



**UNIVERSIDAD PEDAGOGICA
NACIONAL**

Educadora de educadores

**CASTLE CODE - PROTOTIPO DE VIDEOJUEGO EDUCATIVO COMO MATERIAL
DIDÁCTICO PARA EL APRENDIZAJE DE LA PROGRAMACIÓN**

David Moreno Ortiz

Karla Nicole González Barahona

Universidad Pedagógica Nacional de Colombia

Facultad Ciencia y Tecnología

Departamento de Tecnología

Bogotá D.C.

2021

**CASTLE CODE - PROTOTIPO DE VIDEOJUEGO EDUCATIVO COMO MATERIAL
DIDÁCTICO PARA EL APRENDIZAJE DE LA PROGRAMACIÓN**

Trabajo de Grado para optar al título de
Licenciado en electrónica
Licenciado en Diseño Tecnológico

Autores:

David Moreno Ortiz

Karla Nicole González Barahona

Director:

David Peña Morales

Universidad Pedagógica Nacional de Colombia

Facultad Ciencia y Tecnología

Departamento de Tecnología

Bogotá D.C.

2021

Dedicatoria

Queremos dedicar este trabajo principalmente al Docente David Peña por su colaboración y paciencia, y a todos nuestros compañeros y amigos que aportaron con sus opiniones.

Agradecimiento

Agradecemos primeramente a Dios; a la Universidad Pedagógica Nacional por todo el conocimiento dado para culminar este logro académico; A nuestras familias por su amor incondicional y apoyo desde el inicio, también a quien ha hecho posible que nosotros sigamos aquí, subiendo un escalón más en nuestras vidas profesionales.

Tabla de contenido

Índice de figuras	7
Índice de tablas	10
Resumen	11
Abstract	12
Capítulo 1. Introducción	13
Planteamiento del problema.....	17
• Delimitación del proyecto.....	18
Justificación	20
Objetivos.....	23
• Objetivo general.....	23
• Objetivos Específicos.....	23
Estado del Arte	24
Capítulo 2. Marco Teórico.....	30
• Lógica de la programación.....	30
• Algoritmo.....	30
• Videojuegos.	31
• Videojuegos educativos.....	32
• Unity 3D.....	33
• Visual Studio.....	34

• Illustrator	35
• Windows	36
Capítulo 3. Metodología	38
• Metodología Scrum.....	38
• Análisis Del Equipo.	38
• Definición de Roles.....	39
Etapas de la metodología Scrum para el desarrollo de proyecto	41
Capítulo 4. Descripción del desarrollo del prototipo	42
Fases de desarrollo del prototipo	42
Fase 1. Idea del proyecto y definición de requerimientos.	42
Fase 2. Planificación y Diseño del proyecto.	43
Cronograma.....	43
Diagram.....	44
Retroalimentación del proceso por medio de encuesta.....	45
Proceso de creación – Imágenes vectorizadas Adobe Illustrator	49
Capítulo 5. Conclusiones	63
Anexos	69
• Descripción de Scripts	70
• Game Document Design	72
Parte 1 Concepto Inicial:	75

Identidad del Juego / Mantra:	75
Pilares del Diseño:	75
Resumen de Género/Historia/Mecánicas:.....	75
Características:	75
Interfaz:	76
Estilo de Arte:	77
Música/Sonido:	80
Parte 2 Desarrollo:	81
Diseño personaje/objeto.....	81
Nivel/Escenario/Cuarto.....	81
Flujo de Juego	84
Interfaz	85
Menú principal	85

Índice de figuras

Figura 1 Capturas de pantalla de algunos de los videojuegos que abordan temáticas de programación y codificación, elaboración propia.....	16
Figura 2 Línea del tiempo de algunos documentos relacionados sobre los Serious Games, elaboración propia.....	24
Figura 3 Entorno de producción del videojuego dentro de la interfaz de Unity para juego 2D, elaboración propia.....	34
Figura 4 Ambiente virtual de Visual Studio versión 2019, elaboración propia.....	35
Figura 5 Plataforma del programa Adobe Illustrator 2021, elaboración propia.....	36
Figura 6 Tablero con lluvia de ideas sobre el proyecto, elaboración propia.	42
Figura 7 Interfaz de Unity versión 2019.4.17.f1, elaboración propia.....	44
Figura 8 Diagrama del diseño del Videojuego, elaboración propia.....	44
Figura 9 Encabezado utilizado en la encuesta, elaboración propia	45
Figura 10 Grafica, resultados de la pregunta sobre las ilustraciones generales del videojuego, elaboración propia.....	46
Figura 11 Graficas sobre las preguntas de las ilustraciones del videojuego, elaboración propia.	47
Figura 12 Algunas respuestas de las preguntas abiertas efectuadas, elaboración propia	48
Figura 13 Boceto dentro del programa Illustrator, elaboración propia.....	49
Figura 14 Ambiente del videojuego, dispuesto en la plataforma de Adobe Illustrator, elaboración propia.	50
Figura 15 Disposición del Menú de Inicio, elaboración propia.....	51
Figura 16 Wireframe de la disposición del primer nivel, elaboración propia	51
Figura 17 Movimiento del personaje, elaboración propia.	52

Figura 18 Proceso de creación del personaje dentro del programa Adobe Illustrator. elaboración propia.	52
Figura 19 Escena de videojuego de practica en Unity, elaboración propia.	53
Figura 20 David Peña -Pix A Coding Adventure.	54
Figura 21 Escena del nivel uno del videojuego, elaboración propia.	55
Figura 22 Sprite editor Castle Code, elaboración propia.	55
Figura 23 Castle Code Desarrollo, elaboración propia.....	56
Figura 24 Personaje dentro de la interfaz de Unity, elaboración propia.....	57
Figura 25 Personaje, muestra del movimiento, elaboración propia.....	57
Figura 26 Cámara, herramientas de control, elaboración propia.....	57
Figura 27 Desarrollo del videojuego, elaboración propia.....	58
Figura 28 Desarrollo del panel, elaboración propia.....	59
Figura 29 Código Panel, elaboración propia.....	60
Figura 30 Código Diamante, elaboración propia.....	61
Figura 31 Paneles Nivel 7 muestra de la instrucción a cumplir, guardar una variable, Castle Code, elaboración propia.....	62
Figura 32 Carpetas de Scripts, elaboración propia.....	70
Figura 33 Ambiente Juego Extraído De Internet.....	77
Figura 34 Juego Aladino -Extraído De Internet.....	77
Figura 35 Juego Mario - Extraído De Internet.....	78
Figura 36 Video Juego Plataforma - Extraído De Internet.....	78
Figura 37 Juego Scape Room - Extraído De Internet.....	79
Figura 38 Castle Code - Fuente Propia.....	79

Figura 39 Personaje Alex - Fuente Propia	81
Figura 40 Castle Code - Fuente Propia	83
Figura 41 Castle Code Nivel 1 - Fuente Propia	83
Figura 42 Diagrama Flujo Niveles -Fuente Propia.	84
Figura 43 Diagrama De Flujo- Fuente Propia.....	85
Figura 44 Castle Code Menú -fuente propia.....	85

Índice de tablas

Tabla 1 Roles gestionados por la metodología. 40

Tabla 2 Cronograma de actividades generales..... 43

Resumen

En el mundo de las industrias dedicadas a la creación de juegos evidenció una evolución progresiva, desde juguetes de madera como los que en un momento produjo la empresa LEGO® hasta videojuegos como los desarrollados por la empresa Nintendo®. Ahora bien, al hablar de videojuegos es conveniente mencionar aquellos que se utilizan en sectores como la formación, la publicidad o la educación, denominados juegos serios o también *Serious Games*. Actualmente esta clase de videojuegos es característico por tener principalmente contenidos pedagógicos que fortalece alguna temática disciplinar, como por ejemplo el aprendizaje de la programación. Dado lo anterior el presente trabajo de grado cumplió con el objetivo de diseñar y desarrollar un prototipo de videojuego, didácticamente pensado, que facilite el aprendizaje de algunos conceptos importantes relacionados con la codificación y la programación como lo son las secuencias lógicas simples, los condicionales o los ciclos o bucles, haciendo uso del pensamiento algorítmico.

Abstract

The industry of development and creation of games has evolved progressively, from wooden toys produced by LEGO® to videogames designed by Nintendo®. Furthermore, to speak about video games means talking about digital games used in teaching, advertising, and education. Those games are identified as Serious Games. Nowadays, this kind of video game can be applied in different places focused on pedagogical content, strengthening specific disciplines, e.g., grammar, drawing or programming. The current project proposes designing and developing a Serious Game Prototype called Castle Code, conceived didactically, promoting programming learning, enhancing the player's coding skills, centered on relevant concepts such as logic sequences, conditionals, cycles, or loops. All of this, reaching out to algorithmic and computational thinking.

Capítulo 1. Introducción

Diferentes tipos de actividades lúdicas han acompañado a la humanidad a través del tiempo, juegos tradicionales donde se apuesta por la diversión (la lleva ¹), la estrategia (yermis ²) o el trabajo en equipo (policías y ladrones ³), no obstante esto fue cambiando y prontamente las industrias vieron en los juegos una oportunidad de creación, por ejemplo el caso de la empresa danesa LEGO® la cual inicio como un pequeño taller de carpintería fundado en 1932 por Ole Kirk, el cual después de una gran depresión económica en Europa se vio obligado a reformar su estrategia de trabajo y ello dio comienzo a la producción de juguetes, todo esto durante la segunda guerra mundial LEGO®⁴, continuó creciendo elaborando juguetes de madera y con los años los bloques cambiaron de material migrando al plástico, lo que permitió una flexibilidad de diseño, otorgando mayor libertad de creación, dando paso a un juguete seguro, de gran calidad que ayuda a estimular la imaginación y creatividad de los niños. Rápidamente con todos los avances tecnológicos que presentaba la sociedad, emerge el entretenimiento electrónico y aparecen los videojuegos como una alternativa para el ocio y la distracción.

Se presentan casos de éxito como la empresa japonesa Nintendo®, que a finales del siglo XIX se dedicaba al desarrollo de juegos de cartas, para 1969 lanzó propuestas de cuantioso éxito,

¹ La lleva: El juego consiste en que, dentro de un grupo de personas, “una de estas debe tocar a uno de los integrantes del juego si lo toca este queda, cuando ese jugador toque a otro, ese otro jugador queda y así sucesivamente hasta que un integrante del grupo se salga o no quiera jugar más” (Sanchez, 2019).

² Yermis es un “juego tradicional colombiano jugado normalmente en parques, calles y zonas verdes de las diferentes localidades del país”(Hero, 2021).

³ Policías y ladrones: “consiste en dos equipos, uno son los policías y otro los ladrones, el equipo de los policías trata de atrapar a los miembros del equipo de los ladrones y meterlos en la cárcel, por otro lado, los ladrones pueden ser salvados por sus propios compañeros si son tocados en la cárcel. El juego termina cuando todos los ladrones están en la cárcel” (*Los Policías y Ladrones - Los Juegos Clásicos*, 2019).

⁴ The LEGO Group History - The LEGO Group - About us - LEGO.com ES

entre ellas un brazo mecánico llamado Ultra Hand y la pistola Nintendo Beam Gun, poco a poco fue evolucionando hasta que más tarde cerca de 1977 Nintendo® decide invertir en el desarrollo de consolas, a partir de ahí la empresa produce paulatinamente una amplia gama de lo que hoy se conoce como videojuegos; a la par con Nintendo® muchas otras empresas crecían en la industria de los videojuegos, es oportuno nombrar entre ellas a la fundada en 1972 en por Nolan Bushnell y Ted Dabney bajo el nombre de ATARI®, esta ha sido propiedad de múltiples empresas desde su creación, pionera de los juegos de arcade⁵, consolas de videojuegos y computadoras domésticas.

Los videojuegos siguieron evolucionando y en el año 2000 la industria ya contaba con varios géneros y mecánicas, retando las habilidades de los jugadores, enfrentándolos a cualquier tipo de problemas y en diversas situaciones. De manera natural los investigadores y desarrolladores ven en los videojuegos algo más que solo diversión, encontrando que es posible ofrecer beneficios y resultados provechosos en cualquier ámbito, es ahí cuando empieza a surgir los “*Serious Games*”, o conocidos en español como juegos serios. (López, 2016) “los juegos serios hacen referencia a juegos usados en ámbitos como la formación, la publicidad o la educación” (p, 4).

En la actualidad existen innumerables videojuegos que son desarrollados como apoyo a los procesos educativos de niños, niñas y adolescentes, los cuales se caracterizan principalmente por tener contenido pedagógico que fortalece alguna temática o contenido disciplinar puntual (López, 2016) . El principal objetivo de esta clase de juegos es equilibrar la diversión con las funciones formativas o académicas, algunos casos de éxito de este tipo de juegos con un perfil

⁵ Arcade: “es el término genérico de las máquinas recreativas de videojuegos disponibles en lugares públicos de diversión, centros comerciales, restaurantes, bares, o salones recreativos especializados” (ISSUU, 2019).

educativo son: Minecraft: Education Edition, la versión educativa de Minecraft es popular por su entorno virtual, se destaca por incitar el desarrollo de competencias de aprendizaje en equipo o individualmente⁶.

Civilization es un videojuego de estrategia que consiste en conquistar territorios según la época elegida o Dragón Box, diseñado para estudiar álgebra mediante actividades entretenidas para resolver ecuaciones y aplicar la lógica matemática. En general son muchos los campos de aplicación para los *Serious Games*, dado esto, el ámbito de la enseñanza de la programación utiliza esta clase de recursos digitales y multimedia para facilitar de forma divertida su aprendizaje, como lo es Code Monkey, este juego se basa en ayudar a un mono a encontrar las bananas distribuidas por todo el ambiente o escenario, donde los niños pueden aprender a codificar sin ninguna experiencia previa⁷; Code Combat, está planteado al estilo de un juego de rol, en el que cada usuario maneja un héroe y, superando misiones progresivamente, los estudiantes adoptan el aprendizaje mientras juegan y escriben código desde el principio de su aventura⁸, en la figura 1 se observan las pantallas de los juegos nombrados anteriormente, esto para dar un refuerzo visual a lo descrito.

Dado lo anterior el presente trabajo de grado tiene como objetivo diseñar y desarrollar un prototipo de videojuego, didácticamente pensado, que facilite el aprendizaje de algunos conceptos importantes relacionados con la codificación y la programación como lo son las secuencias lógicas simples, los condicionales o los ciclos o bucles, haciendo uso del pensamiento algorítmico.

⁶ Spanish | Minecraft Education Edition

⁷ <https://www.codemonkey.com/es/>

⁸ <https://codecombat.com/home>



Figura 1 Capturas de pantalla de algunos de los videojuegos que abordan temáticas de programación y codificación, elaboración propia.

