior a esto se realiza la planeación completa de la fase de producción. Para esto es necesario de un documento de diseño dentro del cual se exprese la idea principal y detalles de la propuesta de videojuego.

- **Documento de Diseño**

Este documento es una plantilla propuesta por los desarrolladores de este proceso, dentro de esta plantilla se plasma todo lo relacionado con la idea inicial, la temática del juego, un posible nombre del videojuego, en cuales plataformas va a poder ser ejecutado el videojuego. La Figura 5. muestra los subprocesos de la etapa de preproducción.

- **Feature Log**

Una vez que el documento de diseño esté acabado y corregido, se obtiene el feature log, dentro de este se puede identificar los requerimientos o las actividades necesarias para pasar a la etapa de producción.

- **Sprint Plan**

El Sprint Plan es un plan de acción ante los requerimientos obtenidos, una vez que se hayan identificado las tareas a realizar, se realiza un plan y un cronograma de trabajo.

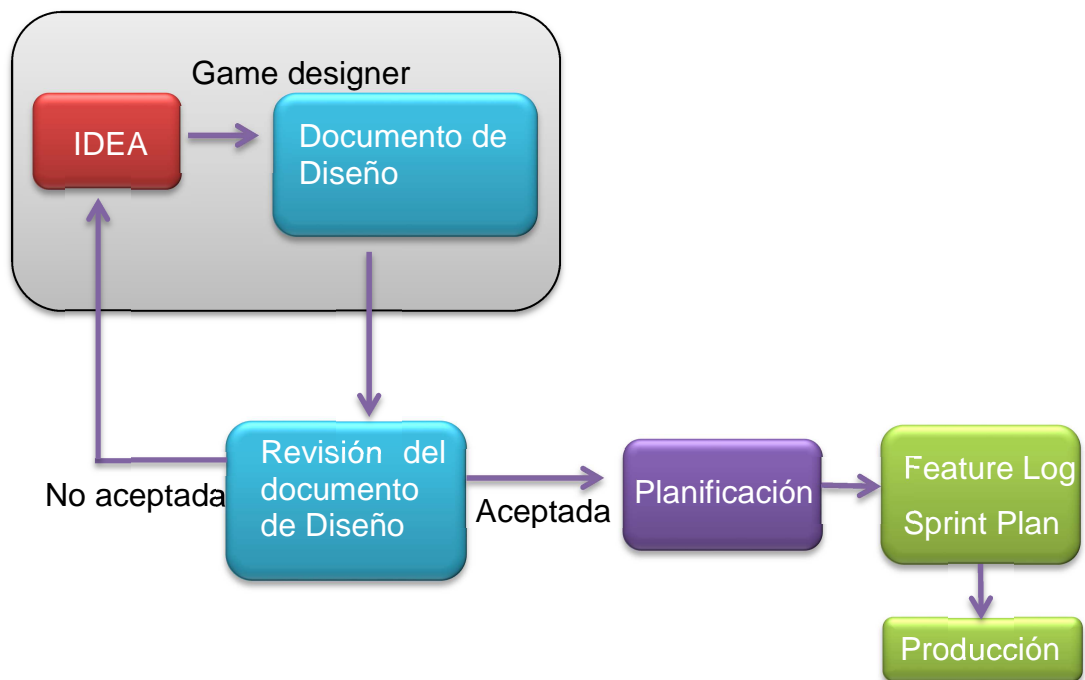


Figura 5. Fases de la etapa de Preproducción

#### 2.1.4.2. Producción

Es la etapa más larga y una de las más importantes. En esta etapa se puede encontrar herramientas provenientes del SCRUM como son los “Daily Meetings”, los “Sprints”, “Sprints Reviews”, y artefactos como el “Sprint Backlog” y “Burn-down Charts”. (Urrutia, López, Martínez, & Corral, 2010).

Esta etapa se enfoca en el desarrollo del videojuego, utilizando como entrada los requerimientos de la fase de preproducción y de los posibles cambios o nuevas características que se puedan agregar durante las iteraciones.

Inicialmente en el primer “Sprint” o iteración, se realizan las llamadas “Sprint Huddle”, dentro de las cuales se analizan los requerimientos y se genera un “Sprint Backlog” el cual contiene las tareas que se necesita realizar para que se cumpla la primera meta a corto plazo. Posteriormente se generan los “Burn-down charts”, en estos se representa de forma visual el trabajo realizado y lo que falta por hacer, permitiendo una mejor organización evitando que el proceso se vuelva lento. Una vez que se cumplan los objetivos del primer “Sprint” se procede a realizar las pruebas alfa, en esta etapa de pruebas se analiza cada característica que se implementó durante la iteración para comprobar que se cumplan los objetivos requeridos y que no existan errores en el código del videojuego. Si se encuentran errores estos se documentan en los “Buglists”, para que el “Sprint” pueda terminar los errores deben ser corregidos.

Una vez corregidos los errores se llega a la versión beta del videojuego, dentro de la cual se realizan pruebas por miembros externos al grupo de desarrollo, en esta etapa se utilizan las pruebas Beta abiertas y cerradas.

Al término de las pruebas y corrección de los últimos errores se obtiene el videojuego final también conocido como Gold Master, el cual pasará a la última etapa llamada Postmortem. (Ver Figura 6.)

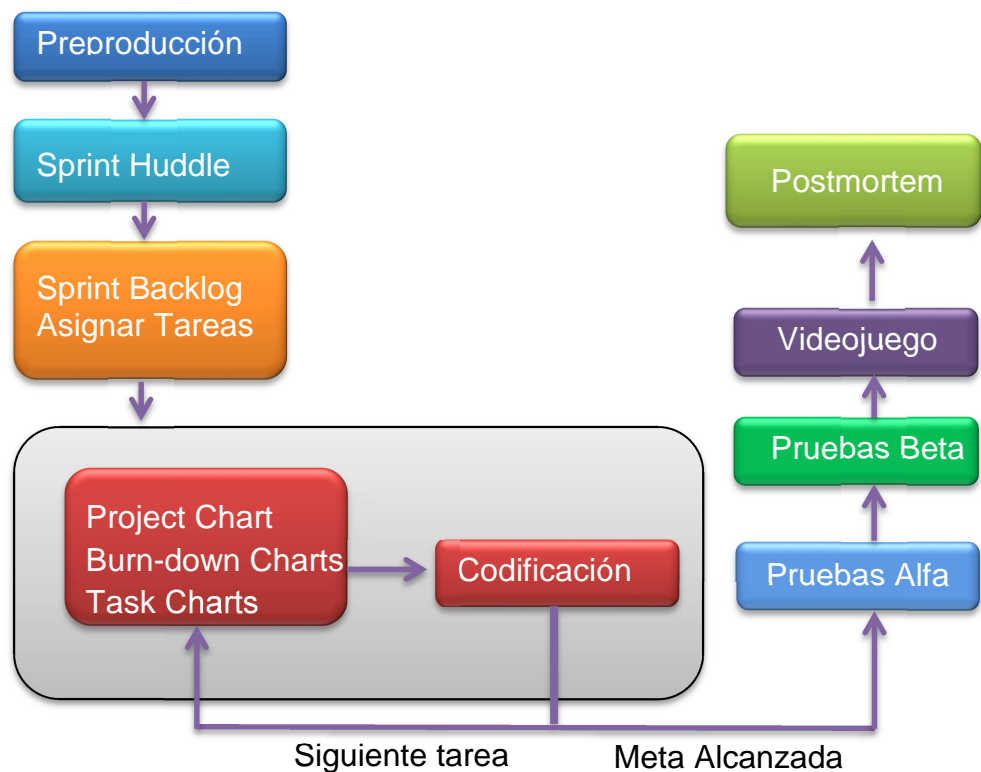


Figura 6. Fases de la etapa de Producción

### 2.1.4.3. Postmortem

Es la última fase de Huddle y dentro de la cual es necesario realizar una última reunión (End-game Huddle) para generar un reporte donde se el objetivo es describir detalladamente las actividades más y menos efectivas del proyecto con la finalidad de tomar en cuenta para un nuevo proyecto.

Una manera de realizar este reporte es respondiendo las siguientes preguntas: ¿Qué salió bien?, ¿Qué salió mal? Y ¿Qué obstáculos se presentaron? (Urrutia, López, Martínez, & Corral, 2010). (Ver Figura 7.)



Figura 7. Fases de la etapa Postmortem

## 2.1.5. Herramientas de desarrollo

### 2.1.5.1. HTML5

#### 2.1.5.1.1. Historia

La historia y evolución del estándar HTML5 proviene de las décadas de los 90, cuando el lenguaje HTML (HiperText Markup Language) se presentó como un proyecto de Internet y fueron desarrolladas las versiones 2.0, 2.3 y 4.0. (Lubbers, Albers , & Salim, 2001)

En 1999 la World Wide Web Consortium (WC3) toma el control de las especificaciones para el lenguaje. A su vez aparecen otros lenguajes como XML y XHTML que pretendían reemplazar al HTML; en el año 2004 se forma el Web Hypertext Application Working Group (WHATWG) y se crea las especificaciones del lenguaje HTML5, en los años siguientes esta especificación se continúa mejorando con la cooperación de los navegadores de Apple, Mozilla y Opera, finalmente en el año 2006 la W3C se involucra nuevamente dentro del proyecto y para el año 2008 se presenta el primer proyecto de la especificación del lenguaje HTML5. (Lubbers, Albers , & Salim, 2001), (Berjon, Leithead, Doyle Navara, O'Connor, & Pfeiffer, 2012)

### **2.1.5.1.2. Definición**

Versión 5 del lenguaje HTML, fue diseñada para reemplazar a su versión anterior HTML4, XHTML y al HTML DOM nivel 2, teniendo como principal objetivo ofrecer una excelente calidad una gran calidad dentro del contenido sin necesidad de plugins. (West, 2013)

### **2.1.5.1.3. Características**

A continuación se mencionan los principios básicos para los cuales se fundamentó la especificación de este lenguaje.

- Compatibilidad
- Utilidad
- Interoperabilidad
- Acceso universal

Con estos principios se pretende que las aplicaciones desarrolladas en este lenguaje puedan funcionar sin que existan cambios de comportamiento bruscos sino que todo sea fluido, tomando siempre como prioridad a los usuarios tratando que este lenguaje sea sumamente práctico para los desarrolladores así la posibilidad a que se produzcan errores durante la ejecución sean menores.

Por otra parte se busca que el acceso a las aplicaciones sea por cualquier tipo de usuarios sin importar sus incapacidades y también sin importar el tipo de dispositivo en el que se la ejecute.

Una de las características que posee HTML5 es la de no depender de plugins ya que provee soporte nativo a ciertas funcionalidades como audio y video por ejemplo. (Lubbers, Albers , & Salim, 2001)



Dentro de las características que trae consigo HTML5 se encuentran las siguientes: (Lubbers, Albers , & Salim, 2001)

- Video Nativo
- Socktes nativos
- Canvas 2D y 3D
- Geolocalización
- Formularios
- MathMIL
- Microdata
- Server-Sent events
- Gráficos Vectoriales Redimensionables
- Almacenamiento Web
- Bases de datos Indexadas
- Aplicaciones Web Offline
- Drag and Drop
- XMLHttpRequest nivel 2

Con estas nuevas características de HTML5 se puede apuntar al desarrollo de aplicaciones mucho mejores y más dinámicas, por otro lado como es un lenguaje que no ha alcanzado su completa finalización se puede apreciar que está en continuo evolución y desarrollo, lo que podría generar algún tipo de problemas para los desarrolladores.

## **2.1.5.2. JavaScript**

### **2.1.5.2.1. Definición**

Es un lenguaje de programación interpretado esto quiere decir que no necesita ser compilado solamente necesita un navegador Web para ser ejecutado y probado, este lenguaje permite desarrollar páginas web dinámicas las cuales se caracterizan por incorporar efectos, animaciones, interacción ante eventos entre otras características. (Eguíluz Pérez, 2009)

Actualmente JavaScript es uno de los lenguajes más utilizados dentro del desarrollo de aplicaciones web, sistemas operativos como Windows 8, Firefox OS, Chrome Os han adoptado los estándares web abiertos como la capa de presentación para aplicaciones nativas. (Elliot, 2014)

Tomando estas definiciones se puede concluir que el lenguaje de programación JavaScript permite hacer páginas web interactivas y de una manera más simple en el momento de la codificación ya que para su desarrollo y pruebas no se necesita de una tercera herramienta.

### **2.1.5.2.2. Características**

Dentro de este lenguaje se puede apreciar ciertas características que se irán describiendo, las cuales permiten un mejor desempeño de este lenguaje en su ejecución y a su vez en la etapa de codificación.