Planteamiento del problema

El tiempo que los jóvenes pasan interactuando con dispositivos tecnológicos como computadores, celulares o tabletas digitales se ha intensificado (Alvino, 2021), solo en Colombia se observa que existen más dispositivos como teléfonos inteligentes (60.83 millones) que habitantes (51.07 millones), de igual forma el crecimiento de usuarios en las redes sociales que en la actualidad equivale a 4200 millones de usuarios en todo el mundo lo que representa más del 53% de la población mundial (Murga, 2021), dado lo anterior se concluyó que es posible aprovechar el impacto de la incursión de aparatos tecnológicos y su masificación para utilizarlo como una herramienta que ayude y posibilite a los estudiantes mediar una temática a través de un videojuego educativo, puesto que sus diferentes características hacen que sean capaces de ser adoptados como un instrumento que propenda por la buena circulación de la información con la finalidad de generar un adecuado ambiente de aprendizaje (Bennet, 2013), convirtiéndose, además, en una valiosa estrategia para mejorar la calidad de la educación. En la actualidad se encuentran gran variedad de videojuegos pensados para la enseñanza de lenguajes de programación en donde es posible interactuar con las líneas de código o diagramas de flujo, haciendo énfasis en lenguajes de programación específicos. Sin embargo, se evidencia la necesidad de diseñar un *Serious Game* entretenido y formativo con un enfoque didáctico que motive el aprendizaje de la programación fortaleciendo el uso del pensamiento algorítmico además de mantener una narrativa que permita atraer la atención de los estudiantes más allá de la intencionalidad pedagógica.

- **Delimitación del proyecto**

Para el desarrollo del Prototipo de videojuego educativo, se toman como base tres componentes para tener en cuenta:

Delimitación Geográfica.

El prototipo de videojuego educativo está diseñado para ser implementado en las aulas de informática de las instituciones educativas que cuenten con computadores disponibles, los cuales funcionen con el sistema operativo Windows 7 en adelante.

Delimitación Poblacional.

La población que hará uso del prototipo del videojuego educativo se enfoca en los estudiantes de los grados de sextos a decimo y los profesores del área de tecnología e informática, relacionados con la enseñanza de la programación, esto teniendo en cuenta la clasificación del videojuego según la Asociación de software de entretenimiento (ESRB) la cual corresponde a “Everyone” caracterizada por poseer contenido apto para todas las edades, con mínima violencia y lenguaje moderado, adicionalmente se tiene en cuenta la etapa de desarrollo de los 7 a 11 años de Piaget, en la que indica que en esas edades inicia el pensamiento lógico, algo que requiere primordialmente el jugador del prototipo de videojuego en cuestión, es por ello que se decide orientarlo a una población de 10 años en adelante.

Delimitación de Tiempo

El prototipo de videojuego educativo no tiene un tiempo determinado, puesto que este varía según las temáticas desarrolladas por el profesor, las bases que dispongan los estudiantes sobre el pensamiento algorítmico y su ritmo de aprendizaje, entre otras cosas.

Delimitación Técnica

A continuación, se muestran los ítems mínimos que se deberían tener para ejecutar el videojuego.

- Sistema Operativo: Windows 7 SP1 o superior.
- Procesador de 1.3 GHz o superior.
- Memoria: 1 GB de RAM o superior.
- DirectX: Versión 10 o superior.
- Almacenamiento disponible: 250 MB.

Justificación

Las sociedades constantemente están cambiando y la educación es la base para el desarrollo de conocimientos básicos y de habilidades o competencias que ayuden a las personas a desenvolverse en un mundo globalizado. En el artículo académico de Roncancio et al., (2017) titulado “*El uso de los videojuegos como herramienta didáctica*” describe que en los últimos años los centros educativos se han planteado nuevas estrategias y metodologías de enseñanza, con el objetivo de mejorar el aprendizaje de los estudiantes, para que sean capaces de enfrentar los nuevos retos que exige la sociedad (pp. 2-11).

A la fecha el uso de videojuegos con finalidades educativas es un campo en apogeo y de interés para los académicos e investigadores. Diversos estudios resaltan las bondades en la enseñanza utilizando herramientas creadas como *Serious Games*, no sólo por ser atractivo y motivador para el estudiante, también por la gran variedad de campos y competencias que pueden abarcar (Gallego Durán et al., 2014). Complementando lo anterior Prensky (2001) afirma que los videojuegos incitan la participación, ayudan a cumplir progresivamente pequeños objetivos, dan recompensas y castigos, y la dificultad de cada nivel se puede adaptar según las habilidades, edad y requisitos puntuales de la población a la cual va dirigido (p. 11). Por otro lado, sabemos que el consumo de la tecnología es algo habitual en las nuevas generaciones, la frase de Claudio Peña (2017) del Instituto Tecnológico de Massachusetts “*es casi como si supieran leer, pero no escribir, con las nuevas tecnologías*” describe puntualmente lo que sucede en la mayoría de los casos dentro de la sociedad donde los niños, niñas y jóvenes son usuarios regulares de la tecnología pero desconocen su composición y funcionamiento intrínseco. Estas nuevas generaciones utilizan normalmente videojuegos y son usuarios activos de las redes sociales, como se observa en el último reporte de la situación digital actual en Colombia es

posible corroborar lo nombrado anteriormente, con un 76.4% de los usuarios activos en internet, poseen una o varias cuentas de redes sociales (Alvino, 2021), donde acostumbran realizar actividades que recompensan su perseverancia y dedicación, por lo que esperan el mismo nivel de remuneración en las actividades pedagógicas (Morán, 2007), Acogiendo a lo anterior, enseñar a los niños a habilidades relacionadas con las nuevas tendencias digitales como codificar o programar les proporcionará la posibilidad de manejar y entender la tecnología de forma fluida, de cara a las necesidades que demanda la sociedad, que cada vez requiere mayor mano de obra en temas relacionados con las ciencias computacionales y la informática (Weforum, 2020), adicionalmente la programación ofrece ciertos beneficios en los niños y niñas como el fomentar la creatividad, la resolución de problemas y el pensamiento crítico y reflexivo del uso de la tecnología (Tecnosfera, 2015), por otro lado esta que la lógica de la programación no requiere de ningún conocimiento previo de computadores ni de tecnología en general, así mismo tampoco exige el conocimiento de ningún lenguaje de programación en específico, después de tener establecida bien la lógica de la programación, esta puede facilitar la implementación de un lenguaje propio de programación (Trejos Buriticá, 1999), así mismo esta que el razonamiento lógico permite al estudiante dar sentido a las cosas mediante el análisis y la comprobación de los hechos, recurrir a sus propios conocimientos y modelos internos para hacer y verificar las predicciones y sacar conclusiones (ECOTEC, 2016).

Dado que los videojuegos pueden traer a los ambientes de aprendizaje aspectos claves como la diversión, la motivación y a su vez fortalecer habilidades mentales y motoras entre otras cosas, también se cree que el proceso de aprendizaje se hace más profundo debido al uso del principio denominado *aprender haciendo* (Roncancio et al., 2017), finalmente los videojuegos pueden ser jugados en múltiples dispositivos, teniendo en cuenta que para el año 2021 el número

porcentual de computadores en Colombia equivale a un 76.6% de la población y el tiempo que emplea parte de sus habitantes en los videojuegos, es de una hora al día en promedio (Alvino, 2021).

Lo anterior evidencia la pertinencia de diseñar e implementar un videojuego tipo *Serious Game* como herramienta de apoyo para el aprendizaje que cuente con una interfaz amigable, una narrativa interesante y una mecánica de juego simple y efectiva donde el estudiante experimente y obtenga un acercamiento “diferente” a la programación.

Objetivos

- **Objetivo general**

- Diseñar y programar un videojuego educativo en dos dimensiones empleando conceptos y conocimientos pedagógicos que permitan incentivar el aprendizaje de la codificación y la programación.

- **Objetivos Específicos**

- Diseñar y digitalizar el arte requerido para la creación de ambientes dentro del videojuego.
- Desarrollar la programación necesaria que permitan generar las mecánicas del videojuego por medio de programación en *Visual Studio* y el motor de producción de videojuegos Unity.
- Vincular algunos conceptos de programación, específicamente los temas relacionados con lógica secuencial y líneas de código a las mecánicas del juego.

Estado del Arte

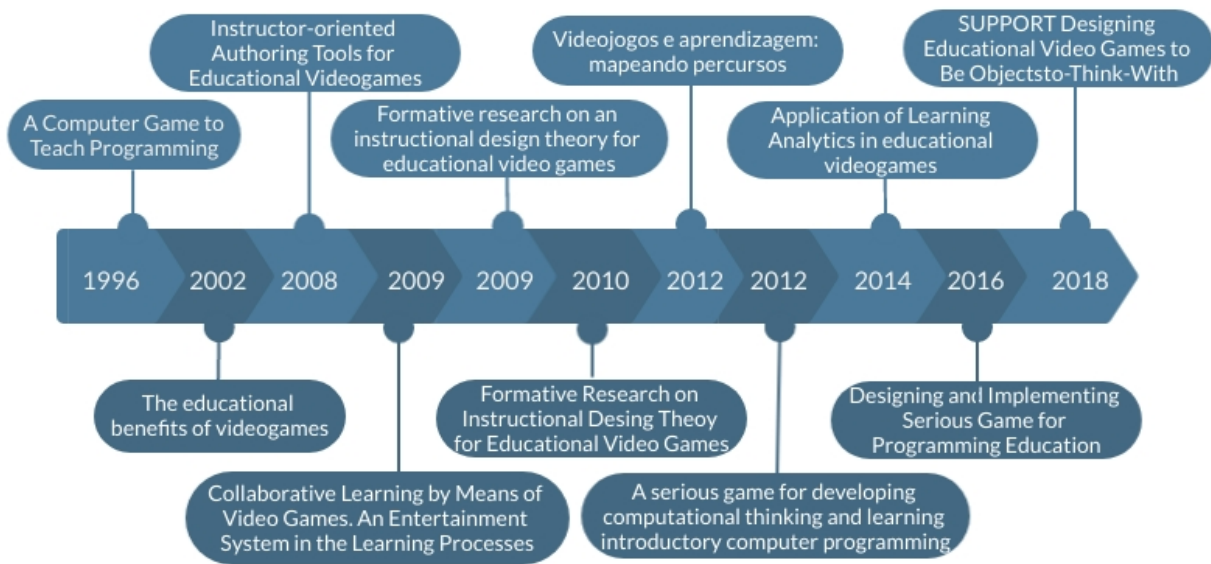


Figura 2 Línea del tiempo de algunos documentos relacionados sobre los Serious Games, elaboración propia.

Los antecedentes que se presentan a continuación fueron ubicados por medio del repositorio de la Universidad Pedagógica Nacional, Google Scholar principalmente en los cuales luego de realizar las búsquedas utilizando palabras clave como videojuegos en la educación, enseñanza de la programación, videojuegos para aprender a programar, salieron ciertos resultados de los cuales después de leer las introducciones se evaluaron cuáles eran provechosos para la investigación y de esta forma finalmente se obtuvo lo siguiente:

A través de la historia los juegos han demostrado tener un papel relevante en la humanidad, estos tuvieron la función de dar diversión tanto a niños como a adultos, en la actualidad los juegos han avanzado progresivamente, explorando nuevas estrategias y campos de acción. Hoy en día es común conocer sobre videojuegos, los cuales poseen un sin fin de géneros y múltiples animaciones. A raíz de estos, surgieron los Serious Games, en cortas palabras pueden

ser videojuegos con un enfoque educativo, a lo largo de la historia se han desarrollado varios videojuegos que se enfocan en diversos temas de aprendizaje, algunos de estos juegos son:

En el artículo académico escrito por Khan Ken en el año 1996 y expuesto en Nacional Educational Computing Conference Proceedings (20^a, Atlantic City, NJ, 22-24 de junio de 1999) habla sobre un juego llamado “The puzzle game of Toon Talk” su narrativa se basa en que una isla se está hundiendo y las personas que están en ella son rescatadas por un marciano llamado Marty que pasa volando, la nave se estrella y Marty resulta herido. En ese momento es cuando el jugador interviene y se ofrece a reparar la nave. Con ayuda de diversos componentes que debe encontrar dentro del ambiente o escena del juego, resolverá cierto número de rompecabezas ordenando de forma lógica tales componentes, la solución a estos rompecabezas implica temas como la medición del tiempo, matemáticas y algunas técnicas de programación.

El juego de rompecabezas de Toon Talk se ha probado con dos grupos de 24 niños de cuarto grado en Menlo Park, California. La metodología que usaron fue observar a cada pareja de niños utilizando Toon Talk durante tres sesiones de 40 minutos, se evidencio que casi todas las parejas consiguieron resolver los primeros 25 rompecabezas sin ayuda.

Concluyen que el hecho de que los niños hayan tenido éxito en resolver los rompecabezas, indica que están aprendiendo programación informática, dado que en un rompecabezas se aplica lo aprendido de anteriores y así sucesivamente a medida que el jugador avanza.

Cabe mencionar también que dentro de las pruebas efectuadas hallaron un niño de 6 años, el cual no consiguió leer los textos explicativos de las pistas, sin embargo, con ayuda de un adulto que le leyera, logro resolver los 25 rompecabezas, a partir de esa anécdota implementaron en el juego la función de conversación de texto de voz (Kahn, 1999).

Torrente et al., (2008) en su trabajo de grado "*Instructor-oriented Authoring Tools for Educational Videogames*" de la Universidad Complutense de Madrid se enfocan en hablar de las herramientas que el docente puede tener para el desarrollo de sus clases, es así como describen que los videojuegos consiguen captar la atención de todo tipo de personas durante horas, ya que son divertidos, motivadores y desafiantes, pero el uso de videojuegos educativos dentro del aula de clases está llena de obstáculos, entre ellos se destacan el costo de desarrollo de los videojuegos comerciales, el cual está sobre los 100.000\$, dado esto se sugiere que los docentes deberían participar de primera mano en el desarrollo de sus propios videojuegos.

En el trabajo de grado de la Universidad de Greenwich (Kazimoglu et al., 2012) titulado "*A serious game for developing computational thinking and learning introductory computer programming*" desarrollan un prototipo de un videojuego llamado "*Program your robot*", este videojuego permite a los estudiantes practicar con programación básica, adquiriendo habilidades como la creación de algoritmos.

Su objetivo es ayudar a un robot a escapar de una serie de plataformas, en cada nivel los jugadores se encuentran con elementos que pueden ser capturados, siendo utilizados para recompensar al jugador.

Dentro de videojuego se destaca la flexibilidad con la que cuentan los niveles, gracias a la distribución aleatoria de los elementos dentro del ambiente, también cuenta con un sistema de puntuación el cual mide la complejidad que tuvo el algoritmo creado por el jugador, generando así una puntuación baja al construir un algoritmo simple. A Pesar de que este *Serious Game* cuenta con grandes semejanzas con otros juegos como, por ejemplo, Ligh-Bot (2008) y Robozzle (2010), defiende su postura argumentando las diferencias que lo identifican. Adicionalmente a lo anterior, realizaron un ejercicio con 25 estudiantes de la Universidad de Greenwich, con el fin de

obtener comentarios positivos y negativos sobre el juego, la mayoría de los estudiantes encontraron el juego adecuado para ayudar al jugador a entender las construcciones básicas de la programación.