Si se habla de rendimiento se caracteriza en la realización de ciertas cosas de una forma más rápida debido a que JavaScript se ejecuta como código nativo así como manejo de eventos de una manera organizada y eficiente.

Su sintaxis es muy fácil de aprender debido a su familiaridad con la sintaxis de otros lenguajes de programación como Java, PHP y C++. (Elliot, 2014)

### 2.1.5.3. CSS

#### 2.1.5.3.1. Definición

Las Hojas de Estilo en Cascada (CSS) es un lenguaje que define estilos que pueden ser aplicados a una página HTML, donde el HTML se encarga de la estructura de la página Web y el CSS de la parte de la presentación. La especificación de este lenguaje fue propuesta por la W3C su última versión es la CSS3 la cual es una mejora de sus versiones anteriores CSS1 y CSS2. (Lunn, 2013)

Al observar la Figura 9. se puede observar que el texto, los botones, los cuadros de texto son simplemente HTML. Por otra parte en la Figura 8. los botones de texto, etc combinan el html con el CSS y se obtiene una mejor presentación hacia el usuario.



Figura 8. Ejemplo de Botones y cajas de texto sin CSS

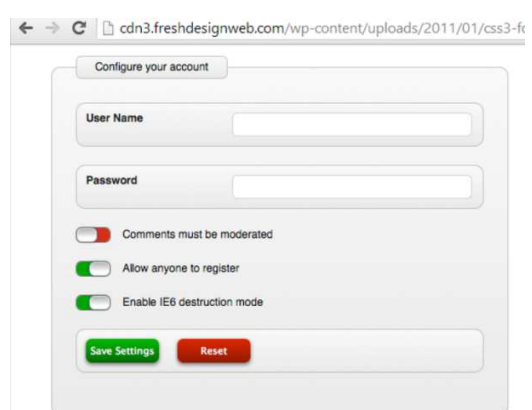


Figura 9. Ejemplo de botones y cajas de texto con CSS.

Recuperado de: <http://cdn3.freshdesignweb.com>.

Una gran ventaja de la utilización de CSS es la facilidad con la cual se puede hacer cambios dentro de la presentación de la página Web solo es necesario cambiar la hoja de estilo y esto se aplicara en todo lugar que se lo esté utilizando, sin CSS hacer un cambio que afecte a todas las páginas de un sitio Web podía tornarse algo tedioso y llevar mucho tiempo, especialmente cuando se trata de sitios con una gran cantidad de páginas Web. (Lunn, 2013)

Tomando en cuenta lo visto anteriormente se puede concluir que al utilizar CSS dentro las páginas Web, permite que exista una mejor calidad visual para el usuario final, teniendo paginas mas atractivas hacia la vista.

## **2.1.6.Educación y videojuegos**

### **2.1.6.1. Teoría de Piaget.**

Jean Piaget propone un sistema teórico que a lo largo de los años ha sido fundamental para el estudio del desarrollo humano, teniendo un gran impacto en el avance de la psicología evolutiva. Piaget creo una base primordial de referencia para nuevas investigaciones en varios ámbitos. (Fuentes, 2012)

La teoría de Piaget, trata sobre el desarrollo de las funciones cognitivas y tiene los siguientes principios fundamentales (Piaget, 1981):

- **El constructivismo**

Según el autor el conocimiento está unido a la acción, a las operaciones, a las transformaciones que el sujeto realiza sobre el mundo externo. Así, la evolución de la inteligencia del niño, el conocimiento resulta de la interacción entre sujeto y objeto.

Piaget establece dos procesos biológicos que se complementan, estos procesos simultáneos son los de asimilación y acomodación. “La organización y la adaptación con sus dos polos de asimilación y de acomodación, constituyen el funcionamiento que es permanente y común a la vida, pero que es capaz de crear formas o estructuras variadas” (Thong, T. (1981). Los estadios del niño en la Psicología Evolutiva: Los sistemas de Piaget. Wallon. Gesell y Freud. Madrid: Pablo del Río). “La adaptación es el equilibrio entre el organismo y el medio” (Piaget, J. (1990). El nacimiento de la inteligencia. Barcelona: Crítica.).

- **Etapas del desarrollo**

Como se ha descrito el desarrollo intelectual se relaciona con el desarrollo biológico. El autor describe una aparición progresiva de diferentes etapas durante la evolución de la inteligencia. La teoría de Piaget descubre los estadios de desarrollo cognitivo desde la infancia a la adolescencia. Explica cómo a partir de los reflejos innatos se desarrollan las estructuras psicológicas, además en la infancia se establecen esquemas de conducta, los cuales se internalizan como modelos de pensamiento en el segundo año de vida, en la infancia y la adolescencia se crean complejas estructuras intelectuales que caracterizan la vida adulta. (Piaget, 1981). La Tabla1. muestra los periodos descritos según Piaget.

**Tabla 1. Estadios del desarrollo según Piaget.**

Edad	Periodo	Estadio
<b>0-2 años</b>	<b>Sensomotor</b> En este periodo existe una inteligencia práctica basada en las acciones. La conducta es esencialmente motora. El pensamiento implica ver, oír, moverse, tocar, saborear en forma sucesiva. No hay representación interna de los	<ul style="list-style-type: none"> <li>• Estadio de los mecanismos reflejos congénitos. 0-1mes</li> <li>• Estadio de las reacciones circulares primarias. 1-4 meses.</li> </ul>

**CONTINÚA →**

	<p>acontecimientos externos, ni piensa mediante conceptos. El niño aprende a partir de experiencia corporal, primero con su cuerpo y posteriormente con el medio exterior.</p>	<ul style="list-style-type: none"> <li>•Estadio de las reacciones circulares secundarias 4-8 meses</li> <li>• Estadio de la coordinación de los esquemas de conducta previos. 8-12 meses</li> <li>•Estadio de los nuevos descubrimientos por experimentación. 12-18 meses</li> <li>• Estadio de las nuevas representaciones mentales. 18-24 meses</li> </ul>
<p><b>2-7 años</b></p>	<p><b>Pre operacional</b></p> <p>Es la etapa del pensamiento y la del lenguaje que gradúa su capacidad de pensar simbólicamente, imita objetos de conducta, juegos simbólicos, dibujos, imágenes mentales y el desarrollo del lenguaje hablado. Aparece la adquisición de la representación mental, pero para poder conocer necesita continuar actuando físicamente y de esta forma consigue soluciones. Pero sus razonamientos no son lógicos. El niño enfoca o centra su atención en la dimensión de la altura.</p>	<p>Se divide en dos subestadios:</p> <ul style="list-style-type: none"> <li>-Período pre conceptual: de los 2 a los 4 años.</li> <li>- Período intuitivo: de los 4 a los 7 años.</li> </ul>
<p><b>7-12 años</b></p>	<p><b>Operaciones concretas</b></p> <p>Los procesos de razonamiento se vuelen lógicos y pueden ser aplicados a problemas concretos o reales. En el aspecto social, el niño se convierte en un ser verdaderamente social y en esta etapa aparecen los esquemas lógicos de seriación, ordenamiento mental de conjuntos y clasificación de los conceptos de casualidad, espacio, tiempo y velocidad. El niño puede llevar a cabo operaciones de primer grado sobre objetos. Existe la noción de conservación de la materia, peso y volumen.</p>	
<p><b>12 años</b></p> <p>- <b>madurez</b></p>	<p><b>Operaciones formales</b></p> <p>En esta etapa el adolescente logra la abstracción sobre conocimientos concretos observados que le permiten emplear el razonamiento lógico inductivo y deductivo. Desarrolla sentimientos idealistas y formación continua de la personalidad, hay un mayor desarrollo de los conceptos morales. El razonamiento es hipotético deductivo.</p>	

Fuente: Elaboración propia. Basado de: Piaget J. La teoría de Piaget Journal: Infancia y Aprendizaje, January 1981.

### 2.1.7. Videojuegos en la Educación

Existen defensores y detractores de los videojuegos como herramientas para el aprendizaje y por el momento no existen estudios cien por ciento efectivos para aprobar o desaprobar esta nueva herramienta educativa.

Lo que sí se puede analizar son los principios que debe cumplir un videojuego enfocado a la educación para servir como complemento para el aprendizaje de los niños.

Para que un videojuego sea efectivo éste necesita que el jugador se sienta atraído hacia él, no sea aburrido para el jugador y que posea claros objetivos. Se puede observar varios atributos de un videojuego que se encuentran relacionados con el modo de aprendizaje de las personas, entre los que se puede observar los siguientes: (Oblinger, 2006).

- **Social.** Debido al gran impacto social de ciertos videojuegos, estos pueden obtener mejores resultados dentro de la educación.
- **Investigación.** Para dominar un videojuego, es necesario que el jugador busque nueva información que sea de ayuda cada situación nueva del juego.
- **Resolución de problemas.** Muchas veces en gran cantidad de videojuegos es necesario tomar decisiones rápidas para resolver problemas específicos. Así el jugador deberá buscar diferente tipo de técnicas las cuales deberá aplicar para lograr ganar el juego.
- **Transferir.** Debe existir una conexión entre el aprendizaje de la vida o de la escuela con el aprendizaje del videojuego.
- **Experimental.** Cada vez que se juega un videojuego los jugadores aprenden de sus resultados.

### 2.1.7.1. Aprendizaje basado en juegos Digitales (DGBL)

La idea de usar videojuegos como sustitución a la educación frente a frente es una idea errónea, lo que sí se puede buscar es la integración de los videojuegos y la educación, mezclándolos con ciertas actividades específicas para buscar un objetivo único. Para integrar los videojuegos dentro de la educación se debe entender cuáles son los tipos de juegos que promueven un resultado de aprendizaje deseado como por ejemplo: (Oblinger, 2006)

- **Juegos de cartas.** promueven la memoria, reconocimiento de patrones y agrupación de conceptos.
- **Juegos de estilo “Jeopardy”.** Promueve la definición rápida de hechos y conceptos concretos.
- **Juegos de estilo “Arcade”.** Promueven la velocidad de respuesta, el procesamiento visual y automático de hechos.

## 2.2. Metodología OOHDM

La metodología OOHDM (Método de Diseño Hipermedia Orientado a Objetos), desarrollada por Daniel Schwabe y Gustavo Rossi, está enfocada al desarrollo de aplicaciones hipermedia utilizando modelos; la cual consta de 5 etapas: obtención de requerimientos, Diseño Conceptual, Diseño Navegacional, Diseño Abstracto de Interfaz e Implementación. (Rossi, Pastor, Schwabe, & Olsina, Web Engineering: Modelling and Implementing Web Applications, 2008)



### **2.2.1. Obtención de requerimientos**

Es la primera de las cinco etapas, y para la obtención de los requerimientos se utilizan los diagramas de casos de uso, con el fin de que los requerimientos sean lo más claros posibles estos son diseñados por escenarios. Esto quiere decir que primero se debe identificar los actores que están involucrados y cuáles van a ser sus tareas, posterior a este paso se se realizan los casos de uso, así se puede observar los diferentes flujos de información entre el usuario y el sistema.

### **2.2.2. Diseño Conceptual**

En esta etapa se utiliza una notación similar a la de UML, para realizar un diagrama conceptual en el cual se representa subsistemas, clases y las relaciones existentes entre ellos. Se enfoca en el dominio semántico de la aplicación, dejando de lado a los tipos de usuarios y sus tareas (Rossi, Pastor, Schwabe, & Olsina, Web Engineering: Modelling and Implementing Web Applications, 2008).

### **2.2.3. Diseño Navegacional**

En esta etapa la aplicación es vista como una vista navegacional por encima del modelo conceptual, por lo cual los objetos utilizados no son objetos conceptuales sino que dependen de los diferentes usuarios y sus tareas (Rossi, Pastor, Schwabe, & Olsina, Web Engineering: Modelling and Implementing Web Applications, 2008)

Eso quiere decir que tiene como principal objetivo demostrar los posibles caminos que la aplicación puede tomar basándose en las tareas realizadas por los usuarios.

#### **2.2.4. Diseño de Interfaz Abstracta**

En el diseño de Interfaz Abstracta se especifica las interfaces con las que el usuario va a interactuar esto incluye las diferentes maneras en las cuales los objetos navegacionales (interfaces, contenido multimedia) se activarán, cuales deberán ser sincronizadas y a su vez cuales serán reemplazadas. (Schwabe & Rossi, 1998, June).