En enero del 2014 nace el proyecto “CodeMonkey Coding Adventure”, el primer juego de codificación basado en texto de CodeMonkey, se lanza en febrero de 2014, posteriormente en el 2018. CodeMonkey Jr., la primera aplicación de codificación basada en bloques de CodeMonkey, es un entorno divertido y educativo basado en juegos donde los niños pueden aprender a codificar sin ninguna experiencia. CodeMonkey proporciona elementos educativos para estudiantes que poseen diferentes niveles de experiencia, ellos no solo aprenden conceptos básicos de codificación, sino que también aprenden a codificar utilizando lenguajes de programación reales como C++. En la página oficial de Code Monkey se observan diversos testimonios de docentes que han llegado a usar el videojuego en su versión completa, para aplicarlo al aula de clases, ellos describen que el videojuego les ayudo a explicar lo básico de la programación, a su vez les proporciono una codificación realista sobre el lenguaje a explicar (Monkey, 2020)

En el trabajo de grado sobre *Serious Games* desarrollado en la Universidad Pedagógica Nacional por López (2016) describe, que las vivencias y estudios sobre la incorporación de los videojuegos como herramienta educativa, debe cambiar, dado que la enseñanza de los últimos tiempos se ha convertido en algo complejo, porque los estudiantes han entrado en el auge tecnológico y prefieren estar jugando que estar concentrados en las clases, por lo anterior afirma que los videojuegos son una gran posibilidad para llegar a los estudiantes y encontrar una forma de enseñar programación.

En un artículo académico publicado en la revista “Journal of Korea Game Society” de la Universidad de Changan por Chan-yong Jeong titulado “*Designing and Implementing Serious Game for Programming Education*” describe la creación de un videojuego educativo, para el aprendizaje de la programación. El videojuego trata sobre componer y operar una línea de producción de una fábrica, los empleados se pueden dirigir directamente a las maquinas en operación, para terminar de completar las tareas, cuando una de estas presenta algún error en la ejecución de la línea de producción, muestra llamas y humo lo cual simboliza un accidente. La interacción con los objetos dentro del ambiente del videojuego, se consideran suficientes para comprender e implementar conceptos básicos de programación.

El videojuego funcional fue implementado con 86 personas que no tenían conocimientos previos en programación, las pruebas se realizaron utilizando plantillas comerciales y activos públicos de modelos 3D, obtuvieron que el videojuego puede ser una buena forma de apoyar la introducción a los conceptos de la programación (Jung, 2016).

Lightbot es un videojuego creado por Danny Yaroslavski, lanzado al público en el año 2017, el cual trata sobre la introducción a la programación por medio de rompecabezas, dentro del él se debe guiar al robot para que realice tareas a través de una serie de instrucciones lógicas, superando dificultades. También cuenta con otra versión disponible para estudiantes mayores de nueve años.

Nintendo en el mes de marzo del 2017 lanza al mercado el *Serious Game Human Resource Machine* en el cual se le indican qué procedimientos pueden hacer los trabajadores de pequeñas oficinas para resolver problemas. Conviértete en un empleado modelo, esa es la

motivación del juego para ganarle al enemigo, el cual es una máquina que pretende reemplazar al humano oficinista.

En septiembre del 2019 ingresa al mercado de los videojuegos el *Serious Game Rabbids Coding*, un juego educativo divertido e interesante, que proporciona a las personas herramientas para aprender a codificar. La tarea del jugador es limpiar la nave espacial que ha sido invadida por Rabid. Esto se puede hacer proporcionando instrucciones sencillas o poniendo salchichas a la vista. El juego no requiere ningún conocimiento previo de codificación. Fue creado para permitirle al jugador usar conceptos de codificación sin supervisión constante o guía del maestro. No importa la edad, permite aprender de forma independiente. Su objetivo en cada nivel es proporcionar las instrucciones más simples para completar la tarea, una vez que el jugador haya demostrado sus conocimientos básicos, puede utilizar el entorno de la caja de arena, lo que le permite explorar y seguir las instrucciones para comprender lo que puede hacer (Fandom, 2020).

Capítulo 2. Marco Teórico

Dentro marco teórico se tiene en cuenta conceptos relevantes para la elaboración del prototipo de videojuego, dado esto se empiezan con:

- **Lógica de la programación**

La lógica está presente en gran parte de las asignaturas de razonamiento principalmente en las relacionadas con las matemáticas y la programación, es el fundamento de todos los métodos de representación del conocimiento y del razonamiento, especialmente en sistemas expertos, razonamiento con incertidumbre (encadenamiento de reglas, lógica difusa, etc.), procesado del lenguaje natural, razonamiento espacial y temporal, visión artificial, robótica, lógica epistémica, etc (ECOTEC, 2016).

La Lógica Computacional, sin embargo, no actúa de manera aislada, para la definición de fundamentos de las ciencias computacionales se involucran junto con la Lógica Computacional, disciplinas como Teoría de la Computación y Análisis de Algoritmos, por otra parte, la Lógica Computacional en conjunción con otras disciplinas permite la resolución eficiente de problemas complejos, la lógica de la programación no requiere de ningún conocimiento previo de computadores ni de tecnología en general, así mismo tampoco exige el conocimiento de ningún lenguaje de programación en específico, después de tener establecida bien la lógica de la programación, esta puede facilitar la implementación de un lenguaje propio de programación (Trejos Buriticá, 1999).

- **Algoritmo**

Los algoritmos son una de las características presentes dentro de la lógica de la dentro de la lógica de la programación; Se le conoce como un conjunto de pasos secuenciales y ordenados que permiten lograr un objetivo. “Que sean pasos secuenciales significa que deben ser ejecutados

uno después de otro y que sean pasos ordenados quiere decir que deben llevar un orden quasi-obligatorio” (Trejos Buriticá, 1999). Para el desarrollo de un algoritmo, hay tres partes que se deben considerar:

1. Input (entrada). Información que damos al algoritmo con la que va a trabajar para ofrecer la solución esperada.
2. Proceso. Conjunto de pasos para que, a partir de los datos de entrada, llegue a la solución de la situación.
3. Output (salida). Resultados, a partir de la transformación de los valores de entrada durante el proceso (Csizmadia et al., 2015).

Estas se tienen en cuenta a la hora de crear una secuencia lógica algorítmica dentro del videojuego, como lo es la secuencia para abrir un candado.

- **Videojuegos.**

La definición de videojuego es importante tenerla presente y aún más saber los requisitos de que hace a un producto un videojuego, esto con el fin de catalogar los requerimientos necesarios para la creación del prototipo, teniendo en cuenta esto; Un Videojuego es un tipo de software que se utiliza generalmente para el entretenimiento, se basa en la interacción entre una o más personas y los dispositivos electrónicos que lo ejecutan, estos pueden ser computadoras, máquinas recreativas, consolas de juegos, teléfonos móviles (Baer et al., 1972), etc.

La palabra "video" en sí misma generalmente se refiere a la visualización de gráficos a través de un dispositivo, por lo tanto, la palabra “videojuego” hace referencia a la interacción con los gráficos digitales. La relación del jugador con la interfaz en un videojuego se hace por medio de dispositivos de salida de video (como pantallas, televisores, monitores o proyectores), donde

los programas se graban en casetes, CD, discos, tarjetas de memoria dedicadas para videojuegos o se juegan en línea.

La enorme popularidad de los videojuegos a fines de la década de 1970 generó industrias importantes, los videojuegos pueden ser muy diferentes entre sí, en términos de complejidad, calidad de gráficos y temas, al igual que las películas y la música; los géneros y subgéneros son largos y complejos, y la clasificación de un mismo título puede ser diferente, según quién lo analice (Belli & López, 2008). La diferencia entre los videojuegos y otras formas de pasatiempo es que deben ser interactivos, en otras palabras, los usuarios deben participar activamente del contenido, para ello, se debe utilizar un controlador (también llamado tablero de juego o joystick), a través del cual se envían las órdenes al dispositivo principal (computadora o consola dedicada), y estas órdenes se reflejan en la pantalla junto con las acciones del personaje guiado por el jugador (Ruiz, Josu Ahedo, 2014).

Con todo lo anterior se visualiza a gran escala lo que es un videojuego, sin embargo, dado que lo que se quiere lograr es un videojuego enfocado a la educación, es prudente tener en cuenta lo que define a un videojuego educativo.

- **Videojuegos educativos.**

Los *Serious Games* son los videojuegos educativos usados en sectores educativo, científico, en la atención médica, planificación urbana, ingeniería y política, principalmente; se desarrollan para que los jugadores adquieran y fortalezcan habilidades y conocimientos.

Dentro de la creación de los Serious Game se tiene en cuenta la mediación que se utiliza para llegar al fin educativo, esto sin olvidar el factor de diversión y motivación que el videojuego debe tener hacia el jugador.

Este tipo de videojuegos va dirigido a todo tipo de personas, usados en diferentes tipos de dispositivos o plataformas, en el cual el computador prevalece como principal fuente de uso, el videojuego como herramienta educativa no es tan simple, ya que no se debe usar como un medio para reproducir información lineal ni transmitir conocimientos de la forma tradicional, cuando se refiere a ellos, hay que pensar y entender las particularidades que poseen, sus maneras de interacción y comunicación de la información (López, 2016).

Para la creación de cualquier videojuego, se requiere un motor en el cual elaborarlo, en este caso para la elaboración del prototipo se utiliza el motor de creación de videojuegos Unity.

- **Unity 3D**

Resulta primordial saber que motor de creación de videojuegos es ideal para idear el prototipo, ante esto Unity3D (Unity) propiedad de Unity Technologies, fundada en 1988 por David Helgason, Nicholas Francis, y Joachim Ante en Copenhague, Dinamarca, en la actualidad es uno de los motores de juego más usados, permite utilizar Javascript3, Boo y C#, y dispone de un editor que agiliza el desarrollo del videojuego, en él se pueden realizar videojuegos 3D y 2D, permite exportar los proyectos generados para múltiples plataformas como Windows, macOS, Linux, Consolas, Android lo que lo hace una herramienta flexible, además se reconoce la gran comunidad que posee, lo cual facilita en gran medida la fluidez de información para la producción de algún proyecto (Fuentes, 2014).

Su interfaz cuenta con paneles de funciones movibles para así generar una estructura de las herramientas que más se utiliza el desarrollador, dependiendo del proyecto, consiguiendo facilitar la creación de un proyecto, si desea realizar un seguimiento del comportamiento del objeto en sí, puede hacerlo desde el cuadro que muestra la función. Además, puede probar el

videojuego en construcción antes de generar el ejecutable, lo anterior permite asegurarse de que la aplicación funcione correctamente.

La figura (3) muestra la interfaz de Unity, con la imagen en PNG seleccionada, para la cual en la columna derecha se encuentra el inspector que muestra las características del objeto seleccionado, en la parte inferior está el Game donde se visualiza el videojuego y finalmente las carpetas de los archivos que contienen las imágenes, códigos, sonidos, etc....



Figura 3 Entorno de producción del videojuego dentro de la interfaz de Unity para juego 2D, elaboración propia

- **Visual Studio**

Este software fue desarrollado por Clan Cooper para Microsoft, fue creado para brindar a los usuarios facilidad de uso y conveniencia dentro de la interfaz del motor Unity3D, maneja una codificación conveniente para usuarios con conocimientos básicos en programación, es compatible con muchos de los lenguajes de programación actuales, lo que permite combinar varios programas de computadora en uno. A continuación, en la figura 4 se muestra la interfaz de Visual Studio en uso, la

cual posee unas opciones internas para cambiar su color, la cual ayuda al descanso visual y en su parte derecha se encuentran las librerías que usa Unity para la lectura del código.

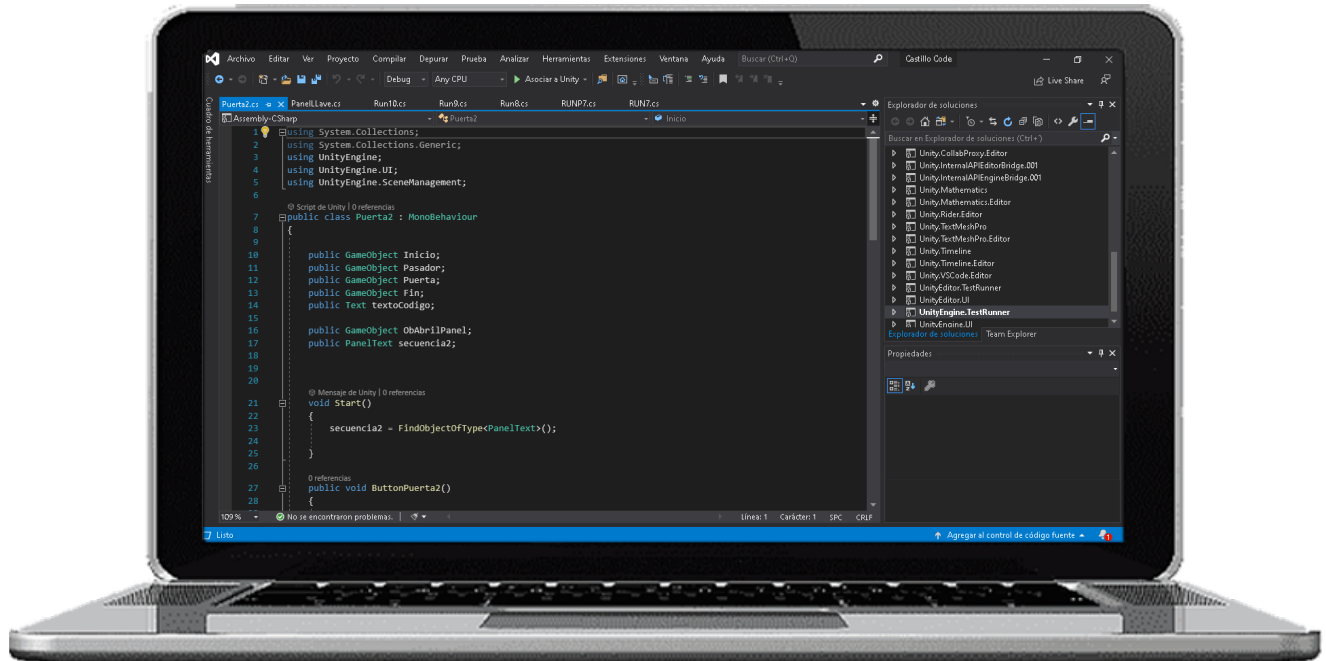


Figura 4 Ambiente virtual de Visual Studio versión 2019, elaboración propia

- **Illustrator**

Para la creación de las gráficas o ilustraciones requeridas dentro del videojuego se hace uso de Adobe Illustrator el cual es un programa desarrollado por la empresa *Adobe Systems* en 1987, el cual se centra en generar gráficos vectorizados, lo que permite escalar la imagen producida a cualquier tamaño, esto le otorga al usuario flexibilidad a la hora de producir gráficos con buena calidad, con el pasar del tiempo el programa ha adquirido nuevas herramientas, las cuales facilitan el trabajo a la hora de diseñar, herramientas como el cuanta gotas, que permite extraer la referencia de un color y aplicarla en el grafico generado, también cabe mencionar que la sección de paleta de colores tiene acceso al código de color, lo cual facilita aplicar la referencia del mismo por ejemplo a un software de programación.

Los gráficos generados en este programa se pueden exportar en diferentes formatos como PNG, JPEG, GIF, PDF entre otros.

a figura 5 muestra un pantallazo de la plataforma del programa Adobe Illustrator, junto con una muestra de color en formato sexagesimal del código RGB.