Dentro de este contexto lo que pretende esta etapa es identificar como el contenido de una aplicación Web se comportará ante un evento externo o a su vez por un evento ocasionado por el usuario, así se puede observar que muchas veces los cambios o transformaciones de las interfaces no están directamente conectados con los estados navegacionales de la aplicación.

#### **2.2.5. Implementación**

En esta etapa se deberá implementar el diseño previamente desarrollado ya que estos fueron diseñados independientemente de la plataforma en la cual se va a realizar la implementación (Schwabe & Rossi, 1998, June).

Aquí se tendrá que elegir las herramientas para el desarrollo ya que al llegar a esta etapa el desarrollador sabe cómo va a funcionar la aplicación, también se tiene bien identificado los actores y sus roles. Finalmente se utiliza un lenguaje de programación o herramienta de software que permita implementar lo diseñado anteriormente.

## CAPÍTULO 3: ANÁLISIS Y DISEÑO

En este capítulo se presenta la documentación propuesta por el proceso de desarrollo de software para videojuegos Huddle.

### 3.1 Preproducción

Dentro de esta etapa se realizará el documento de diseño con el cual se podrá obtener los requerimientos para posteriormente realizar los diagramas correspondientes de la metodología OOADM.

#### 3.1.1 Documento de Diseño

En este documento se detalla la temática central del videojuego, sus características principales y personajes. (Ver Tabla 2.)

Tabla 2. Documento de Diseño

Plantilla del Documento de Diseño	
CAMPO	DESCRIPCIÓN
	CONCEPTO
<b>Título</b>	Recoge Letras
<b>Estudio/Diseñadores</b>	David Reyes
<b>Género</b>	Educativo
<b>Plataforma</b>	Computador, navegador
<b>Versión</b>	0
<b>Sinopsis de Usabilidad y Contenido</b>	El videojuego se basa en recoger letras para formar una palabra dada.
<b>Licencia</b>	Juego original.
<b>Mecánica</b>	El jugador se encuentra dentro del mar, este deberá nadar hasta las letras para recolectarlas y formar la palabra deseada.
<b>Tecnología</b>	Turbulenz Engine, JavaScript, HTML5, Audacity.
<b>Público</b>	niños entre los 7-11 años de edad
HISTORIAL DE VERSIONES	

CONTINÚA →

---

**Versión 0**
**VISIÓN GENERAL DEL JUEGO**

El jugador se encuentra en un mundo submarino dentro del cual deberá recoger las letras pertenecientes a una palabra dada.

**MECÁNICA DEL JUEGO**

<b>Cámara</b>	Perspectiva 2D
<b>Periféricos</b>	Teclado, mouse.
<b>Controles</b>	Se utiliza las flechas del teclado para los movimientos, y el mouse para los botones.
<b>Puntaje</b>	NA
<b>Guardar/Cargar</b>	NA

**NIVELES**

<b>Título del nivel</b>	Fácil, Medio, Difícil
<b>Encuentro</b>	Se encuentran en la pantalla principal
<b>Descripción</b>	Letras formadas aleatoriamente van cayendo dentro del mundo, el jugador deberá evitar las letras que no pertenezcan a la palabra dada, y recolectar las correctas
<b>Objetivos</b>	Para terminar el nivel el jugador deberá completar 4 palabras.
<b>Progreso</b>	Cuando el jugar gana el nivel pasa a un nuevo nivel con una dificultad mayor
<b>Enemigos</b>	NA
<b>Ítems</b>	NA
<b>Personajes</b>	Diver Boy
<b>Música y Efectos de Sonido</b>	Música ambiental de mar, efectos de sonido cuando acierta o cuando se equivoca
<b>Referencias BGM y SFX</b>	NA

**PROGRESO DEL JUEGO**

**El videojuego consta de tres niveles de dificultad, cada uno de ellos tiene 4 palabras de diferente longitud**

**PERSONAJES**

<b>Nombre del personaje</b>	Diver Boy
<b>Descripción</b>	Es un dibujo animado de un hombre nadando bajo el mar.
<b>Concepto</b>	primer nivel
<b>Encuentro</b>	primer nivel
<b>Habilidades</b>	moverse y recolectar letras

**MÚSICA Y SONIDOS**
**Creative Commons**
**MIEMBROS DEL EQUIPO**


---

**CONTINÚA →**

<b>David Reyes</b>	
DETALLES DE LA PRODUCCIÓN	
<b>Fecha de inicio</b>	1/4/2015
<b>fecha de terminación</b>	19/6/2015

### 3.1.2 Especificación de Requerimientos

#### 3.1.2.1 Identificación de Roles y Tareas

- **Roles**

**JUGADOR:** Es el usuario que puede interactuar con la aplicación. Es capaz de modificar el juego, iniciar una partida, pausar una partida, salir de la partida.

- **Tareas**

##### **Jugador**

- Iniciar el juego
- Pausar el juego
- Leer acerca de
- Consultar controles
- Silenciar juego
- Salir del juego

#### 3.1.2.2 Especificación de Escenarios

##### **Rol Jugador**

- **Iniciar el juego:** El jugador podrá iniciar el juego mediante la pantalla de inicio.
- **Pausar el juego:** Dentro del juego el jugador podrá detener el juego para luego reanudarlo.
- **Leer el acerca de:** El jugador podrá leer los credits del juego mediante la pantalla inicial.

- **Consultar Controles:** El jugador podrá consultar los controles del juego y cómo jugarlo dentro de la pantalla inicial.
- **Silenciar Juego:** El jugador podrá activar o desactivar el sonido del juego.
- **Salir del juego:** El jugador podrá salir del juego en cualquier momento perdiendo lo avanzado.

### 3.1.2.3. Especificación de casos de uso por Actor

- **Actor: Jugador**
  - JUG-RECOGELETRAS-01: Iniciar Juego
  - JUG-RECOGELETRAS-02: Pausar Juego
  - JUG-RECOGELETRAS-03: Leer Acerca de
  - JUG-RECOGELETRAS-04: Consultar Controles
  - JUG-RECOGELETRAS-05: Silenciar juego
  - JUG-RECOGELETRAS-06 Salir del juego

La Figura 10. Muestra los casos de uso correspondientes al videojuego.

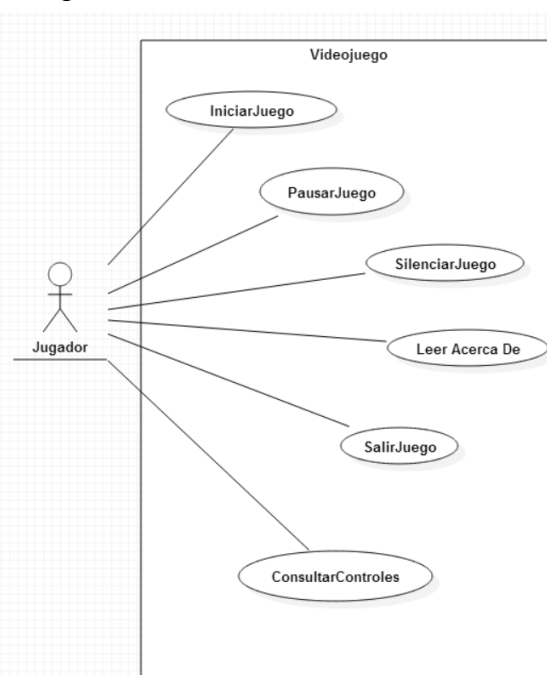


Figura 10. Casos de uso del Videojuego "Recoge letras"

- **Especificación de casos de uso**

En la tabla 3 hasta la tabla 8 se detallan las especificaciones de caso de uso del actor Jugador del videojuego RecogeLetras.

Tabla 3. Caso de uso Iniciar videojuego

<b>JUG-RECOGELETRAS-01: Iniciar Juego</b>	
Resumen:	Proceso en el cual el usuario puede iniciar la partida
Prioridad	Escencial
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario inicia la partida en la pantalla inicial.</li> <li>2. El usuario da click en jugar</li> <li>3. El usuario inicia la partida</li> </ol>
Pre-condiciones	Ninguna

Tabla 4. Caso de uso Pausar videojuego

<b>JUG-RECOGELETRAS-02: Pausar Juego</b>	
Resumen:	Proceso en el cual el usuario puede Pausar y reanudar el juego
Prioridad	Opcional
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario da click en el botón de pausa.</li> <li>2. El juego se detiene.</li> </ol>
Pre-condiciones	Iniciar Partida

Tabla 5. Caso de uso Leer Acerca de

<b>JUG-RECOGELETRAS-03: Leer acerca de</b>	
Resumen:	Proceso en el cual el usuario puede leer los créditos del juego.
Prioridad	Opcional
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario da click en el botón Acerca de.</li> <li>2. El usuario puede leer los créditos del videojuego.</li> </ol>
Pre-condiciones	Ninguna

Tabla 6. Caso de uso Consultar controles

<b>JUG-RECOGELETRAS-04: Consultar controles</b>	
Resumen:	Proceso en el cual el usuario puede consultar los controles del juego.
Prioridad	Opcional
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario da click en el botón controles</li> <li>2. El usuario observa cuales son los controles del juego</li> </ol>
Pre-condiciones	Ninguna



Tabla 7. Caso de uso Silenciar Videojuego

<b>JUG-RECOGELETRAS-05: Silenciar Juego</b>	
Resumen:	Proceso en el cual el usuario puede activar o desactivar el sonido del juego
Prioridad	Opcional
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario da click en el botón del sonido</li> <li>2. El sonido se desactiva</li> </ol>
Pre-condiciones	Ninguna

Tabla 8. Caso de uso Salir del Videojuego

<b>JUG-RECOGELETRAS-06: Salir del juego</b>	
Resumen:	Proceso en el cual el usuario puede salir de la partida
Prioridad	Opcional
Actores Directos:	Jugador
Escenarios	
Tipo de escenario	Descripción
Principal	<ol style="list-style-type: none"> <li>1. El usuario da click en el botón de pausa.</li> <li>2. El usuario da click en el botón salir</li> <li>3. El usuario sale de la partida</li> </ol>
Pre-condiciones	Iniciar Juego, Pausar Juego

### 3.1.3. Diseño Conceptual

Debido a que el lenguaje de programación JavaScript no es un lenguaje orientado a objetos, se puede simular a estos mismos mediante prototipos (prototypes). La Figura 11. muestra el diagrama de clases del videojuego Recoge Letras.

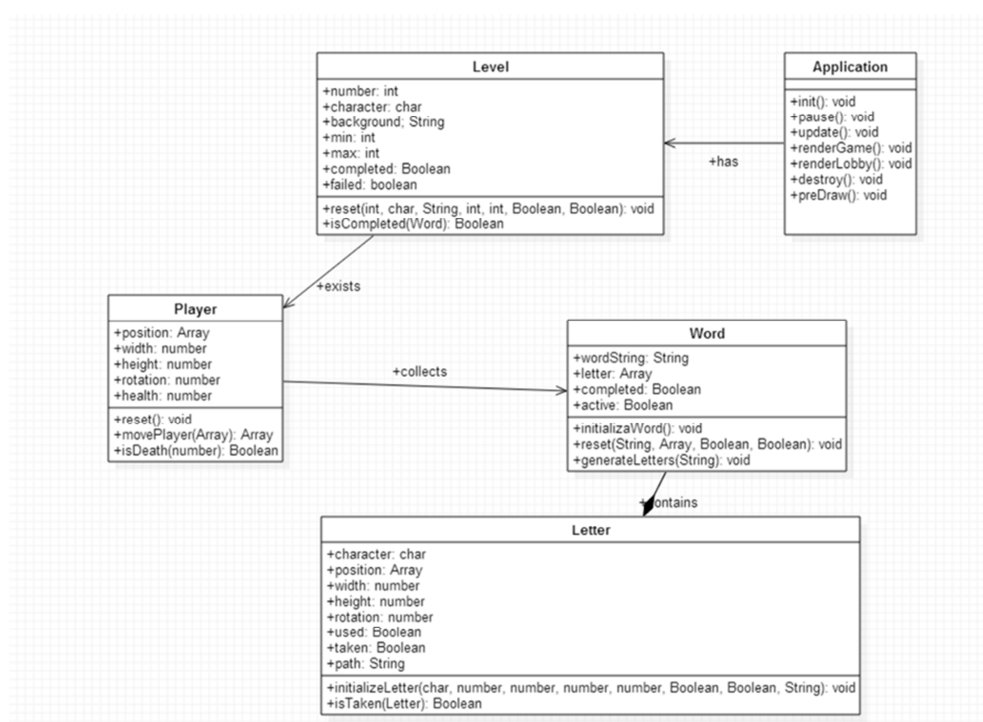


Figura 11. Diagrama de Clases del videojuego "Recoge Letras"

### 3.1.3. Diseño Navegacional

El sistema posee los siguientes objetos y contextos navegacionales. (Ver Figura 12.)

#### 3.1.3.1. Objetos Navegacionales

- Pantalla Inicial
- Pantalla de Acerca de
- Pantalla de Pausa

- Pantalla de Controles
- Pantalla de Juego
- Pantalla de Logro Completado
- Pantalla de Fin de Juego.

### 3.1.3.2. Contextos Navegacionales

- Iniciar partida
- Consultar controles del juego
- Pausar juego
- Terminar juego

### 3.1.3.3. Esquema Navegacional

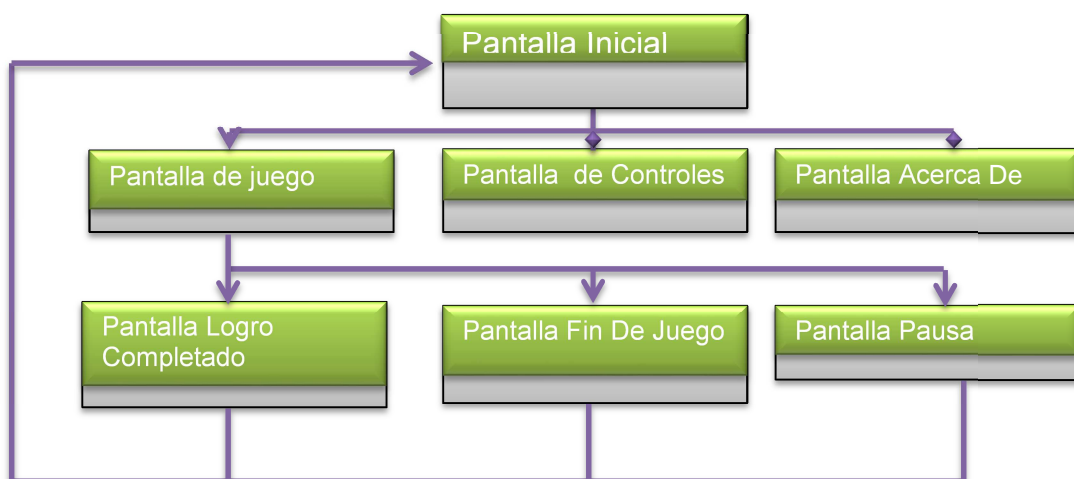


Figura 12. Esquema navegacional del videojuego "Recoge Letras"

### 3.1.3 Diseño de Interfaz Abstracta

A continuación se detallan las interfaces con las que el usuario interactuará y su forma de aparición dentro del juego. Las mismas que se muestran dentro de las figuras 13,14,15,16,17,18,19.

- Pantalla Inicial

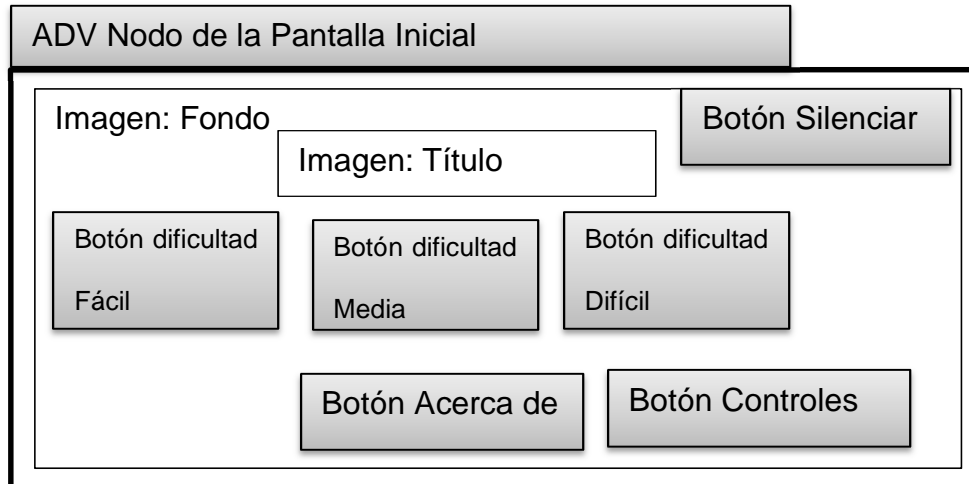


Figura 13. ADV Pantalla Inicial

- Pantalla Acerca de

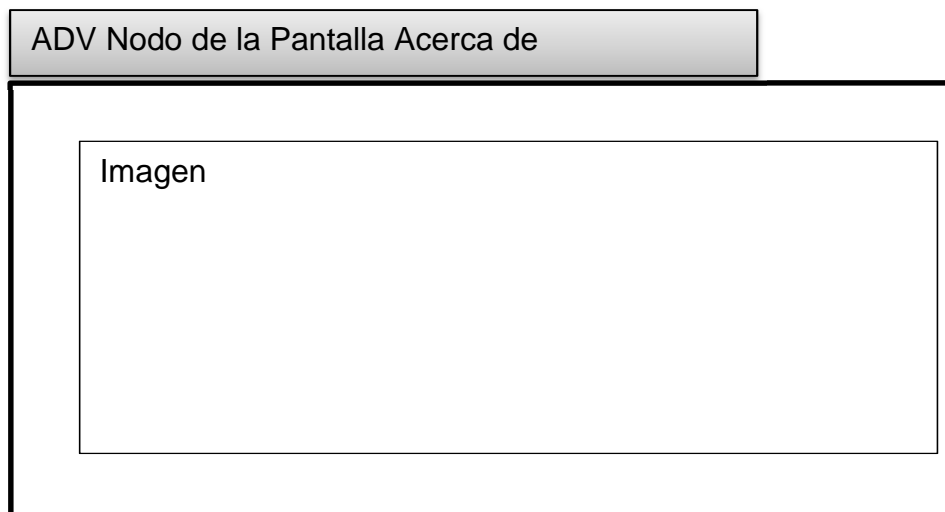


Figura 14. ADV Pantalla Acerca de

- Pantalla de Pausa

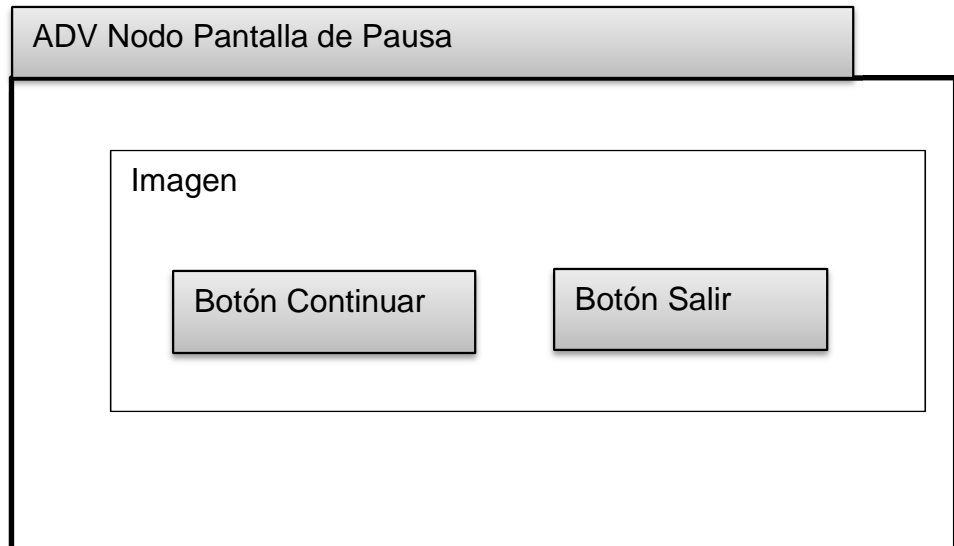


Figura 15. ADV Pantalla Pausa

- Pantalla Controles

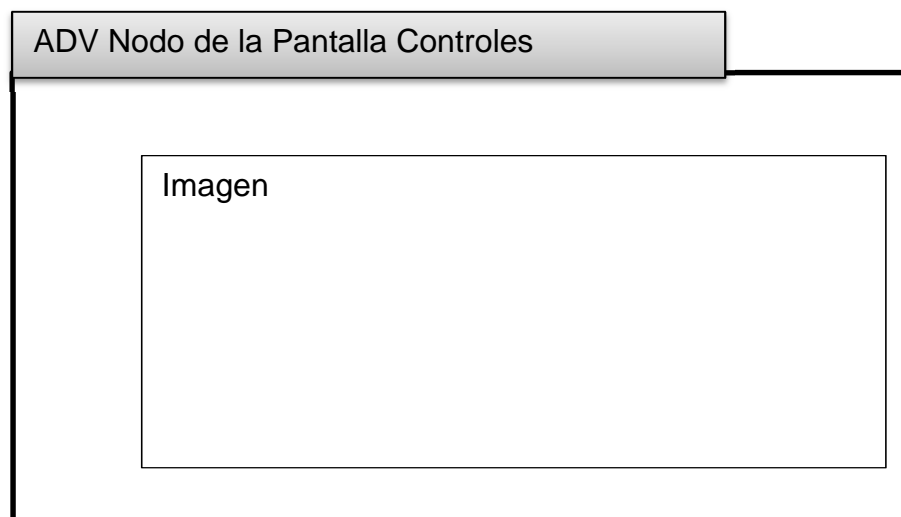


Figura 16. ADV Pantalla Controles

- Pantalla de Juego

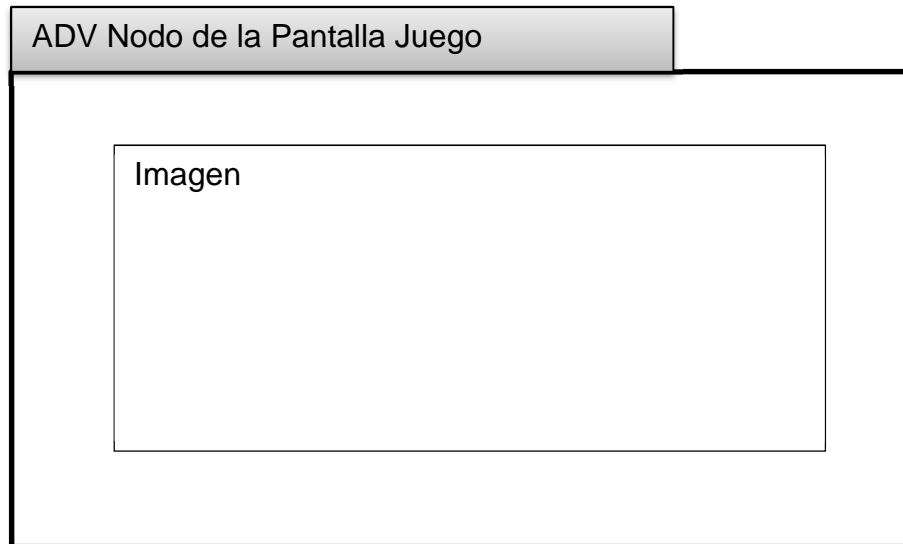


Figura 17. ADV Pantalla de Juego

- Pantalla de Logro Completado

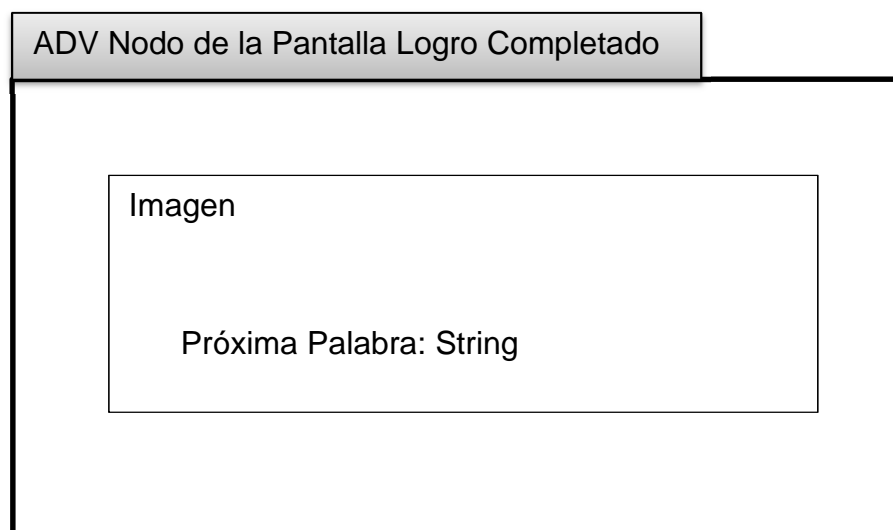


Figura 18. ADV Pantalla Logro Completado

- Pantalla de Fin de Juego.