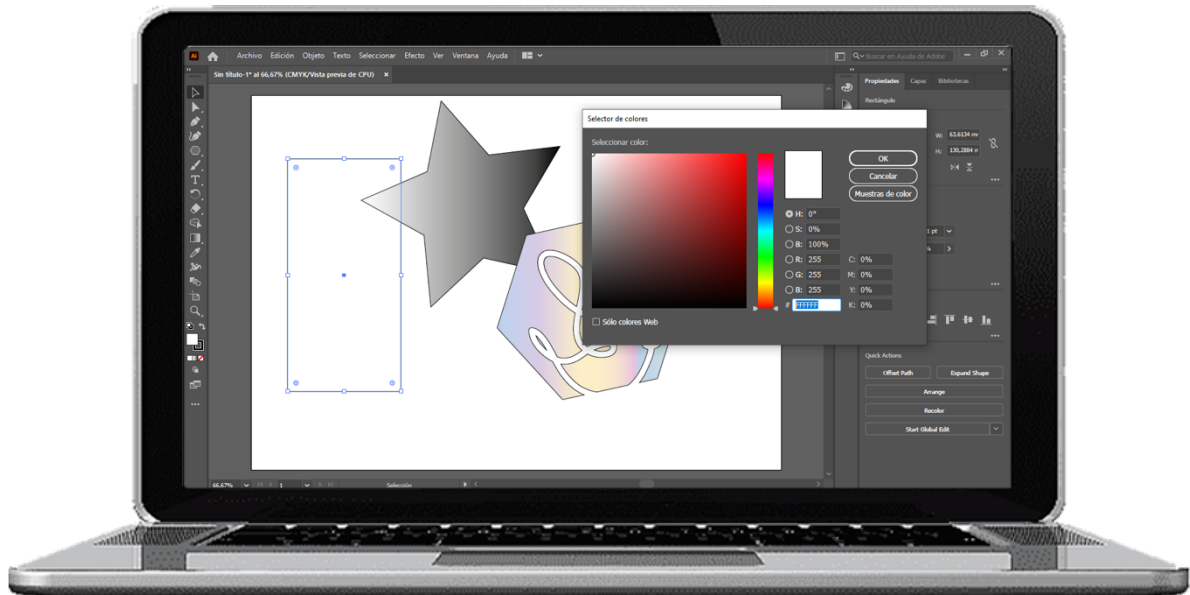


Figura 5 Plataforma del programa Adobe Illustrator 2021, elaboración propia

- **Windows**

La compañía Microsoft fundada en 1970 por Paull Allen y Bill Gates, en noviembre de 1985 lanza Microsoft Windows, una extensión del sistema operativo MS-DOS que proporciona un entorno operativo gráfico. Windows con el pasar del tiempo se genero varias versiones, contando en la actualidad con más de 10 (Microsoft Prensa, 2015).

Microsoft Windows dominó el mercado mundial de computadoras personales, con más del 90% del mercado superando al Mac OS introducido en 1984, las últimas versiones son: Windows 10 para computadoras de escritorio, Windows Server 2019 para servidores y Windows 10 Mobile para dispositivos móviles. El primer idioma español fue Windows 2.1. Como sistema

operativo, Microsoft Windows comienza a ejecutarse desde el momento en que enciende su computadora para administrar todos los componentes de hardware, simplificando la administración de los recursos de la computadora al nivel más bajo.

Proporciona una interfaz gráfica que hace que la interacción del usuario sea simple e intuitiva, y tiene todas las herramientas que los usuarios necesitan para controlar y usar sus computadoras de la manera más conveniente y fácil.

Capítulo 3. Metodología.

- **Metodología Scrum.**

La metodología Scrum se puede aplicar a cualquier tipo de trabajo en equipo para ayudar a administrar y estructurar proyectos de manera efectiva, cabe mencionar que esta metodología está enfocada en equipos de trabajo de gran tamaño, sin embargo su flexibilidad y adaptabilidad en el transcurso del proceso lo hacen una buena metodología para el desarrollo de este proyecto, es simple, una vez formado el equipo, se reparten las tareas de alta prioridad, los plazos de entrega se dividen en ciclos denominados *Sprints*, para los cuales en este documento hacen referencia a cada uno de los momentos implementados, los momentos tuvieron una retribución semanal.

Todos los miembros colaboran con sus propios conocimientos y el trabajo se optimiza gracias a las distintas reuniones que se realizan al final de cada sprint. El objetivo principal del enfoque Scrum es comunicarse con los usuarios, a través de pequeños pasos que mejoran la productividad y la calidad del producto y, lo más importante, mejoran la eficiencia del desarrollo del proyecto.

- **Análisis Del Equipo.**

El primer paso para el desarrollo del proyecto es responder al análisis del equipo y establecer una visión general de la comprensión de la metodología Scrum, lo que les permite ver un registro cualitativo de cómo se está realizando el trabajo internamente.

El objetivo del proyecto se planteó y aplicó a un grupo de 3 personas las cuales la metodología Scrum les permitió llevar un proceso adaptable en el desarrollo del proyecto, dentro de este grupo se hace referencia al asesor de trabajo de grado y los dos docentes en formación, dentro del desarrollo de la metodología se tiene en cuenta al asesor, porque cumple un papel

importante de retribución de comentarios semanales y análisis del proceso, aparte de delimitar los aspectos esenciales dentro del prototipo de videojuego. implementando un seguimiento controlado de tareas, pero a su vez con la flexibilidad de adaptarlas a los cambios que se pueden llegar a generar.

- **Definición de Roles.**

Después del análisis anteriormente nombrado el equipo de trabajo se capacita para integrar conocimientos y proporcionar el material base para entrar en el inicio de la ejecución del proyecto. Como paso a seguir se estableció la asignación de roles y se elaboró el plan de implementación para seguir la metodología scrum.

El equipo está formado por 3 miembros los cuales son el Docente Asesor y los dos Docentes en formación que presentan el proyecto Prototipo de un videojuego Educativo como opción de grado. Las sesiones para retroalimentar ideas y mostrar avances del proyecto se programan una vez a la semana, empleando la plataforma TEAMS como mediación digital para llevar a cabo la sesión, los horarios propuestos para la elaboración del proyecto se dispusieron de la siguiente forma:

Entre semana de lunes a jueves se apartaba un espacio para desarrollar la parte tanto grafica como de programación y finalmente los viernes se realizaba un encuentro con el asesor del trabajo de grado, el cual dicho encuentro según la metodología scrum posee el nombre de *Sprints*, para evaluar los avances y retrospectivas del proyecto, para llevar un control de los compromisos y avances del proyecto se realizaron 15 actas, las cuales se almacenaron en el equipo de trabajo dentro de la plataforma TEAMS.

Los tres roles gestionados por esta metodología se establecen de la siguiente manera:

Rol	Función	Encargado
Scrum Máster / Docente asesor	Es su responsabilidad asegurarse de que la metodología y cada actividad establecida se desarrollen de la forma correcta y que el producto esté correctamente configurado e implementado.	Lic. David Peña Morales
DEVELOPMENT TEAM (DT)	Equipo de desarrollo, es un grupo de personas que trabajan juntas para crear el producto. Son programadores o desarrolladores que codifican los requisitos del sistema.	David Moreno Ortiz Karla Nicole González Barahona
Desarrollo De Gráficos	Equipo encargado de combinar o crear medios artísticos y técnicos para transmitir ideas utilizando imágenes digitales. Usa una variedad de elementos para crear efectos llamativos en una variedad de medios, incluidos medios digitales.	Karla Nicole González Barahona
Programación o Codificación	Navegar por los distintos entornos de desarrollo, utilizando diferentes lenguajes de programación disponibles en su entorno, herramientas de desarrollo (compilador, base de datos, Los datos, el depurador, etc.) que se encuentran disponibles en el idioma que elija, generando código y depuración.	David Moreno Ortiz

Tabla 1 Roles gestionados por la metodología.

Etapas de la metodología Scrum para el desarrollo de proyecto.

La metodología Scrum para el desarrollo de proyectos se divide en 5 SPRINT detallados que se describen a continuación.

SPRINT1-Modulo Requisitos: Esta etapa de desarrollo describe cada requisito mínimo y específico requerido para desarrollar el producto final, como las preguntas que surgen al planificar cada paso, posteriormente se resuelven.

SPRINT2-Modulo Diseño: Esta etapa de desarrollo se divide en dos sub-etapas:

Diseño lógico y diseño físico. El primer paso intermedio se denomina diseño lógico en él se intercambian todas las ideas y posibles soluciones, el segundo paso, denominado diseño físico, donde se transforman las ideas teóricas y los esquemas expuestos en el paso de diseño lógico en especificaciones concisas para el desarrollo del producto final.

SPRINT3-Modulo Implementación: En esta tercera fase se analiza cada requisito y especificación de la fase de pre-desarrollo, se utiliza para generar el código básico para el desenvolvimiento del producto final.

SPRINT4-Modulo Verificación: En este punto, es necesario cumplir completamente con los requisitos y especificaciones identificados en el paso anterior para determinar si el software es estructural y cualitativamente adecuado.

SPRINT5- Modulo Mantenimiento: En este punto, los usuarios pueden utilizar la aplicación de forma regular para detectar errores de software y anomalías en la aplicación. En esta sección, los desarrolladores de software realizan los ajustes necesarios para corregir los errores.

Capítulo 4. Descripción del desarrollo del prototipo

A grandes rasgos los programas principalmente usados para la elaboración del proyecto “Prototipo de un videojuego para la Enseñanza de la Programación” fueron: En el aspecto gráfico Adobe Illustrator 2021 y para el desarrollo de la programación se usó Unity3D versión 2019.4.15f1 de 64 bits la cual es la última versión estable disponible al público.

Fases de desarrollo del prototipo.

Fase 1. Idea del proyecto y definición de requerimientos.

En la primera fase para el desarrollo del proyecto de grado, se plasmaron muchas ideas, a su vez se definieron requisitos necesarios para iniciar a ejecutar el proyecto, dado esto se hizo uso de un tablero para realizar una lluvia de ideas, la figura seis (6) presentada a continuación muestra algunos de los requerimientos que se tuvieron en cuenta y posibles personajes.



Figura 6 Tablero con lluvia de ideas sobre el proyecto, elaboración propia.

. Ante lo anterior se determinó que Unity sería el programa para utilizar en el desarrollo de la programación y Adobe Illustrator para el desarrollo gráfico, estas decisiones se tomaron con base en los conocimientos previos que poseen los integrantes del equipo.

Para adquirir información valiosa se revisaron diversos documentos de grado que trataran sobre el tema de videojuegos en la educación, también se tomó como referencia videos en los que se implementaron videojuegos educativos en diversos campos de la educación.

Finalmente, las ideas y requerimientos fueron aprobados por el Asesor de Trabajo de Grado, para la realización del proyecto Prototipo de un Videojuego educativo para la enseñanza de la programación.

Fase 2. Planificación y Diseño del proyecto.

La planificación del proyecto fue proyectada por el equipo de trabajo de forma general, teniendo en cuenta los roles que se designaron con base en los conocimientos previos.

Cronograma

La tabla dos (2) muestra el cronograma de las fases planteado desde un principio del proyecto, dando cumplimiento según tiempos.

DESARROLLO DEL PLAN			SEMANAS																							
DURACIÓN DEL PLAN	ACTIVIDAD	PORCENTAJE COMPLETADO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
MES 1	Fase 1 /Fase 2	0%																								
	Consulta																									
	Buscar antecedentes																									
MES 2	Fase 3	0%																								
	Diseñar Escenarios																									
	Diseñar acertijos																									
MES 3	Fase 4	0%																								
	Corrección de alg.																									
MES 4	X																									
MES 5	X																									
MES 6	X																									

Tabla 2 Cronograma de actividades generales

Se decide utilizar la plataforma de desarrollo Unity para operar y desarrollar el prototipo del videojuego, para ello se instala la versión 2019.4.17.f1 la cual se puede apreciar en la figura

Fase 3. Implementación, verificación y mantenimiento del proyecto.



Figura 9 Encabezado utilizado en la encuesta, elaboración propia

Retroalimentación del proceso por medio de encuesta

La encuesta tenía como fin, retroalimentar el proceso del funcionamiento propio del videojuego, como la orientación de las ilustraciones, errores de desarrollo, y posibles mejoras.

Para realizar la prueba previa se utilizó una beta del videojuego el cual contaba con cinco niveles disponibles, la prueba se realizó con 23 personas de diferentes edades, estas por medio del correo electrónico tuvieron acceso a la versión beta del videojuego el cual ejecutaron en un computador ya sea de escritorio o portátil, interactuaron con él y finalmente respondieron una encuesta de Google formulario, esta contenía preguntas tanto abiertas como de escala lineal de uno (1) a cinco (5), siendo uno (1) lo más bajo y cinco (5) lo más alto en relación a la apreciación personal de la pregunta correspondiente.

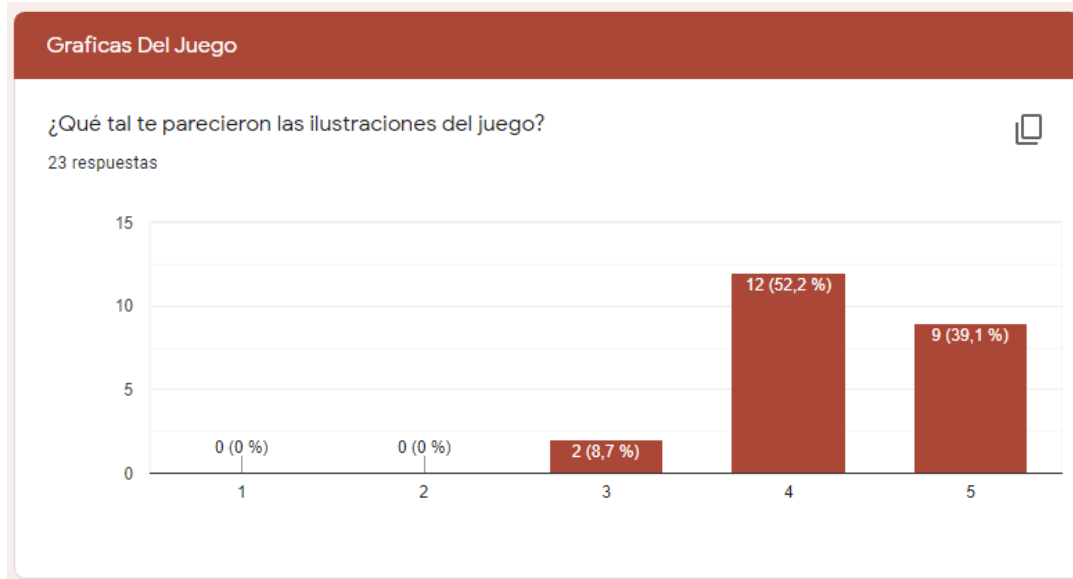


Figura 10 Grafica, resultados de la pregunta sobre las ilustraciones generales del videojuego, elaboración propia.