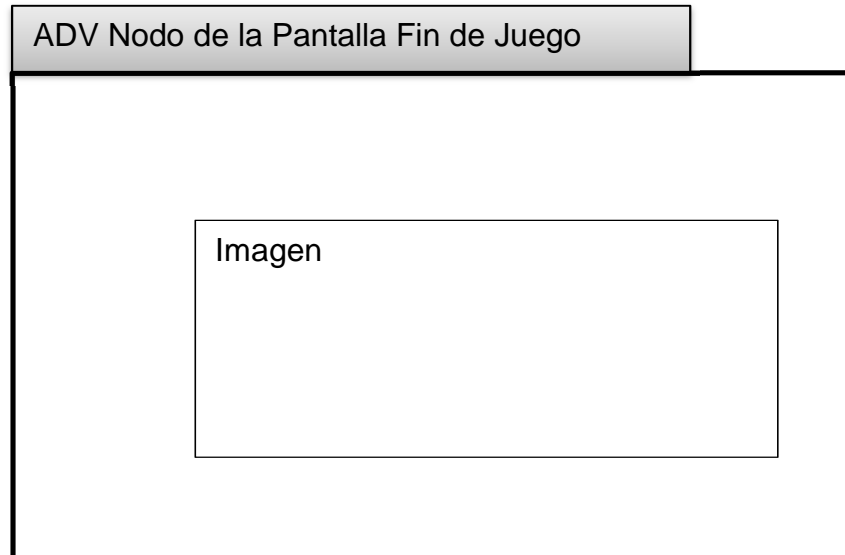


Figura 19. ADV Pantalla Fin de Juego

## **CAPÍTULO 4:**

# **IMPLEMENTACIÓN DEL VIDEOJUEGO**

Para la implementación del videojuego se utilizará un editor de texto para la codificación, para ejecutar el videojuego y realizar pruebas locales se necesita instalar el servidor local de Turbulenz, descargar el SDK correspondiente.

### **4.1 MOTOR DE VIDEOJUEGOS TURBULENZ**

#### **4.1.1 INSTALACIÓN**

El SDK de Turbulenz incluye todas las librerías, herramientas, documentación y ejemplos para comenzar a desarrollar una aplicación.

Se puede descargar los instaladores del TurbulenzEngine y del TurbulenzSDK desde la dirección Web: <https://hub.turbulenz.com>. Una vez descargados e instalados se obtienen las siguientes herramientas para el desarrollo:

##### **4.1.1.1 Enviroment**

Permite la ejecución de las herramientas provistas dentro del SDK. Se encuentra dentro de la dirección Start Menu -> Programs -> Turbulenz -> SDK 0.28.0 -> Run Environment 0.28.0. (Ver Figura 20.)



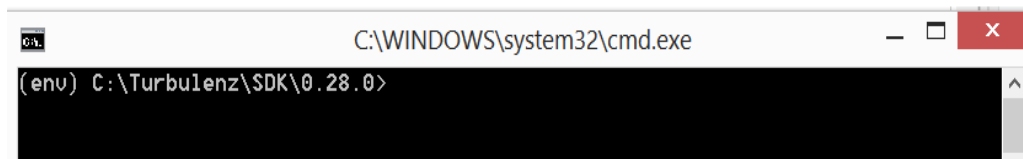


Figura 20. Consola de comandos Turbulenz

#### 4.1.1.2 Servidor Local

Permite probar los videojuegos dentro de una computadora o en una red local. Para que el servidor local se encuentre en funcionamiento es necesario activarlo desde la siguiente dirección: Start Menu -> Programs -> Turbulenz -> SDK 0.28.0 -> Run Local Server 0.28.0. (Ver Figura 21.)

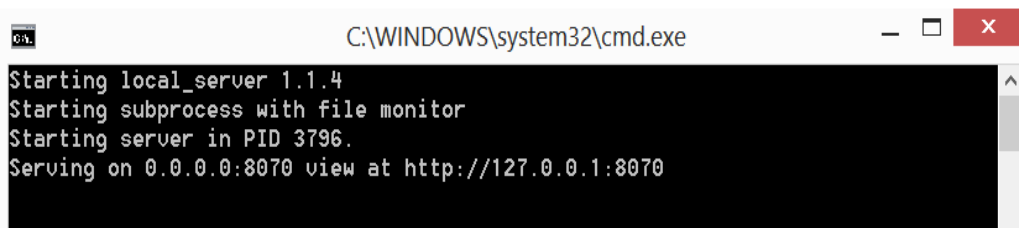
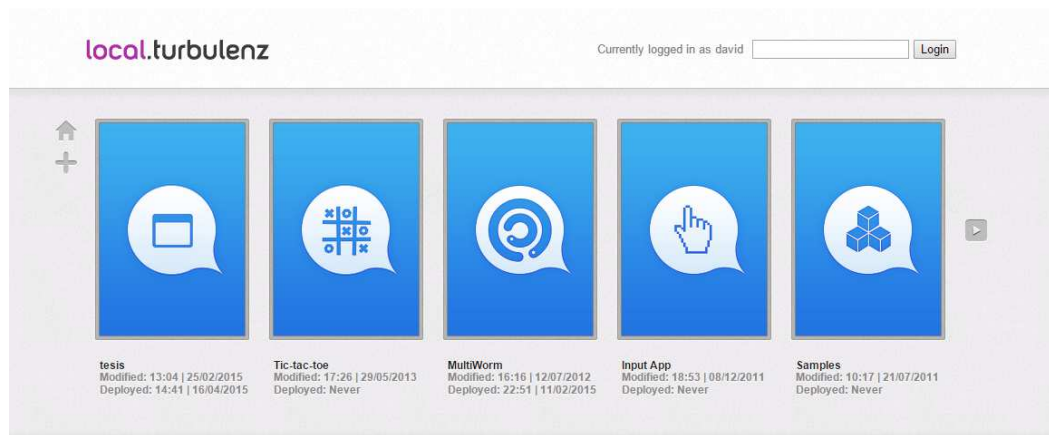


Figura 21. Servidor local de Turbulenz activado

Una vez que el servidor local este activado su puede probar las aplicaciones dentro del navegador utilizando la dirección <http://127.0.0.1:8070>. Aquí se encontraran todos nuestros juegos así como los ejemplos descargados dentro del SDK. (Ver Figura 22.)



**Figura 22. Página principal Servidor Local Turbulenz**

Cuando el videojuego haya sido terminado, Turbulenz ofrece una herramienta para subir automáticamente el juego online, esta herramienta se llama Turbulenz Hub.

#### **4.1.2 Turbulenz Hub**

Es parte fundamental para la publicación de los juegos online desarrollados con Turbulenz, esta herramienta facilita el alojamiento del videojuego online. La Figura 23. muestra cómo se encuentra conformada la plataforma Web de Turbulenz y cómo esta interactúa con el servidor local.

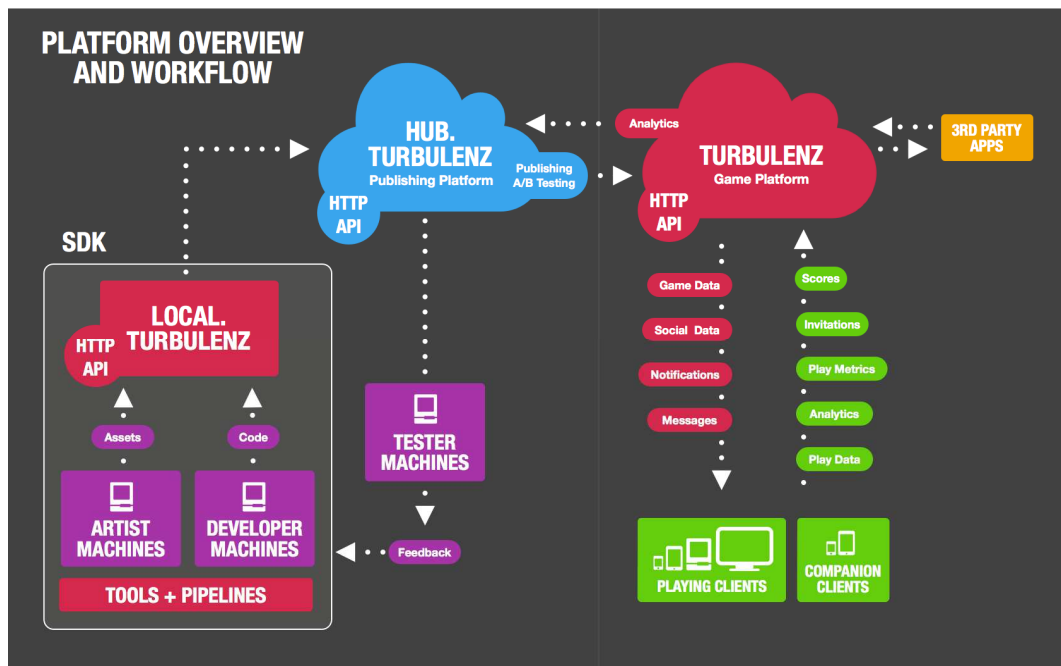


Figura 23. Plataforma Turbulenz.  
Recuperado de [http://docs.turbulenz.com/hub/user\\_guide.html#what-is-the-hub](http://docs.turbulenz.com/hub/user_guide.html#what-is-the-hub)

Esta plataforma permite a los desarrolladores:

- Probar los videojuegos remotamente
- Realizar pruebas en un ambiente de rendimiento real
- Opción de pre publicación del videojuego.
- Métricas como el número total de veces jugado el videojuego, duración promedio, usuarios activos semanales, diarios, entre otros.
- Control de versiones. (Ver Figura 24.)

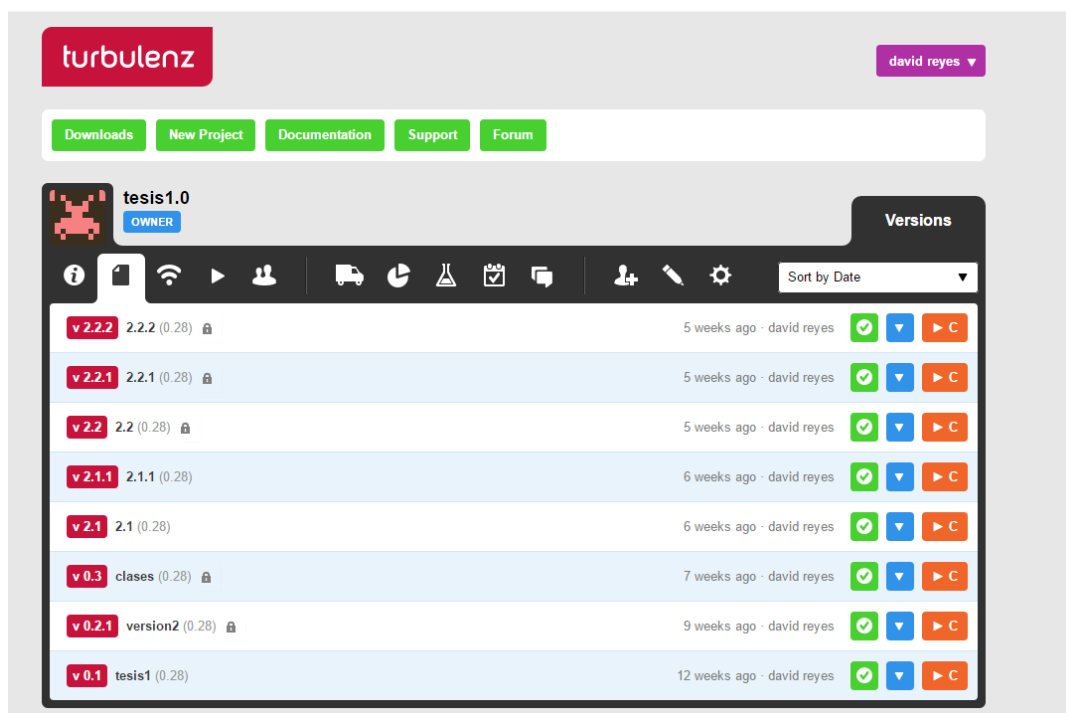


Figura 24. Plataforma de Turbulenz Hub (online)

## 4.2 Producción del Videojuego

Siguiendo las fases del proceso de desarrollo Huddle, se encuentra la etapa de producción dentro de la cual el objetivo es el desarrollo del videojuego. Aquí se encuentra el documento de producción propuesto por el proceso de desarrollo de videojuegos Huddle. (Ver Tabla 9.)