En la figura diez (10) se observan los resultados de la percepción general que poseen los encuestados sobre las gráficas del videojuego, dos (2) personas lo estimaron con una apreciación de tres (3), doce (12) personas con cuatro un puntaje de (4) y finalmente nueve (9) personas con una calificación de cinco (5). Se realizaron otras preguntas puntuales relacionadas con los gráficos como:

¿Qué tal te parecieron las ilustraciones del panel del código?, ¿Qué tal te parecieron las ilustraciones del Personaje?, ¿Qué tal te parecieron las animaciones del personaje?, en el formato de escala lineal, las cuales arrojaron de igual forma buenos resultados, a continuación, en la figura

once (11) se aprecian las gráficas conjuntas de los descrito anteriormente.

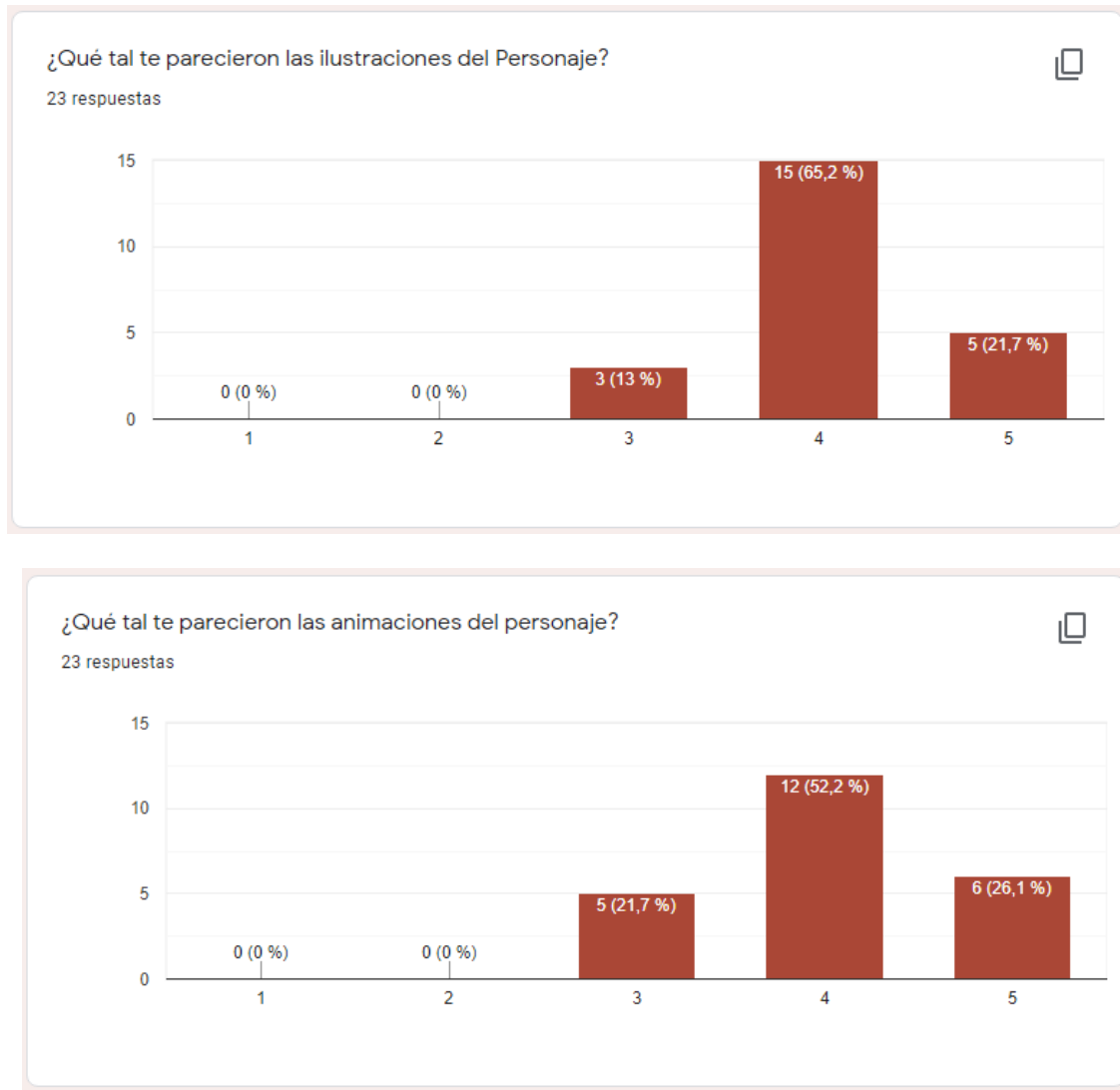


Figura 11 Gráficas sobre las preguntas de las ilustraciones del videojuego, elaboración propia

A su vez se efectuaron preguntas de carácter abierto, las cuales arrojaron los siguientes resultados: Sobre los gráficos los encuestados describieron su aceptación y gusto de estos, también hicieron comentarios acerca de las mejoras que ellos realizarían.

¿Qué opinión tienes sobre las ilustraciones ?

23 respuestas

que tiene buena calidad

Todas muy buenas y la fluidez excelente

que esta muy bien ilustrado además se hace entender arto.

son buenas y acordes

Son buenas y entretienen de que se puede interactuar

es un estilo cartoon bastante bonito, yo diria que si se retocan un poquito mas quedaria muy bonito

Un interesante fondo, perfectamente elaborado teniendo en cuenta el ambiente de castillo antiguo

Son buenas, sin ser excelentes

bastante buenas las ilustraciones, pienso que podrían mejorar, pero están excelentes

Se ven muy bien pero falta un poco de movimiento en los escenarios para que se vea un poco más vivo el

¿Qué consideras MALO en el Nivel 1? Describe.

22 respuestas

Es bueno y es una actividad que entretiene

el personaje es muy lento

Los marcadores de Diamante no empiezan en ceros (el rojo específicamente)

Se me travo

casi no hay nada malo en el nivel 1, pero creo que la jugabilidad debería ser mejor, más opciones o cosas para interactuar

Poco intuitivo, no se sabe que se tiene que hacer. Si no tomas ninguno de los diamantes y vas a la puerta el juego se crashea y deja de funcionar

no se pueden usar las teclas de direccionamiento y se bloqueo el juego

no le veo nada malo al nivel

Figura 12 Algunas respuestas de las preguntas abiertas efectuadas, elaboración propia

A partir de la prueba realizada se logró reafirmar que los gráficos fueron aceptados por el grupo de encuestados, revelando que la orientación grafica debería seguir la misma línea o patrón base, también fue de gran ayuda para identificar errores de sintaxis, movimientos límites del personaje e interacción del videojuego con el usuario.

Proceso de creación – Imágenes vectorizadas Adobe Illustrator

La implementación del desarrollo del proyecto inicio con el trabajo de generar bocetos de las ideas socializadas, en la Figura 13 se observa el boceto generado en el programa Adobe Illustrator, haciendo uso de una tableta gráfica, se logró exponer las ideas, muchas de estas se ilustraron para decidir la viabilidad de esta, teniendo en cuenta tanto el trabajo grafico de animación, así como el de programación, esto haciendo referencia al personaje y el ambiente propio del videojuego.

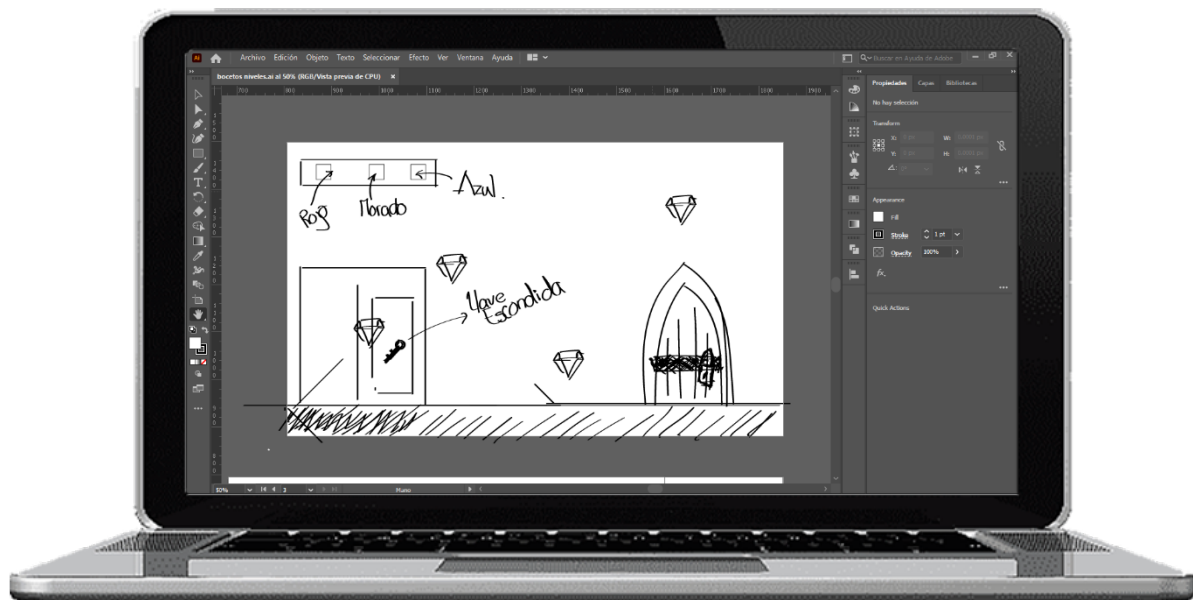


Figura 13 Boceto dentro del programa Illustrator, elaboración propia

Se comenzó por evaluar las ideas empezando por la narrativa, para la cual la idea seleccionada previamente fue la de un castillo, dado esto se realizaron búsquedas de referencias de ambiente de los castillos, para así poder iniciar con la creación del ambiente; el diseño del

personaje se derivó de esa idea inicial, al igual que para el diseño del ambiente, se inició con bocetos de la idea del personaje y finalmente con trabajo y retroalimentación de ideas se logró detallar una identidad del mismo.

En la Figura 14 se evidencia una escena terminada en el programa Adobe Illustrator, cabe mencionar que los primeros diseños del ambiente y del personaje ayudaron a determinar una la ubicación espacial de la estructura dentro del Unity, para apoyar este proceso también se hizo uso de un *Wireframe* realizado en el programa Balsamiq Mockup como se puede ver en la Figura 15 y 16, la disposición del menú de inicio y el primer nivel respectivamente, el programa Balsamiq ayudo también a recrear la interactividad que tiene el jugador con la interfaz del videojuego.

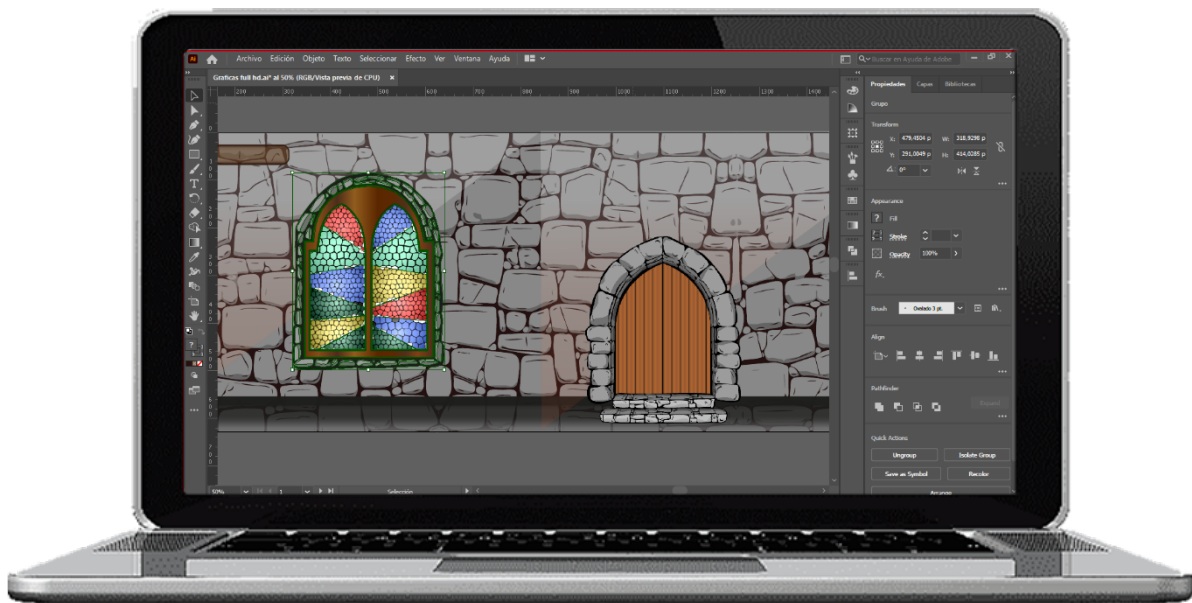


Figura 14 Ambiente del videojuego, dispuesto en la plataforma de Adobe Illustrator, elaboración propia.

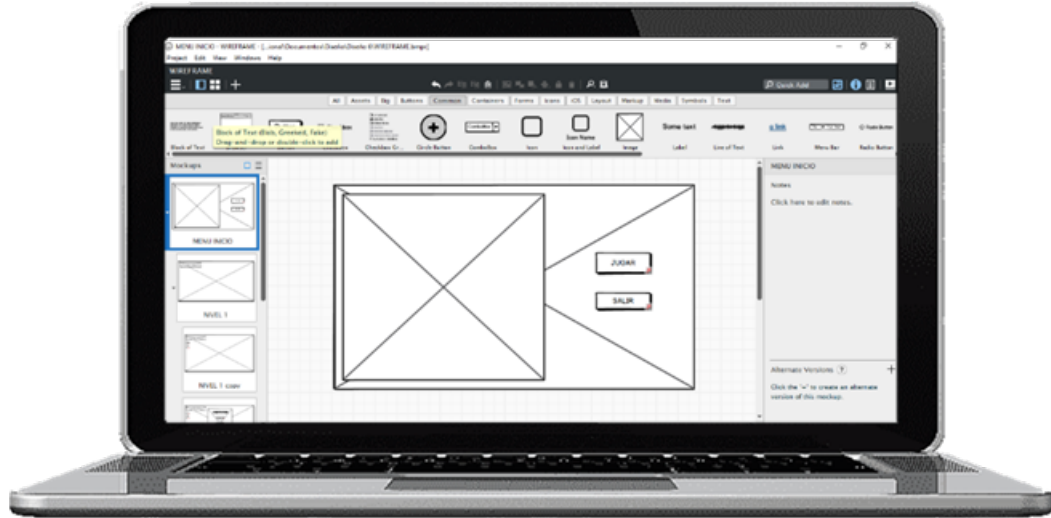


Figura 15 Disposición del Menú de Inicio, elaboración propia

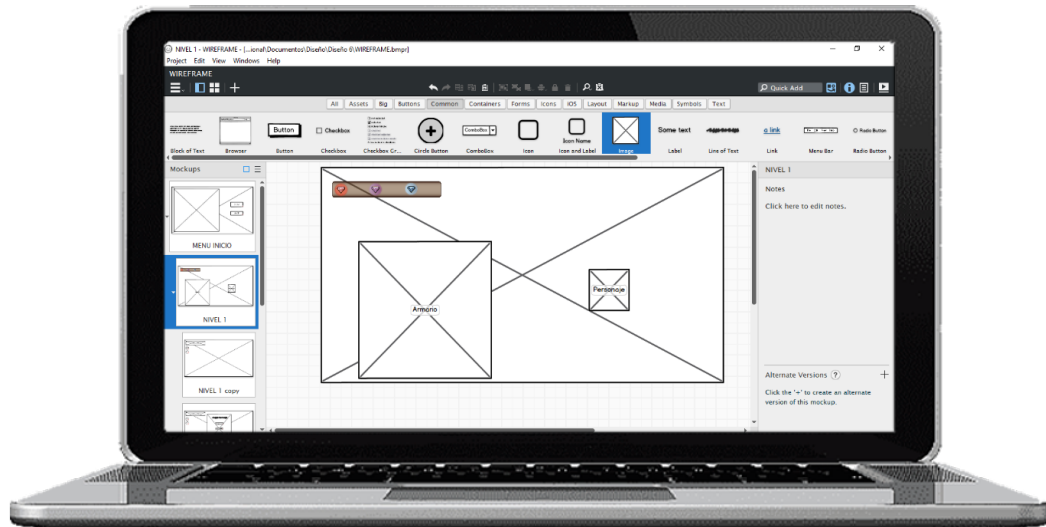


Figura 16 Wireframe de la disposición del primer nivel, elaboración propia

La flexibilidad en el cambio de tamaño de los gráficos a la hora de exportar, facilito el trabajo y en cierta forma redujo el tiempo de este, por ejemplo, cuando se requerían diferentes tamaños de un mismo gráfico.