Tabla 9. Documento de Producción

Plantilla del Documento de Producción	
CAMPO	DESCRIPCIÓN
	DETALLES DEL PROYECTO
<b>Título</b>	Recoge Letras
<b>Estudio/Diseñadores</b>	David Reyes
<b>Género</b>	Educativo
<b>Plataforma</b>	Computador, navegador
<b>Fecha de Inicio</b>	1/4/2015
<b>Fecha de Término</b>	19/6/2015
<b>Planeado</b>	0

CONTINÚA →

<b>En desarrollo</b>	El videojuego se basa en recoger letras para formar una palabra dada.
<b>Terminado</b>	Juego original.
<b>SPRINT PLAN</b>	
<b>SprintID</b>	1
<b>Inicio</b>	1/4/2015
<b>Días</b>	5
<b>Fin</b>	6/4/2015
<b>INTERFAZ DE USUARIO</b>	
<b>%</b>	90
<b>FEATURE LOG</b>	
<b>FeatureID</b>	Perspectiva 2D
<b>Nombre</b>	Teclado, mouse.
<b>Estado</b>	El jugador se encuentra dentro del mar, este deberá nadar hasta las letras para recolectarlas y formar la palabra deseada.
<b>Días</b>	30
<b>SprintID</b>	NA
<b>SPRINT BACKLOG</b>	
<b>Sprint # Backlog</b>	1
<b>Días</b>	5
<b>Tareas</b>	mejorar gráficos
<b>Tendencia</b>	cambiar personaje principal
<b>Esfuerzo Restante</b>	Funcionalidad del juego, niveles, sonidos
<b>Tendencia Actual</b>	Completada.
<b>Progreso Ideal</b>	Acabar en 2 meses.
<b>Tareas restantes</b>	Funcionalidad del juego, niveles, sonidos
<b>Nombre de la Tarea</b>	lógica del videojuego
<b>FeatureID</b>	fácil
<b>Miembro</b>	David Reyes
<b>TASK CHART SPRINT #1</b>	
<b>dentro del primer Sprint se realizó la pantalla de inicio</b>	
<b>BUGLIST</b>	
<b>BugID</b>	1
<b>Descripción</b>	las imágenes no se cargan apropiadamente, aparece un rectángulo rosado
<b>Descripción Técnica</b>	El tiempo de carga de la imagen es muy lento.
<b>Autor</b>	David Reyes
<b>Estado</b>	Arreglado
<b>FeatureID</b>	1
<b>TASK SLIPS</b>	
<b>FeatureID</b>	2
<b>Nombre</b>	lógica del videojuego, mejorar personaje principal, mejorar pantallas

CONTINÚA →

<b>Tarea</b>	por realizar
<b>Miembro</b>	David Reyes
<b>Esfuerzo Estimado</b>	4 días
<b>Terminado</b>	80%
<b>Restante</b>	20%

Al final de esta etapa se obtuvo el videojuego final llamado “Recoge Letras”, el cual esta conformado por las siguientes pantallas.

- Pantalla Inicial del videojuego.(Ver Figura 25.)



Figura 25. Pantalla principal

- Pantalla Como Jugar(Controles). (Ver Figura 26.)

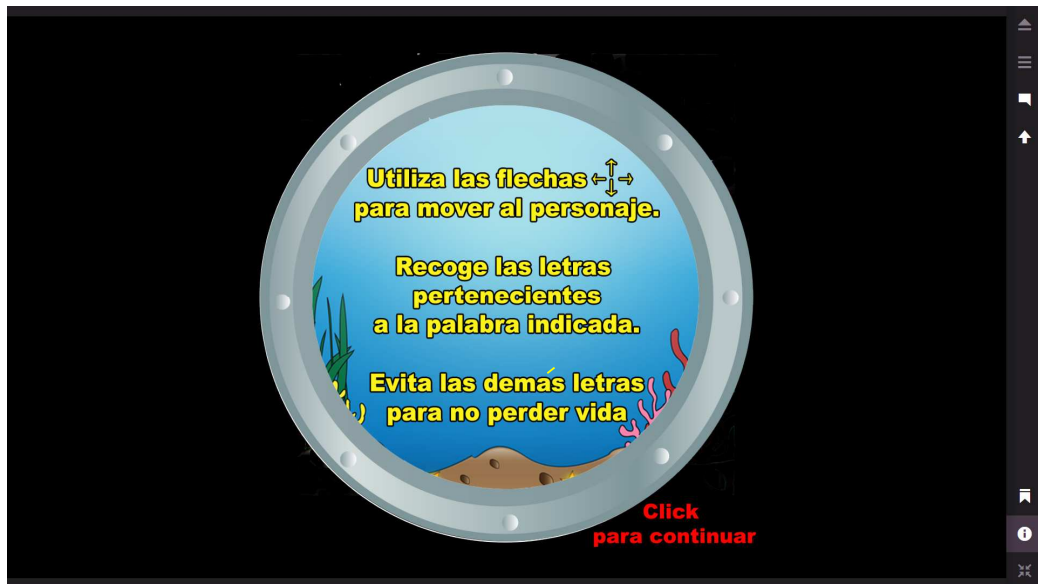


Figura 26. Pantalla Como Jugar

- Pantalla Créditos (Acerca de). (Ver Figura 27.)



Figura 27. Pantalla Créditos

- Mensaje Primer nivel y palabra a completar. (Ver Figura 28.)



Figura 28. Pantalla Inicio del Nivel

- Videojuego en ejecución. (Ver Figura 29.)



Figura 29. Pantalla del videojuego en Ejecución



- Pantalla del videojuego en Pausa. (Ver Figura 30.)



Figura 30. Pantalla Juego Pausado

- Pantalla de Intentalo de nuevo. (Ver Figura 31.)



Figura 31. Pantalla Fin del Juego

- Mensajes Ganadores. (Ver Figura 32,33,34.)

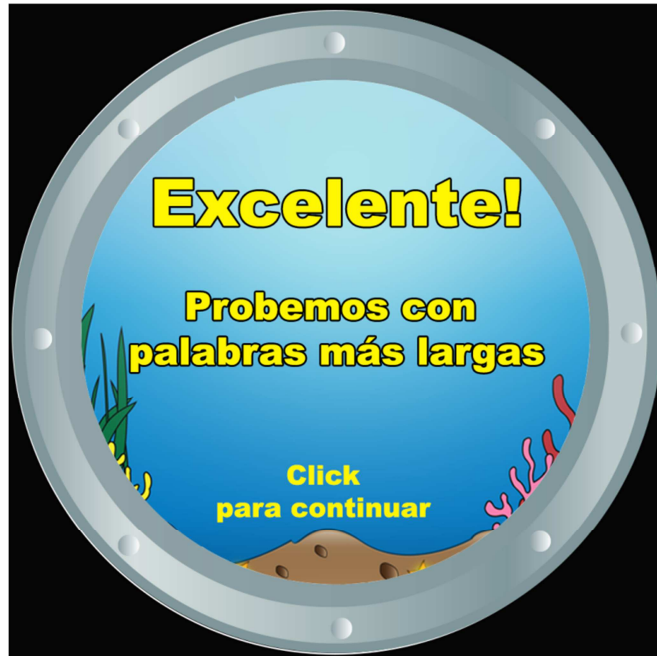


Figura 32. Pantalla de logro completado nivel 1



Figura 33. Pantalla de logro completado nivel 2



Figura 34. Pantalla de logro completado nivel 3

- Título del videojuego "Recoge Letras". (Ver Figura 35.)



Figura 35. Diseño del Título del Juego

- Diseño del personaje principal. (Ver Figura 36.)

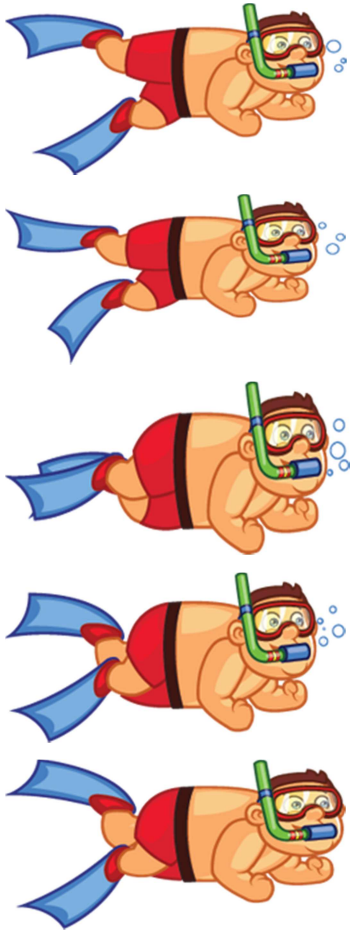


Figura 36. Diseño de la animación del personaje principal

## **CAPÍTULO 5:**

# **PRUEBAS Y EVALUACIÓN DE RESULTADOS**

### **5.1 Pruebas funcionales**

Las pruebas funcionales se realizaron en la escuela “Marqués de Selva Alegre” con una muestra de 66 estudiantes entre los 7 y 9 años de edad, de los cuales 33 son niñas y 33 son niños.

La muestra utilizada para la pruebas corresponde a la fórmula de cálculo siguiente:

$$n = \frac{N * \delta^2 * Z^2}{(N - 1)e^2 + \delta^2 * Z^2}$$

#### **5.1.1. Duración de la evaluación**

Para la realización de las pruebas en las escuelas se contó con quince minutos por alumno para que evalúe el juego y poder registrar los datos

#### **5.1.2. Infraestructura utilizada**

La infraestructura utilizada para las pruebas fue el laboratorio de computación del colegio “Marqués de Selva Alegre”, las cuales constaban con computadoras de escritorio con el sistema operativo Windows 7 instalado. El videojuego se ejecutó en diferentes navegadores sin inconveniente alguno.

### 5.1.3. Variables a medir

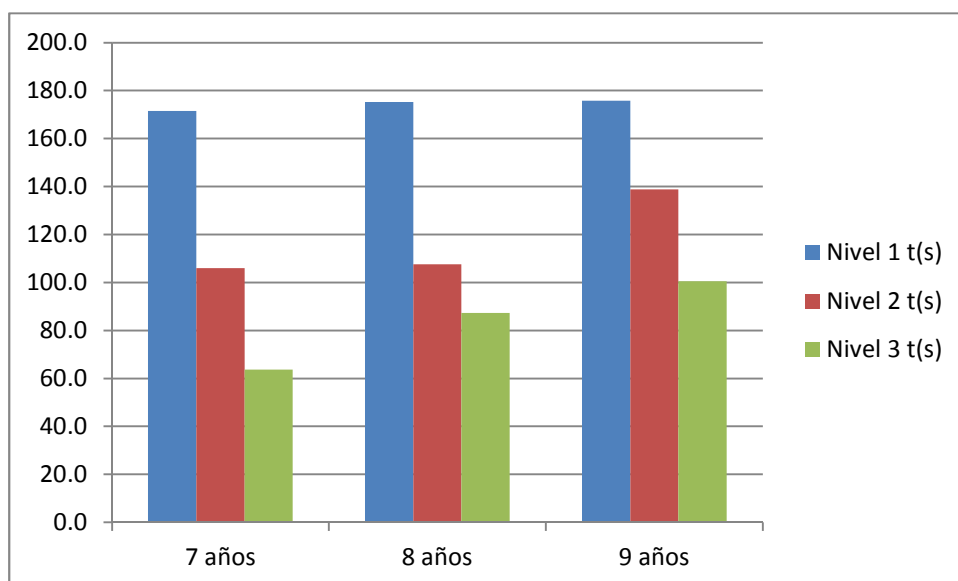
Las variables medidas fueron:

- Tiempo
- Número de aciertos

### 5.1.4. Resultados obtenidos

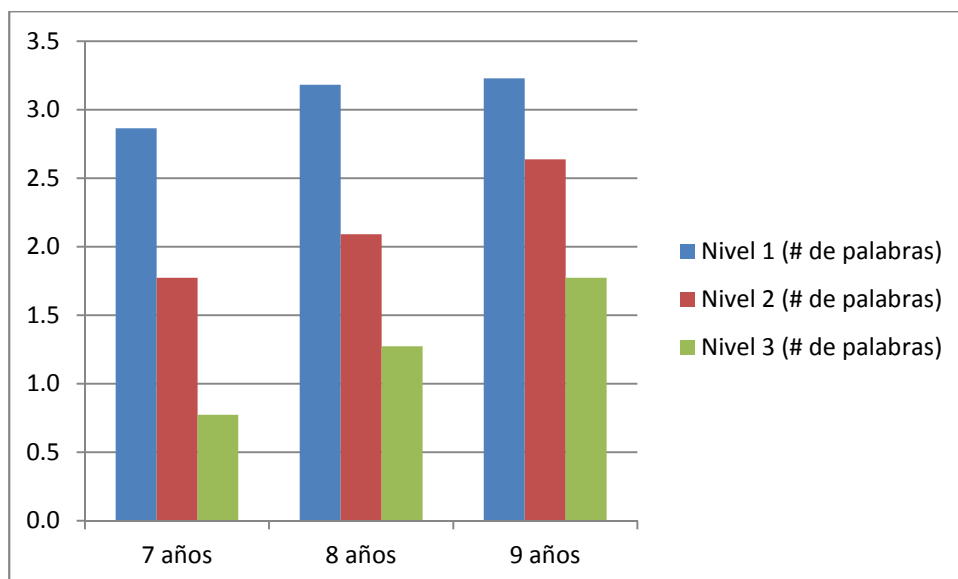
Los resultados obtenidos se clasificaron de acuerdo ha los siguientes parámetros :

- Tiempo promedio por edad y nivel. (Ver Figura 37).