Figura 17 Movimiento del personaje, elaboración propia.

Para realizar la animación se debía tener cada uno de los movimientos del personaje para de esta forma generar una secuencia, la imagen anterior muestra tal secuencia, para lograr esto se utilizaron las capas de Adobe Illustrator. Se generó una primera imagen en una de las capas, en seguida se dispuso una encima y la capa de abajo se bloqueó, para de esta forma garantizar que no se afectará el trazo anteriormente hecho, luego en la capa actual se generaba un nuevo trazo respecto al movimiento que se quería mostrar.



Figura 18 Proceso de creación del personaje dentro del programa Adobe Illustrator. elaboración propia.

Proceso de codificación y montaje - Unity3D

Después de realizar los respectivos gráficos del ambiente y tener el movimiento del personaje, se exportan las ilustraciones en formato PNG y del tamaño correcto para optimizar el

peso del archivo en Unity, en seguida se importan al programa para así empezar el montaje del videojuego. Después de importar varios gráficos, la persona encargada de la programación comenzó a entender las herramientas de Unity de mejor forma, que eran y cuáles de estas le ayudaban a desarrollar con mayor facilidad un proceso. El docente tutor indica revisar algunos tutoriales sobre cómo usar esta herramienta, además de esto, se tienen en cuenta los artículos de la vasta comunidad que posee Unity. Dada la sugerencia del docente se establece un tiempo de cuatro días para hacer la revisión de la directriz.

Se produjeron errores de programación, edición de elementos y edición específica del proyecto al crear este videojuego. El primer trabajo desarrollado con este software fue la creación de un videojuego 2D para investigar y utilizar la plataforma de desarrollo mencionada anteriormente, el videojuego que muestra la figura (19) fue desarrollada con el fin de practicar e interactuar con el programa, con ayuda de lineamientos otorgados por el servicio de la plataforma Domestica.



Figura 19 Escena de videojuego de practica en Unity, elaboración propia.

A su vez se emplea como base el video juego previamente diseñado y programado por el licenciado David Peña, en el cual se toma como ejemplo para el desarrollo del prototipo, muestra del mismo se encuentra en la figura (20) mostrada a continuación.

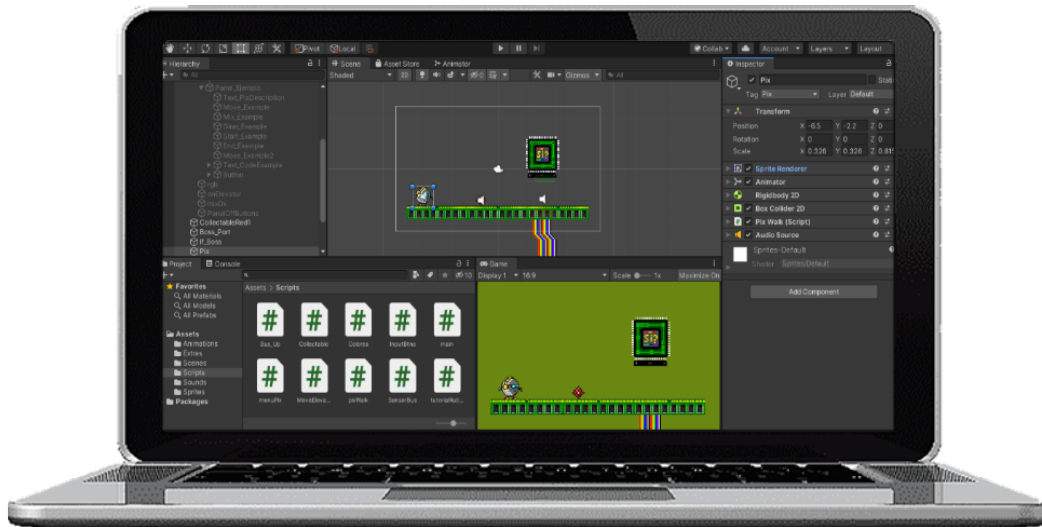


Figura 20 David Peña -Pia A Coding Adventure.

El nivel 1 del videojuego dio comienzo con la creación de las animaciones del personaje, la importación de las ilustraciones que conforman el ambiente y la programación de las diferentes interacciones que posee el personaje con los objetos, la cual se realiza por medio de un *Script*. En la figura (21) se visualizan las carpetas de *scripts* mencionadas.

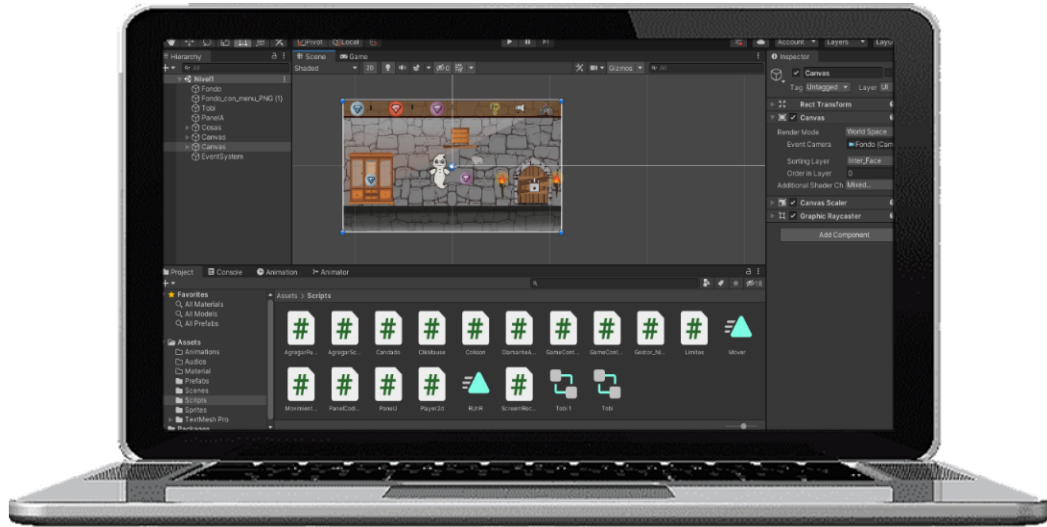


Figura 21 Escena del nivel uno del videojuego, elaboración propia.

La composición grafica de este nuevo videojuego fue creada por la Docente en formación Karla González y la figura (22) refleja el uso del contenido creado con anterioridad en el programa Adobe Illustrator, y también muestra el uso de la herramienta *Sprite Editor*.

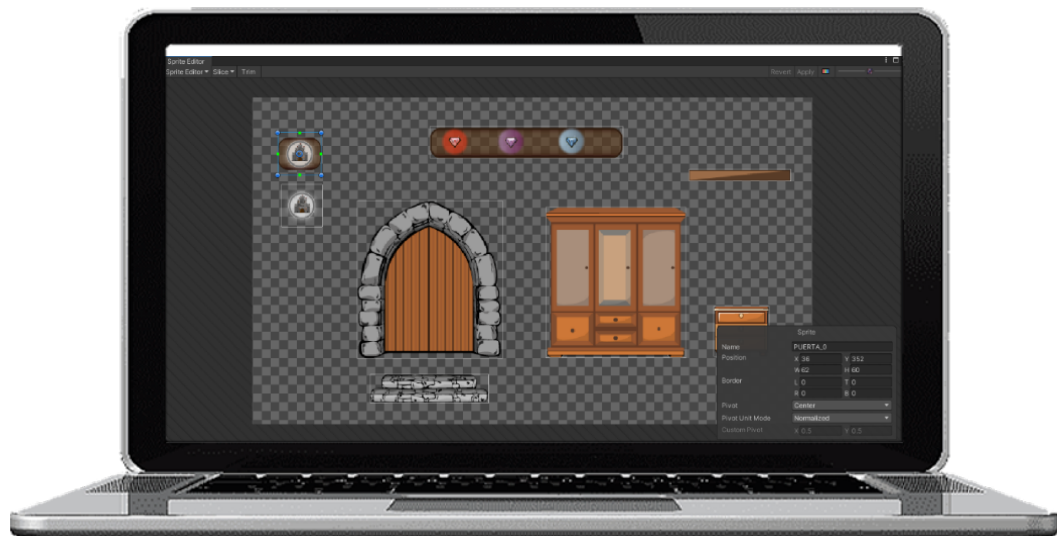


Figura 22 Sprite editor Castle Code, elaboración propia.

Teniendo listo los ambientes de los niveles se procede a colocar los diamantes que servirán de ayuda otorgando pistas para poder salir del nivel, además de eso el diamante rojo cumple la función de informar cuantas vidas, los diamantes funcionan por medio un *Box Collider* que permite al personaje la interacción al momento de colisionar un diamante con el personaje, tal colisión se ve reflejada en un contador, ubicado espacialmente en la parte superior izquierda del escenario, en él también se puede observar la pérdida de vidas ocasionada por no superar el acertijo. En la figura veinte tres (23) se logra observar el contador ubicado dentro del primer nivel.

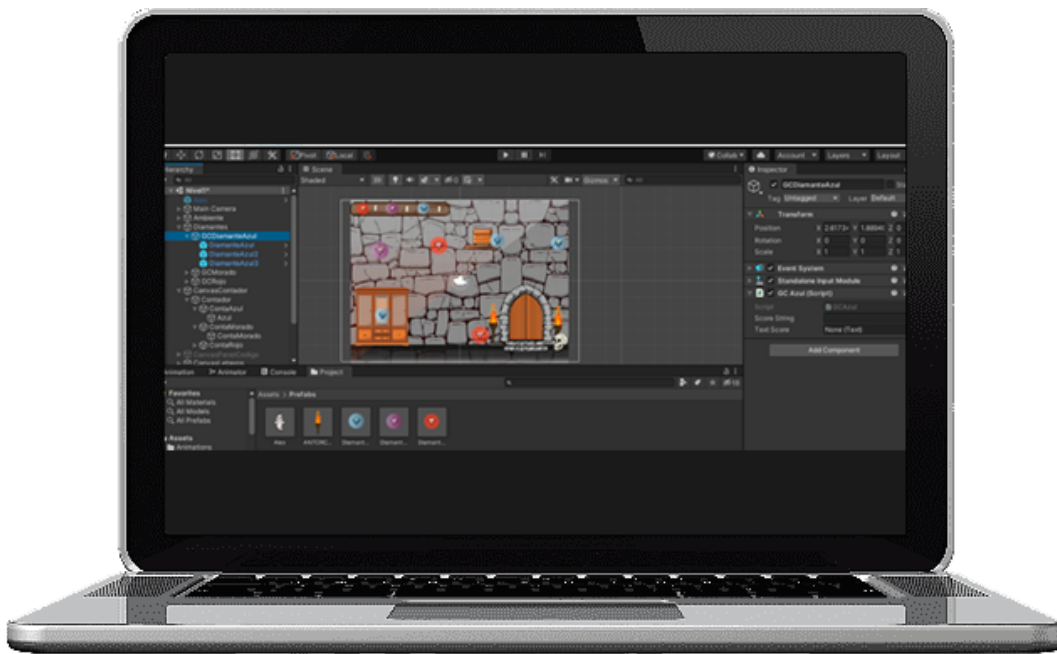


Figura 23 Castle Code Desarrollo, elaboración propia

La herramienta *Animator* de Unity permite realizar la configuración de transición de animaciones, modificar el tiempo de cambio entre las transiciones y asignar si es una animación cíclica o no, esta herramienta facilito el trabajo en ese aspecto a la hora de realizar el proceso del movimiento del personaje, asignando tiempo y ciclos de repetición e interacción con los comandos de movimiento de Alex.

En la figura (25) y (26) se aprecia el *Animation* y el *Animator* con los tiempos entre *Sprites* y la interacción entre la consola y las animaciones.

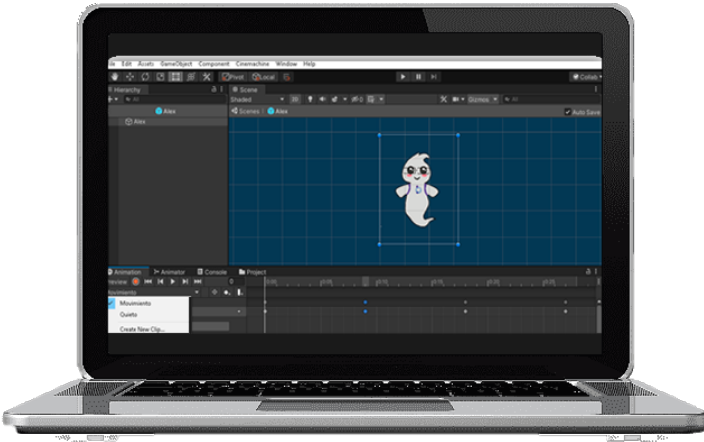


Figura 24 Personaje dentro de la interfaz de Unity, elaboración propia

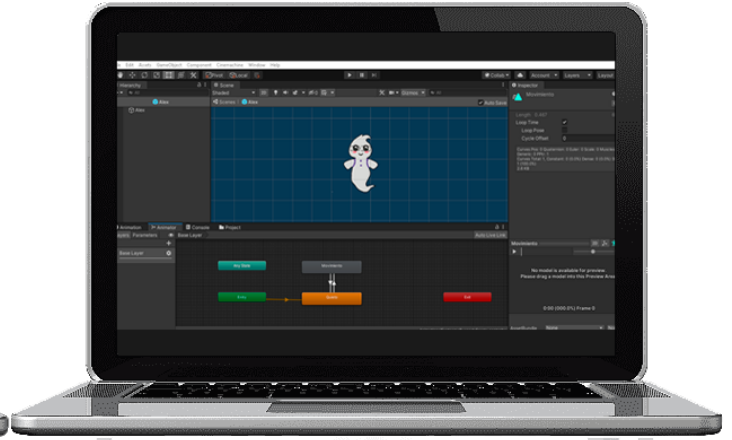


Figura 25 Personaje, muestra del movimiento, elaboración propia

Realizado las animaciones de Alex y sus transiciones se proceden a configurar la herramienta Cinemachine en la cual permitirá mover la cámara por toda la escena siguiendo al personaje Alex, generando el efecto de movimiento, la cámara de la escena se puede visualizar en la figura (26) se observa la cámara y el espacio limitante.



Figura 26 Cámara, herramientas de control, elaboración propia

El siguiente paso luego de tener el personaje y el ambiente programados con sus respectivas interacciones se procede a realizar el *Collider* que se puede ver en la figura (27) el cual será el mediador para pasar de nivel.

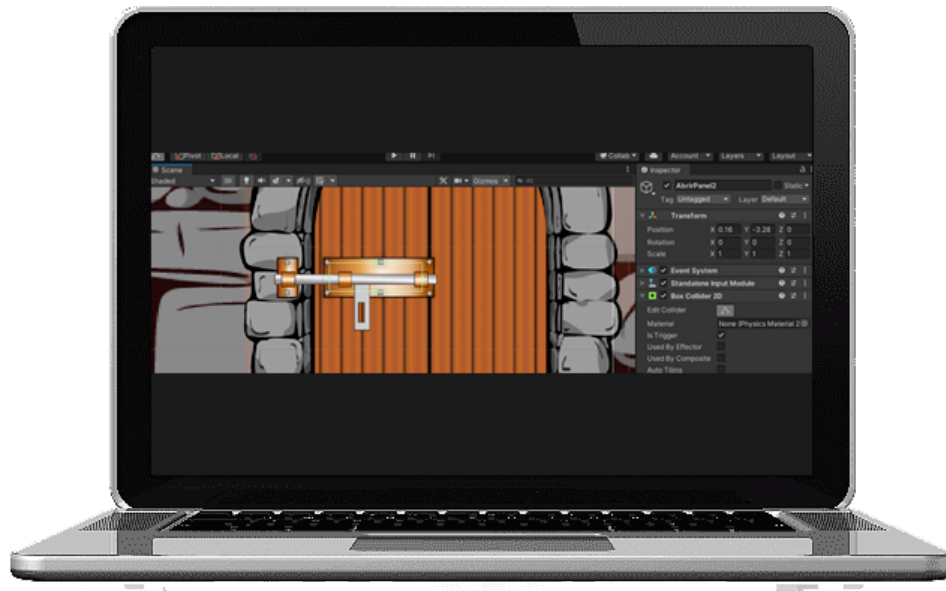


Figura 27 Desarrollo del videojuego, elaboración propia

Al configurar todos los *collider*, se crea un UI (Usuario de Interfaz) Canvas en el cual se va a configurar el panel donde se ejecuta el código, para poder salir del nivel, como lo muestra la figura (28), posee diversos botones el cual cada uno tiene una instrucción que será reflejada en el panel en forma de código, con la condición de que si se tiene bien el acertijo, se pasará al siguiente nivel, de lo contrario como se describió antes, se restara una vida dentro del contador, cabe mencionar que después del último acto, se reiniciara la información contenida en el panel, para de esta forma permitir un nuevo intento.



Figura 28 Desarrollo del panel, elaboración propia.

Visual Studio.

Para generar interacción entre ambiente, personaje contador y panel, se creó un algoritmo en *Visual Studio*, el cual se refleja visualmente en la figura (30); consiste en realizar una pregunta de si (If en programación) está bien escrito el código o si está mal escrito, el algoritmo genera un análisis de cada posibilidad de combinación entre los botones. Se tiene en cuenta que al momento de tener varios botones ejecutando una parte del código se pueden generar diversos códigos al momento de oprimir los botones, sin embargo, se mantiene una estructura en cada código de salida la cual ayuda a la comprensión de la programación y sus componentes.

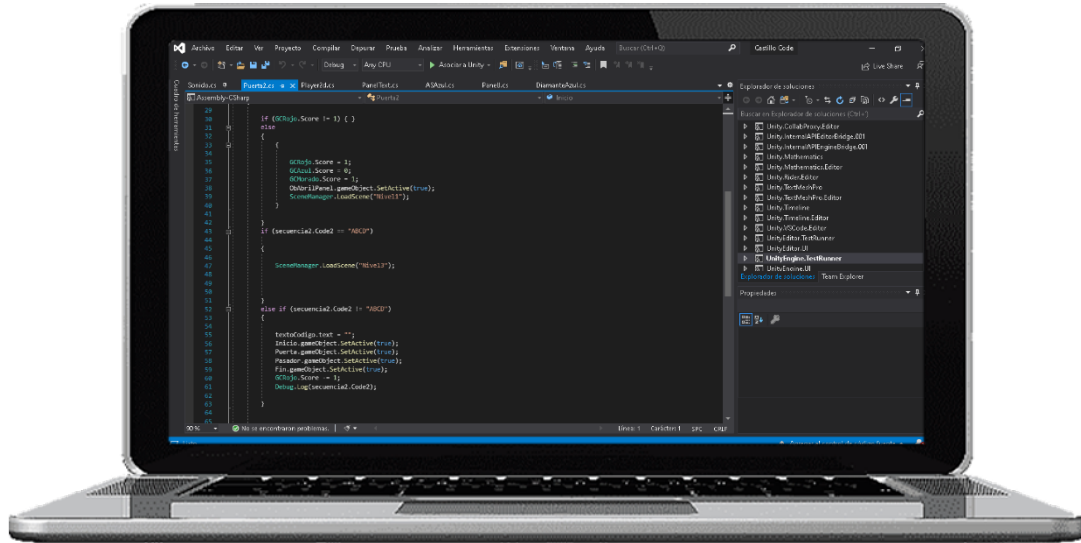


Figura 29 Código Panel, elaboración propia

Al momento de tener diferentes paneles en los niveles, se genera un script, mostrado en la figura (30) para tener un apoyo visual, en el cual se incluyen todas las acciones de los botones, cada botón genera una función diferente la cual se ve reflejada en el panel en forma de texto, este se visualiza en la pantalla del panel, generando un script para cada nivel.

El personaje principal se controla por medio del teclado periférico, esto se hace utilizando las teclas “D” para mover hacia arriba al personaje, la tecla “S” para mover hacia abajo, la tecla “A” para la izquierda y la tecla “D” para la derecha, esto se hace mediante código el cual se asigna al personaje Alex, se ejecutará este script indefinidamente siempre y cuando Alex esté activo en la pantalla. Los diamantes dispersos por la escena están conectados por medio de script al personaje principal Alex, el cual al momento de pasar por encima de ellos y colisionar con los collider, para esto se generaron dos scripts, uno para destruir el diamante, otro para agregar el puntaje en el contador y generar la suma nivel tras nivel, el código de ese Script se representa en la siguiente figura:

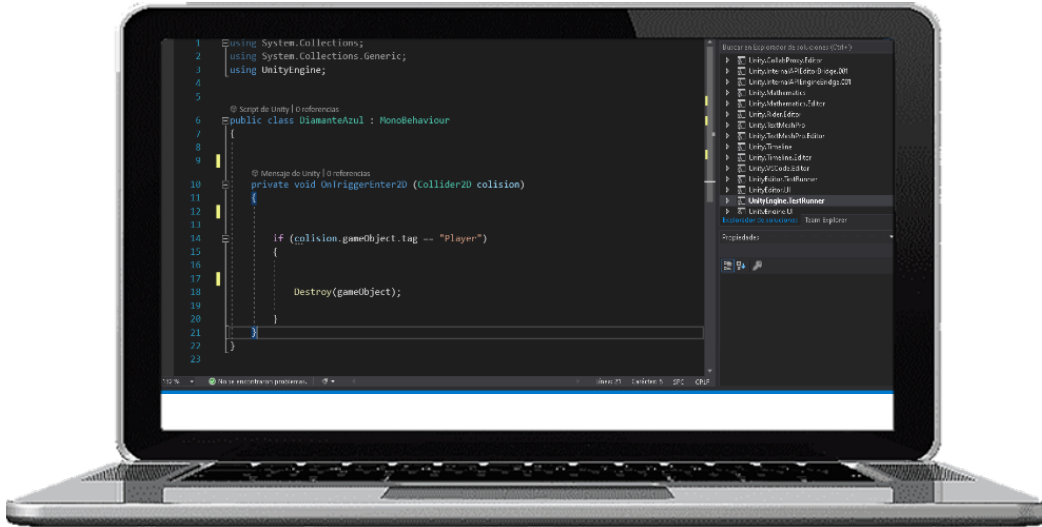


Figura 30 Código Diamante, elaboración propia.

Una vez realizados todos los Sprite, organizado el ambiente, generado las animaciones, se procede a realizar un ejecutable, el cual permitirá acceder al videojuego en cualquier computador que use el sistema operativo Windows.

Se consta de 10 niveles en el que cada uno contienen diferentes metodologías de programación, los primeros cinco niveles se centran en los comandos directos que se pueden generar en un código de programación, a su vez en la importancia de seguir instrucciones y de establecer un orden dentro del código, el cual simboliza que toda operación debe tener un principio y un fin.

Los niveles del 6 en adelante tratan sobre conceptos de programación exacto como los If , While, o variables, estos niveles cuentan con la modificación de que el panel se verá en pantalla todo el tiempo, como se muestra en la figura (33) esto con el fin de tener un poco más de interacción con el ambiente, esta modificación se realiza para que el jugador pueda visualizar mucho mejor cómo se ejecutó un código, haciendo uso de preguntas específicas y variables, se introduce el concepto de nombre de variable y cómo afecta la ejecución de un código también, se muestra cómo utilizar correctamente un While.



Figura 31 Paneles Nivel 7 muestra de la instrucción a cumplir, guardar una variable, Castle Code, elaboración propia

Capítulo 5. Conclusiones

El desarrollo del presente trabajo de grado tuvo como objetivo diseñar y desarrollar un prototipo de videojuego, didácticamente pensado, que facilite el aprendizaje de algunos conceptos importantes relacionados con la codificación y la programación como lo son las secuencias lógicas simples, los condicionales y los ciclos o bucles, haciendo uso del pensamiento algorítmico. El aspecto gráfico influye en gran medida en la población a la cual va dirigido, en este caso la población son niños de diez años o más (10+), en esas edades es importante un diseño de ilustraciones de personajes, escenarios y ambientación que resulten atractivas y generen curiosidad al jugador, lo anterior se logra corroborar en los resultados de la encuesta efectuada, donde el ítem de las ilustraciones fue uno de los más relevantes y comentados por los jugadores, en el cual daban a conocer su punto de vista, tanto de lo positivo como de lo negativo, generando así una retroalimentación que permitió orientar y generar continuamente mejoras.

El error tiene lugar en cada una de las tareas que los niños realizan a diario, resultados de la encuesta efectuada sugieren que las pistas de orientación para cada nivel a través de los diamantes, es una buena estrategia, estos ayudan a que el equivocarse se pueda prevenir, siempre y cuando el jugador realice una buena interpretación de la pista otorgada.

El desarrollo de videojuegos orientados al sector educativo puede ser de gran relevancia conociendo a priori las necesidades de la población objetivo ya que pueden estar muy distantes de lo que se imagina el equipo de desarrolladores.

La elección de Unity 3D como motor de renderización para el videojuego permitió integrar diversas funcionalidades a partir de los scripts dispuestos en los objetos que componen cada uno de los niveles, aportando gran variedad en las acciones y funcionalidades del juego. Sin

embargo, la integración con los gráficos a partir de píxeles genero conflictos al momento de trabajar con imágenes en alta resolución, por lo que utilizar otros motores enfocados en gráficos vectoriales puede permitir mayor libertad a la hora de aplicar diseños más elaborados.

Por último es importante resaltar la valiosa oportunidad de diseñar y construir un videojuego desde cero aplicando diversos conocimientos adquiridos a lo largo de los espacios académicos que brinda la Universidad Pedagógica Nacional en las carreras de Licenciatura en Electrónica y Diseño Tecnológico, permitiendo la construcción de una herramienta didáctica mediante un ejercicio interdisciplinar que reúne conocimientos y habilidades que conforme avanza el proyecto se fueron profundizando progresivamente.

Como trabajo a futuro cabe la posibilidad de añadir una capa de personalización (por ejemplo, el orden de los niveles o el grado de dificultad de estos) lo que permite una adaptación por parte del equipo docente que aplicara la herramienta. Por otra parte, se evidencio que permitir al jugador la personalización del personaje principal incrementa la motivación e interés por el juego ya que se eleva el nivel de compromiso al sentirse parte del juego y no solamente el que guía una marioneta.

También se propone, ampliar la secuencia de los niveles, dado que se realizaron con temas específicos, generando una brecha de subtemas necesarios para la comprensión total de estos conceptos. Lo anterior puede permitir la adaptabilidad al contexto construyendo diferentes niveles de profundización según sea el caso, retomando las observaciones de los docentes que aplicaron la herramienta en el aula y observaron que dependiendo del nivel de comprensión cada estudiante puede requerir más o menos interacción con el juego.

Asimismo, es posible crear versiones del juego aplicando un lenguaje de programación específico e.g., Java, Python o C++ para todos los niveles, con la intencionalidad de que el

estudiante apropie el lenguaje utilizando los conocimientos previos adquiridos con la versión genérica. De igual manera con el creciente uso de supercomputadores a nivel mundial para manejar grandes volúmenes de datos, la posibilidad de crear niveles que aborden temáticas más complejas como computo paralelo o distribuido o lenguajes de alto nivel como OpenMP puede aportar en gran medida ya que al momento de escribir este documento no existe un juego en el mercado que aborde este tipo de conceptos.

Referencias

- Alvino, C. (2021). *Colombia*. We Are Social y Hootsuite. <https://branch.com.co/marketing-digital/estadisticas-de-la-situacion-digital-de-colombia-en-el-2020-2021/>
- Baer, R. H., Rush, W. T., & Harrison, W. L. (1972). *Television gaming apparatus and method*. <http://www.freepatentsonline.com/3659285.html>
- Belli, S., & López, C. (2008). Breve historia de los videojuegos: Simone Belli, Cristian López. *Athenea Digital: Revista de Pensamiento e Investigación Social*, 179(14), 159–179.
- Bennet, A. (2013). Long-Term Effects of Conservation Training with Educationally Subnormal Children. *Journal of Special Education*, 8(3), 237–245. <https://doi.org/10.1177/002246697400800304>
- Claudio Peña. (2017). *Programación para niños* (M. Duarte (Ed.); Primera ed).
- CodeCombat. (2014). *Code Combat*. <http://codecombat.com/about#>
- Csizmadia, A., Curzon, P., Humphreys, S., Ng, T., Selby, C., & Woollard, J. (2015). *Pensamiento Computacional Guía para profesores*. 1–17.
- ECOTEC. (2016). *Lógica Computacional*.
- Fandom. (2020). *Rabbids _ Coding! _ Raving Rabbids Wiki _ Fandom*.
- Fuentes, A. (2014). *Desarrollo de un videojuego de aventuras en C# sobre Unity*. 1–57.
- Gallego Durán, F., Villagrà-Arnedo, C., Satorre Cuerda, R., Compañ Rosique, P., Molina Carmona, R., & Llorens Largo, F. (2014). Panoràmica: serious games, gamification y mucho más. *ReVision - A Journal of Consciousness and Transformation*, 7(2), 2.
- Hero, C. (2021). *yermis*. Coursera Hero. <https://www.coursehero.com/file/79683760/yermisdocx/>
- ISSUU. (2019). Arcade. *Revista Electronica*, 99. <https://issuu.com/jpom2001118/docs/e5->

ortegajohan-fusionado

Jung, C. (2016). 프로그래밍 교육을 위한 기능성 게임의 설계와 구현 ※. *16(6)*, 143–150.