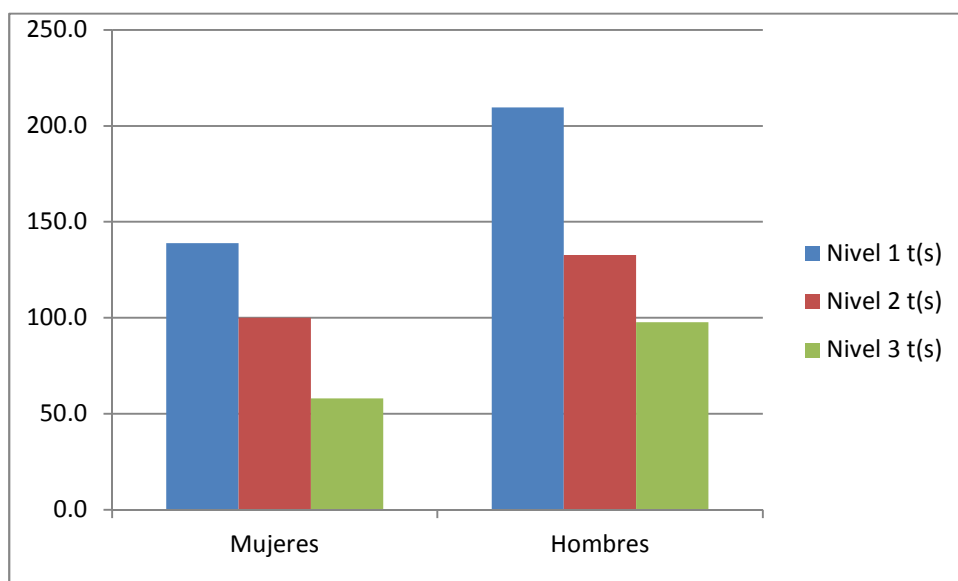
**Figura 37. Tiempo promedio por nivel para estudiantes entre 7 a 9 años**

En la Figura 37 se observa que mientras la edad aumenta, el tiempo de permanencia dentro del nivel también aumenta. Analizando la Figura 37 y 38 se puede observar que el tiempo promedio de permanencia en el nivel y el promedio de palabras acertadas tienen una relación directa entre ellas ya que mientras el jugador permanezca en el juego sin perder, mayor serán las palabras acertadas.



**Figura 38. Promedio de palabras acertadas por nivel para estudiantes entre 7 a 9 años**

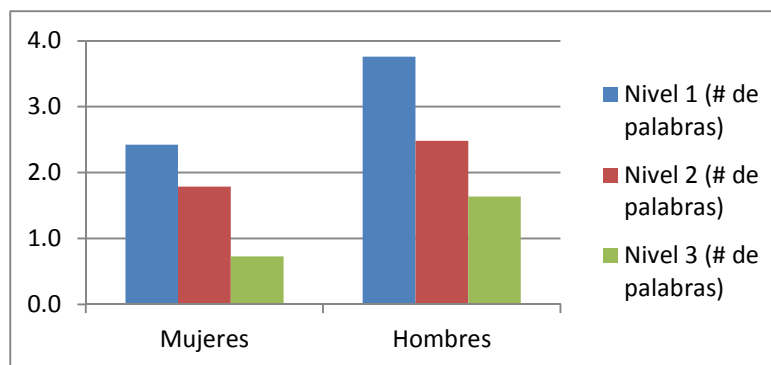
- Tiempo promedio de cada nivel dependiendo el género



**Figura 39. Tiempo promedio de cada nivel dependiendo el género**

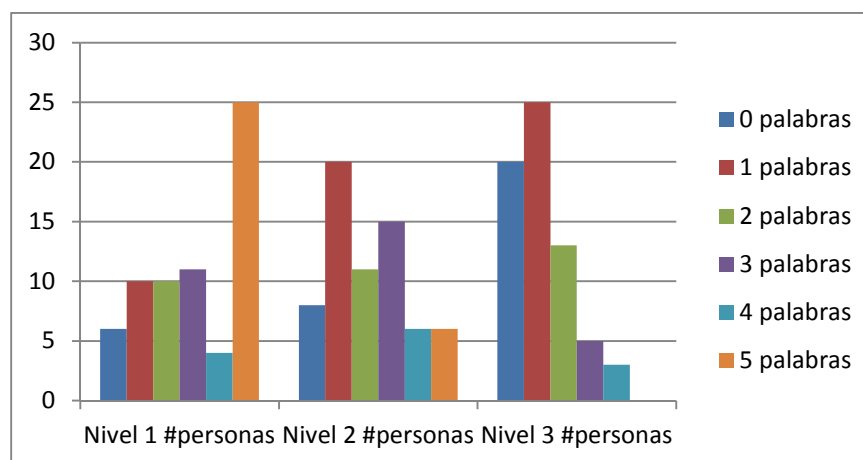
La Figura 39 y 40 muestran el tiempo promedio que jugaron los niños y niñas en cada uno de los niveles y el promedio de palabras acertadas respectivamente. Además muestra que el género masculino tiene un mejor desempeño dentro de los 3 niveles de dificultad.

- Promedio de palabras acertadas dependiendo el género



**Figura 40. Promedio de palabras acertadas en cada nivel dependiendo el género**

- Número de palabras acertadas por nivel



**Figura 41. Cantidad de estudiantes por número de palabras acertadas en cada nivel**

La Figura 41 muestra la cantidad de palabras acertadas que obtuvieron los estudiantes entre los 7 y 9 años de edad dentro de cada nivel de dificultad. Analizando esta figura se puede observar que en el primer nivel de dificultad existe una mayor cantidad de estudiantes que lograron acertar todas las palabras, a diferencia del nivel de mayor dificultad donde se puede apreciar que existe una mayor cantidad de estudiantes con una o cero palabras acertadas.



- Porcentaje de niños por cantidad de aciertos nivel 1

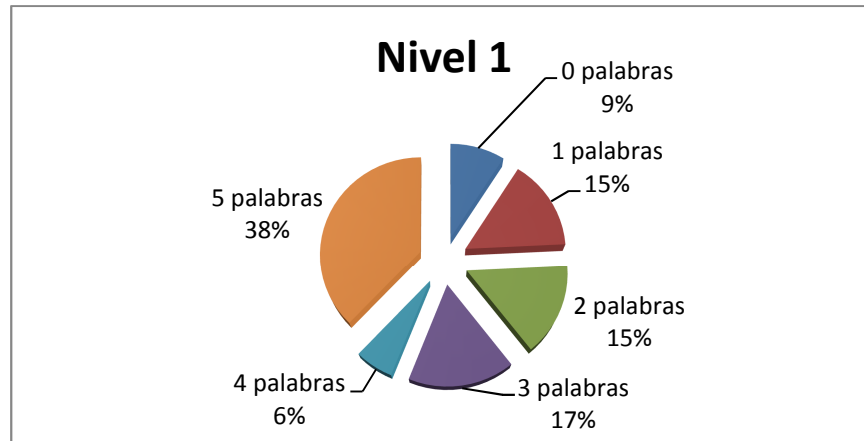


Figura 42. Porcentaje de estudiantes por cantidad de aciertos nivel 1

- Porcentaje de niños por cantidad de aciertos nivel 2

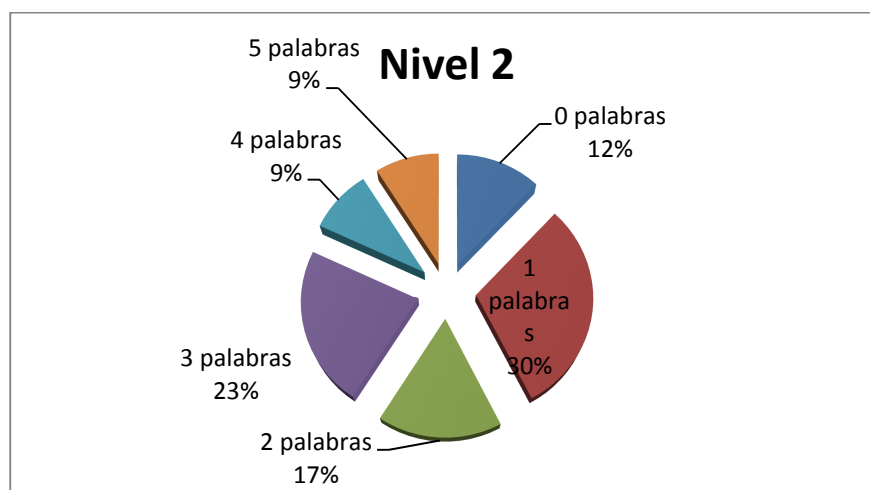
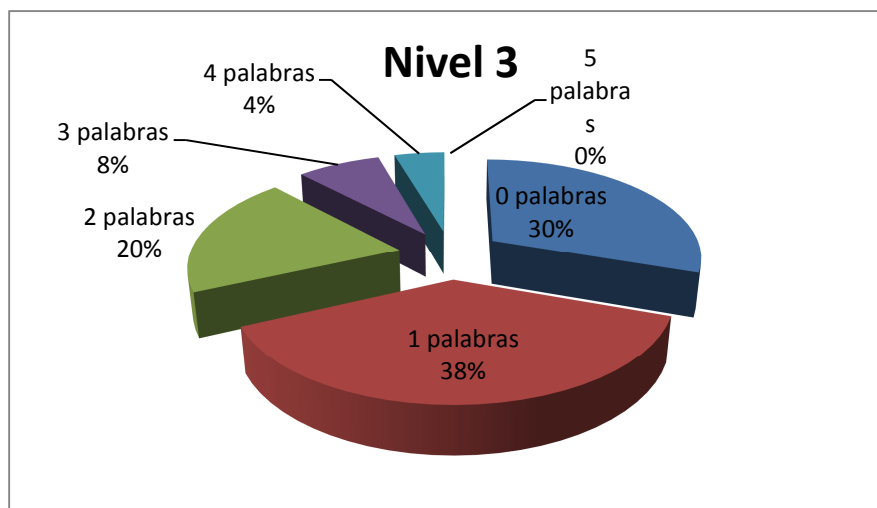


Figura 43. Porcentaje de estudiantes por cantidad de aciertos nivel 2

- Porcentaje de niños por cantidad de aciertos nivel 3



**Figura 44. Porcentaje de estudiantes por cantidad de aciertos nivel 3**

Las Figuras 42, 43 y 44 muestran el porcentaje de estudiantes con sus respectivos aciertos en cada nivel. Analizando estas figuras se puede observar que el número de aciertos disminuye conforme el nivel de dificultad aumenta ya que el primer nivel de dificultad el 38% de estudiantes logró 5 aciertos, en el segundo nivel el 9% de estos lo lograron y en tercer nivel el 0% de estos logró obtener los 5 aciertos. Esto confirma la existencia de un aumento de dificultad del videojuego conforme va avanzando en él.

### **5.1.5. Evaluación pedagógica**

Para la evaluación pedagógica se utilizó la plantilla propuesta por Pere Marqués para la evaluación de videos Educativos. De acuerdo a la siguiente escala:

Tabla 10. Escala utilizada para la valoración de resultados

1	Baja
2	Correcta
3	alta
4	Excelente

Los resultados obtenidos para la validación del videojuego se observa en la Figura 45.