Kahn, K. (1999). A Computer Game to Teach Programming. *NECC'99: Proc. Ntl. Educational Computing Conf.*, 127–135.

Kazimoglu, C., Kiernan, M., Bacon, L., & Mackinnon, L. (2012). A Serious Game for Developing Computational Thinking and Learning Introductory Computer Programming. *Procedia - Social and Behavioral Sciences*, *47*, 1991–1999.

<https://doi.org/10.1016/j.sbspro.2012.06.938>

LEGO. (2020). *lego historia*. <https://www.lego.com/es-es/aboutus/lego-group/the-lego-group-history/>

López, C. (2016). El videojuego como herramienta educativa. Posibilidades y problemáticas acerca de los serious games The video game as an educational tool. Possibilities and problems about Serious Games López Raventós. *Revista Apertura*, *8(1)*, 1–10.

Los Policias y Ladrones - Los juegos clásicos. (2019). Los Juegos Clasicos.

<https://sites.google.com/site/juegosclasicos2/policias-y-ladrones>

Microsoft Prensa. (2015). *Momentos destacados en la historia de Microsoft - Centro de noticias*.

Abril 6. <https://news.microsoft.com/es-es/2015/04/06/historia-microsoft-40-aniversario/>

Monkey, C. (2020). *Codificación para niños _ Programación basada en juegos _ CodeMonkey*.

Morán, P. (2007). En El Aula. In *Hallazgos-Revista de Investigaciones* (Vol. 5, Issue 1986).

<https://doi.org/10.1017/CBO9781107415324.004>

Murga, C. (2021). *Redes sociales*. Lucerna Iuris et Investigatio.

<https://doi.org/10.15381/lucerna.v0i1.20137>

Prensky, M. (2001). The Games Generations: How Learners Have Changed. *Computers in*

Entertainment, 1(1), 1–26. <http://portal.acm.org/citation.cfm?doid=950566.950596>

Roncancio, Ortiz-Carrera, M. F., Llano, H., Malpica, M., & Bocanegra, J. J. (2017). EL USO DE LOS VIDEOJUEGOS COMO HERRAMIENTA DIDÁCTICA PARA MEJORAR LA ENSEÑANZA-APRENDIZAJE: UNA REVISIÓN DEL ESTADO DEL TEMA The Use of Video Games as a Teaching Tool to Improve Teaching-Learning: State of the Art Review. In *Revista Ingeniería, Investigación y Desarrollo* (Vol. 17, Issue 2). Universidad Javeriana.

Ruiz, Josu Ahedo, I. (2014). Vii Las Nuevas Tecnologías Como Herramientas Que Facilitan La Educación Formativa En La Educación. *Estrategias Innovadoras Para La Docencia Dialógica y Virtual*, 25–40. [http://www.seeci.net/cuiciid2013/pdfs/unido mesa 2 docencia.pdf](http://www.seeci.net/cuiciid2013/pdfs/unido_mesa_2_docencia.pdf)

Sanchez, A. (2019). *Juegos Tradicionales*. <https://juegostradicionales.net/bolos/>

Tecnosfera. (2015). *tecnosfera*. <https://www.eltiempo.com/archivo/documento/CMS-16389650>

Torrente, J., Moreno-Ger, Pablo Manjón, B., & Sierra, J. L. (2008). Instructor-oriented authoring tools for educational videogames. *Proceedings - The 8th IEEE International Conference on Advanced Learning Technologies, ICALT 2008*, 516–518. <https://doi.org/10.1109/ICALT.2008.177>

Trejos Buriticá, O. I. (1999). *La esencia de la lógica de programación*.

<https://ingenieriadesistemasunc.blogspot.com/2016/11/la-esencia-de-la-logica-de-programacion.html>

Weforum. (2020). *futuro*. <https://www.weforum.org/reports/the-future-of-jobs-report-2020>

Anexos

Organismo	Licenciatura en electrónica		
Proyecto	CASTLE CODE - PROTOTIPO DE VIDEOJUEGO EDUCATIVO COMO MATERIAL DIDÁCTICO PARA EL APRENDIZAJE DE LA PROGRAMACIÓN		
Entregable	Ejecutable		
Autor	David Moreno Ortiz, Karla Nicol Gonzales Barahona		
Versión/Edición	1.0	Fecha Versión	16/06/2021
Aprobado por	David Peña Morales	Fecha Aprobación	
		Nº Total de Páginas	

REGISTRO DE CAMBIOS

Versión	Causa del Cambio	Responsable del Cambio	Fecha del Cambio
1.0	Versión inicial	David Moreno Ortiz, Karla Nicol Gonzales Barahona	16/06/2021

CONTROL DE DISTRIBUCIÓN

Nombre y Apellidos
<Nombre Apellido1 Apellido2>

- **Descripción de Scripts**



Figura 32 *Carpetas de Scripts, elaboración propia*

Los Scripts se realizan en el programa *Visual Basic*, el video juego se desarrolló de tal manera que se realizaron pequeños scripts para cada una de las actividades o acciones de interacción dentro del videojuego, para acceder a las carpetas que contiene la información se siguen las siguientes pautas las cuales se componen de tres partes:

- La declaración de las librerías a usar.
- La declaración de variables que se usarán en el *Scripts*.
- Los diferentes procesos de la expresión *void*.

pautas

- El nombre de los Script siempre empieza con mayúscula la primera letra.
- Las variables siempre comienzan con mayúscula en su nombre.
- La función *Void Start* se ejecutará al momento de cargar un nivel
- Al colisionar *OnTriggerEnter2D*, la comparación entre el *collider* de los objetos y el personaje se debe realizar por medio del Tag “Player”

- **Game Document Design**

JUNIO 2021

<CASTLE CODE>

GDD

Plantilla de GDD elaborada por Jimmy Andrés Rodríguez basada en © 2017 Game Dev Underground. Free to use/modify/distribute under CC 4.0.

Título: Castle Code

Desarrollador: David Moreno Ortiz, Karla Nicole González Barahona

Plataforma: Pc

Audiencia: +10 /género/intereses.

Parte 1 Concepto Inicial:

Identidad del Juego / Mantra:

Juego de estilo plataformas/aventura de un fantasma

Pilares del Diseño:

Educación, Intriga, Diversión

Resumen de Género/Historia/Mecánicas:

Describe de qué va el juego en términos de jugabilidad y/o historia.

Este juego utiliza la exploración que debe hacer un fantasma en varias habitaciones de un castillo, para encontrar pistas recogiendo los diamantes dispersos en el cuarto, que lo ayuden a salir de él, por medio del proceso de abrir una serie de puertas por medio de códigos.

Características:

Enumera las características o elementos únicos que quieres incluir en tu juego.

- Lógica de programación
- Intriga.
- Aventura.

Interfaz:

Los niveles se desarrollarán en diferentes cuartos del castillo, donde el personaje principal deberá recoger una serie diferentes de pistas, representados por tres clases de diamantes, los cuales poseen 3 diferentes tipos de ayudas.

El personaje es movido por el jugador con ayuda del teclado, ya sea usando las teclas A W D S, asociadas con los movimientos abajo, izquierda, derecha, por la habitación para recolectar una serie de diamantes, los cuales están divididos en tres opciones de colores, los diamantes rojos son la cantidad de oportunidades que tiene el jugador para intentar abrir la puerta resolviendo el problema propuesto, el diamante “violeta-morado” son las ayudas o pistas que tendrá el jugador para resolver el problema en niveles próximos y el diamante color azul es una ayuda para próximos niveles.

Estilo de Arte:



Figura 33 Ambiente Juego Extraído De Internet



Figura 34 Juego Aladino -Extraído De Internet



Figura 35 Juego Mario - Extraído De Internet



Figura 36 Video Juego Plataforma - Extraído De Internet



Figura 37 Juego Scape Room - Extraído De Internet



Figura 38 Castle Code - Fuente Propia

Música/Sonido:

La Música utilizada para ambientar es tipo de genero Novelty, Electrónica con track Halloween, para el sonido de los botones de los paneles se utiliza sonido de piedras rozándose.

Parte 2 Desarrollo:

Diseño Visual

Personaje/Objeto/Protagonista/Héroe:

El personaje principal, es un fantasma muy inteligente, aventurero llamado “Alex Phantom” cuenta con una personalidad amigable; que se encuentra en busca de restos para aprender, de las situaciones que le suceden.

Diseño personaje/objeto



Figura 39 Personaje Alex - Fuente Propia

Nivel/Escenario/Cuarto

El tema por desarrollar en el primer nivel será la concepción de que todo proceso tiene un inicio y un fin.

Desarrollo del primer nivel

La dinámica del juego se compone en dos partes, la primera se centra en recolectar los diamantes, la segunda parte es de “pensamiento lógico” en esta, el jugador resuelve un acertijo

oprimiendo los botones de un panel, los cuales debe oprimirlos en un orden lógico correcto para así formar un código que permitirá abrir la puerta.

De forma más descriptiva tenemos:

El jugador se encontrará con un panel el cual tiene 3 botones, el primero de ellos tiene la palabra Inicio, esta palabra estará directamente conectada a la pista de uno de los diamantes que se refiere a que todo proceso o actividad que se realice debe tener un inicio. El segundo botón tiene la palabra proceso o actividad, esta se refiere a los procesos o pasos que deben tener los algoritmos o secuencias que se realizan para hacer determinada actividad en este caso abrir la puerta, por último, el tercer botón tiene la palabra fin, hace referencia al cierre o a la terminación de la actividad que se realiza o se pide para poder abrir la puerta, eso también tiene la semejanza directa con las terminaciones de los códigos en programación.

Cada vez que el jugador oprime un botón se mostrará en el panel un mensaje con su respectiva acción en texto, este mensaje está relacionado con el proceso que se debe seguir para poder quitar los bloqueos de la puerta y lograr el objetivo de abrirla, para ello se deben oprimir los botones en un orden correcto, donde se evidenciara una secuencia lógica de pasos donde si es correcta tal secuencia, al oprimir el botón de start o Run se abrirá la puerta si es incorrecta, se quitará una vida de lo contrario, esta última representada por el de diamantes rojos encontrados y recogidos en el cuarto, al terminar todas las vidas deberá reiniciar el nivel.

Los niveles consecuentes hasta el 5 se utiliza la misma metodología en los paneles para salir de los cuartos, se agrega la dificultad, con nuevos botones, estos niveles se enfocan en instrucciones directas. Los Niveles 6 a 10 se ingresa la opción de variables para realizar operaciones en las funciones explicadas.

Diseño escenario



Figura 40 Castle Code - Fuente Propia



Figura 41 Castle Code Nivel 1 - Fuente Propia

Flujo de Juego

En esta sección se detalla el transcurso de una partida típica. Se comentarán los pasos a seguir del Jugador desde el arranque del juego hasta completar un nivel y pasar al siguiente. De esta manera detallaremos el funcionamiento exacto del juego.

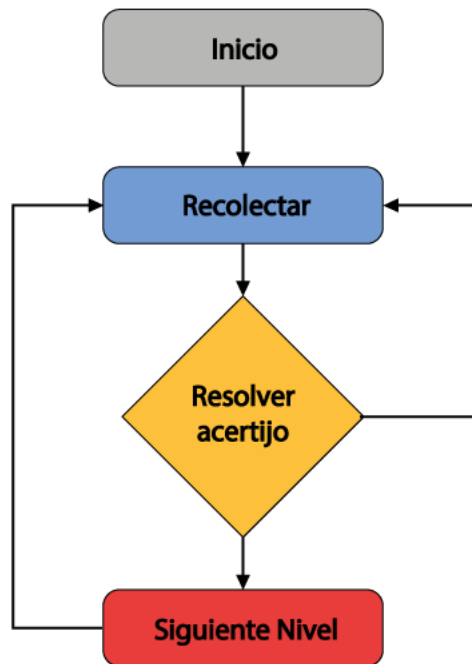


Figura 42 Diagrama Flujo Niveles -Fuente Propia.

Interfaz

Diagrama de flujo de las pantallas que va a tener el videojuego.

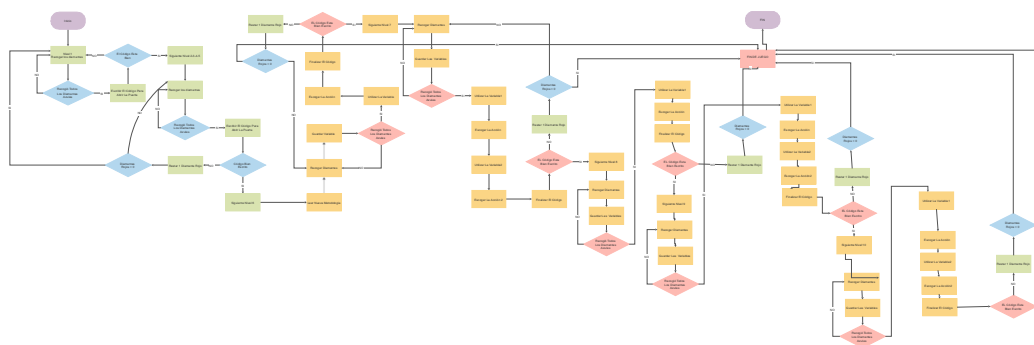


Figura 43 Diagrama De Flujo- Fuente Propia.

Menú principal

Descripción gráfica del menú principal.

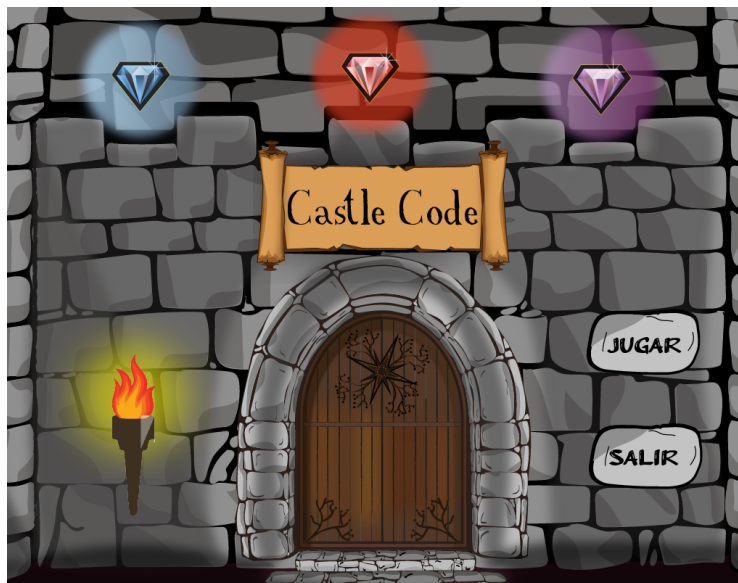


Figura 44 Castle Code Menú -fuente propia.