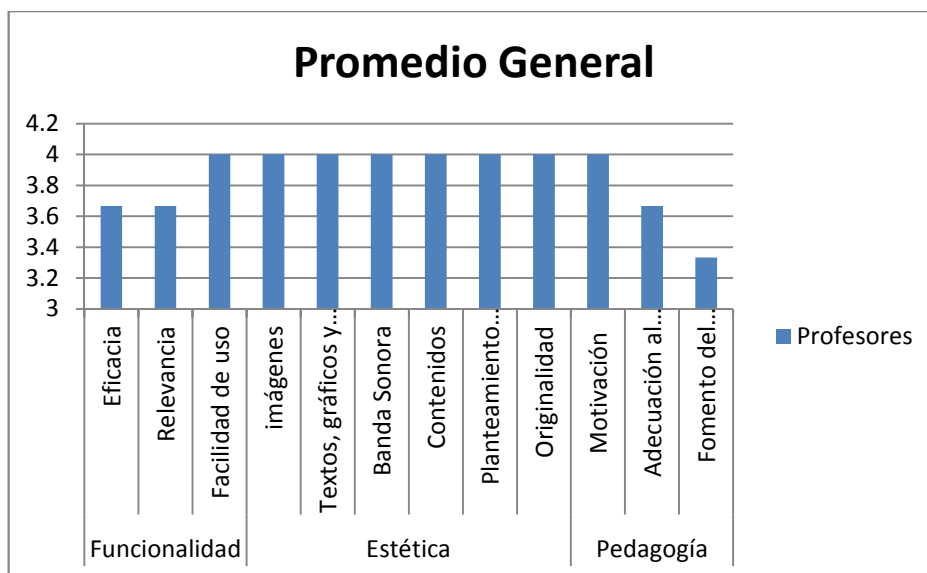


Figura 45. Promedio general de la valoración del videojuego

Como se observa en la Figura 45 la valoración global del videojuego por parte de los profesores se encuentra en la escala 4 (excelente). Las características correspondientes al parámetro estética se encuentran en la escala 4 (excelente), las del parámetro funcionalidad se encuentran en la escala 3 y 4 (alta y excelente), las del parámetro pedagogía se encuentran en la escala 3 y 4 (alta y excelente). El análisis de esta última figura permite validar que el videojuego puede ser utilizado como un recurso educativo.

## 5.2. Pruebas de Rendimiento

Las pruebas de rendimiento se realizaron en una computadora con las siguientes características:

- Marca Asus
- Windows 8.1
- RAM 8 GB
- Procesador Intel core I7
- Disco duro 500gb

Tabla 11. Resultados pruebas de rendimiento 1.

No.	Navegador	Versión	Tiempo de Carga primera vez (s)	Tiempo de Carga Segunda vez (s)
1	Chrome	43.0.2357.132 m	21.1	2.24
2	Firefox	39	35.56	15.77
3	Internet Explorer	11.0.9600.17842	10.84	7.35
4	Safari	6.1.1	12.3	3.4
5	Opera	30.0.1835.125	21.53	2.1

También se realizó las pruebas en otra computadora con las siguientes características:

- Windows 7 escritorio
- Pentium dual core 2.20 GHz ram 4 32 bits
- disco duro 320 GB

Tabla 12. Resultados pruebas de rendimiento 2

No.	Navegador	Versión	Tiempo de Carga primera vez (s)	Tiempo de Carga Segunda vez (s)
1	Chrome	43.0.2357.132 m	23.19	3
2	Firefox	39	26.54	13.21
3	Internet Explorer	11.0.9600.17842	12.41	8.66
4	Safari	6.1.1	12.3	3.4
5	Opera	30.0.1835.125	15.31	3.13

Como se puede observar dentro de la tabla 20 y 21, el videojuego se ejecuta sin problema alguno dentro de cualquier navegador. También se puede observar que los datos en cache son de mucha ayuda para la reproducción del videojuego ya que el tiempo de carga del videojuego por segunda vez es significativamente menor que el primer tiempo de carga.

### 5.3 Etapa Postmortem

Siguiendo los pasos del proceso de desarrollo de videojuegos Huddle, una vez finalizado el videojuego con sus respectivas pruebas y correcciones se obtiene el reporte Postmortem, el cual sirve como retroalimentación para un futuro proyecto.

Tabla 13. Reporte Postmortem

Plantilla del Documento de Producción	
CAMPO	DESCRIPCIÓN
<b>ANTECEDENTES</b>	
Título	Recoge Letras
Estudio/Diseñadores	David Reyes
Género	Educativo
Plataforma	Computador, navegador
Antecedente	NA

CONTINÚA →

---

**MIEMBROS DEL EQUIPO**

David Reyes

**SIGN-OFFS**

David Reyes –Desarrollador

**EFECTO POSITIVO**

**Declaraciones** una vez empezado el proyecto las correcciones continuas ayudaron a corregir y mejorar al videojuego

**Cómo continuar así?** corregir cada avance realizado

**EFECTO NEGATIVO**

**Declaraciones** Se tomó demasiado tiempo en el aprendizaje de la herramienta para desarrollo

**Cómo corregir?** dedicar mayor tiempo al estudio de la nueva herramienta

**CONCLUSIÓN**

Realizando una buena planificación sin acumular trabajo, los avances lograron ser exitosos dentro de un tiempo prudente

---

## **CAPÍTULO 6:**

# **CONCLUSIONES Y RECOMENDACIONES**

### **6.1. Conclusiones**

- El proceso de desarrollo Huddle para el desarrollo de videojuegos está enfocado a proyectos independientes, puede ser adaptado a grupos de trabajos multidisciplinarios ya que éste es un proceso de desarrollo ágil.
- Pese a tener una curva de aprendizaje larga, el motor de videojuegos Turbulenz es una herramienta muy completa que permite la realización de videojuegos complejos para la Web, a su vez posee una plataforma de soporte propia y es de código abierto.
- La etapa de operaciones concretas según Piaget concuerda con los objetivos del videojuego que están enfocados justamente a los niños de entre 7 y 11 años.
- El uso de videojuegos dentro de la educación es útil siempre y cuando el juego sea un complemento al aprendizaje de los alumnos ya que para que un videojuego sustituya a la educación formal, falta muchos avances por realizar.
- Con los datos obtenidos de las pruebas realizadas a niños entre los 7 y 9 años se puede concluir que existe una diferencia de desempeño entre edades y género de los estudiantes, siendo mejor el desempeño de los estudiantes mayores.

## 6.2. Recomendaciones

- Se recomienda a la universidad que dentro de las materias de desarrollo se evalúen herramientas nuevas como Turbulenz entre otros.
- Se recomienda el uso del proceso de desarrollo de videojuegos Huddle, ya que es un proceso muy ágil.
- Se recomienda la utilización del motor de videojuegos Turbulenz ya que este utiliza el estándar html5, lo cual permite que el juego sea ejecutado dentro de cualquier navegador actual.
- Se recomienda el desarrollo y el uso de videojuegos como parte complementaria al desarrollo del aprendizaje de los niños.



## Bibliografía

- Berjon, R., Leithead, T., Doyle Navara, E., O'Connor, E., & Pfeiffer, S. (Diciembre de 2012). *W3C Working Draft*. Recuperado el 30 de Julio de 2014, de W3C: <http://www.w3.org/TR/2012/WD-html51-20121217/Overview.html#contents>
- Eguíluz Pérez, J. (25 de Marzo de 2009). *Introducción a JavaScript*. Recuperado el 31 de Julio de 2014, de librosweb: [http://dis.um.es/~lopezquesada/documentos/IES\\_1314/IAW/curso/UT7/libroswebjavascript/www.librosweb.es/javascript/pdf/introduccion\\_javascript.pdf](http://dis.um.es/~lopezquesada/documentos/IES_1314/IAW/curso/UT7/libroswebjavascript/www.librosweb.es/javascript/pdf/introduccion_javascript.pdf)
- Elliot, E. (2014). *Programming JavaScript Applications*. O'REILLY.
- Fuentes, R. (2012). Jean Piaget, aportes a la educación del desarrollo del juicio moral para el siglo XXI . 55-69.
- Gil Juárez, A., & Vida Mombiela, T. (2007). *Los videojuegos*. Barcelona: Editorial UOC.
- Griffiths, M. (2002). Recuperado el 28 de Mayo de 2014, de Education and health: <http://dh101.humanities.ucla.edu/DH101Fall12Lab4/archive/files/6070f0882cc89baa5a12992aaea56028.pdf>
- Lubbers, P., Albers , B., & Salim, F. (2001). *Pro HTML5 Programming*. apress.
- Lunn, I. (2013). *CSS3 Foundations*. Missouri: treehouse.
- Nagle, D. (2014). *HTML5 Game Engines: App Development and Distribution*. CRC Press.
- Oblinger, D. (2006). *Simulations, games, and learning*. ELI White Paper.
- Piaget, J. (1981). La teoría de Piaget . *Infancia y aprendizaje* , 13-54.
- Rossi, G., Pastor, O., Schwabe, D., & Olsina, L. (2008). *Web Engineering: Modelling and Implementing Web Applications*. London: Springer London.
- Schwabe, D., & Rossi, G. (1998, June). *Developing Hypermedia Applications using OOHDM*.

Squire, K. (2003). Video games in education. *Int. J. Intell. Games & Simulation*, 49-62.

Turbulenz Limited. (15 de Enero de 2015). *Turbulenz 0.28.0 documentation* .  
Obtenido de Welcome to the Turbulenz documentation!:  
<http://docs.turbulenz.com/index.html>

Urrutia, G. A., López, C. E., Martínez, L. F., & Corral, M. A. (2010). Procesos de desarrollo para videojuegos. *CULCyT: Cultura Científica y Tecnológica*, 25-39.

West, M. (2013). *HTML5 Foundations*. treehouse.

Wolf, M. J. (2008). *The Video Game Explosion: A History from PONG to Playstation and Beyond*. Greenwood Press.

**REYES ROSERO DAVID ESTEBAN****DATOS PERSONALES**

<b>APELLIDOS Y NOMBRES</b>	David Esteban Reyes Rosero
<b>DOCUMENTO DE IDENTIDAD</b>	1718098153
<b>FECHA DE NACIMIENTO</b>	31-03-1990
<b>LUGAR DE NACIMIENTO</b>	Quito, Pichincha, Ecuador
<b>ESTADO CIVIL</b>	Soltero
<b>DIRECCIÓN</b>	Sangolqui, Urb. Los Chillos
<b>TELÉFONO CASA</b>	2871003
<b>TELÉFONO MÓVIL</b>	0995543139
<b>E-MAIL PERSONAL</b>	<a href="mailto:david.esteban.reyes@gmail.com">david.esteban.reyes@gmail.com</a>

## FORMACIÓN ACADÉMICA

<b>Universitarios:</b>	Universidad de las Fuerzas Armadas - ESPE
	Departamento de Ciencias de la Computación Ingeniero en Sistemas e Informática (2008-2015)
<b>Estudios Secundarios:</b>	Unidad Educativa “Ángel Polibio Chaves” Bachiller Físico – Matemático (2000-2008)
<b>Estudios Primarios:</b>	Unidad Educativa “Ángel Polibio Chaves”
<b>Idioma Extranjero:</b>	Suficiencia del idioma en la Escuela Politécnica del Ejército
	Dominio del idioma ( Bueno)

## CURSOS Y SEMINARIOS

### UNIVERSIDAD SAN FRANCISCO DE QUITO

CCNA (Cisco Certified Network Associate)

CCNA1 – Aprobado

## DESTREZAS

- Dominio de Sistemas Operativos
  - Windows
  - Linux
  - MacOS
- Lenguajes de programación
  - Java
  - C++
  - JavaScript
  - Html5
  - C#

**EXPERIENCIA LABORAL****MIPRO (Ministerio de Industrias y Productividad)**

Pasantías-Área de Soporte Técnico

Ago. 2012 – Sept. 2012

**JEANVEOIL CIA. LTDA**

Área de Desarrollo

Mayo. 2013 – Enero 2014

**ICS-EC-AJ CIA. LTDA (Integral Consulting Services del Ecuador)**

Analista de Procesos

Septiembre 2015 – Actualmente

**REFERENCIAS**

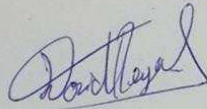
Anita Correa	Gerente Jeanveoil	0991215131
Arión Jaramillo	Gerente General ICS	0999806641

---

**David Esteban ReyesRosero**  
**1718098153**

HOJA DE LEGALIZACIÓN DE FIRMAS

ELABORADO POR



Sr. Reyes Rosero David Esteban

DIRECTOR DE LA CARRERA



Ing. Mauricio Campana

Sangolquí, Octubre 2015