

**DESARROLLO DE UN VIDEOJUEGO COMO ESTRATEGIA METODOLOGICA
PARA EL APRENDIZAJE DE ALGORITMIA Y PROGRAMACIÓN EN LA
UNIVERSIDAD DE CUNDINAMARCA SEDE CHIA**

**TRABAJO DE GRADO PARA OPTAR EL TITULO DE INGENIERO DE
SISTEMAS**

**ANDRÉS FELIPE CANO RODRÍGUEZ
CÓDIGO: 561213210**

**LEIDY VANESSA SÁNCHEZ NUÑEZ
CÓDIGO: 461209133**

**UNIVERSIDAD DE CUNDINAMARCA EXTENSIÓN CHIA
PROGRAMA DE INGENIERÍA DE SISTEMAS
FACULTAD DE INGENIERÍA**

2018

**DESARROLLO DE VIDEOJUEGO COMO ESTRATEGIA METODOLOGICA
PARA EL APRENDIZAJE DE ALGORITMIA Y PROGRAMACIÓN EN LA
UNIVERSIDAD DE CUNDINAMARCA SEDE CHIA**

ANDRÉS FELIPE CANO RODRÍGUEZ

CÓDIGO: 561213210

LEIDY VANESSA SÁNCHEZ NUÑEZ

CÓDIGO: 461209133

DIRECTOR

M.Sc. Ing. MISAEL FERNANDO PERILLA BENITEZ

**UNIVERSIDAD DE CUNDINAMARCA EXTENSIÓN CHIA
PROGRAMA DE INGENIERÍA DE SISTEMAS
FACULTAD DE INGENIERÍA**

2018

AGRADECIMIENTOS

A los docentes Misael Fernando Perilla, Jairo Márquez Díaz, que han creído en nuestro proyecto y a través de sus enseñanzas, consejos y conocimientos han contribuido el desarrollo de este proyecto.

A nuestras familias y amigos que nos han apoyado para el desarrollo del videojuego.

DEDICATORIA

A todas aquellas personas que tienen un gran interés en el mundo de los videojuegos, donde por medio de la interacción e imaginación, sueñan con ser partícipes en la creación y desarrollo de un juego, a través de este proyecto, les presentamos que es posible por medio del interés y dedicación llevar a cabo la elaboración de este tipo de proyecto.

RESUMEN

El desarrollo de un videojuego como estrategia metodológica para el aprendizaje de algoritmia y programación en la Universidad de Cundinamarca Sede Chía, fue elaborado bajo la metodología SCRUM diseñada para un proceso iterativo e incremental bajo entornos basados en el desarrollo ágil del software.

Este videojuego se creó para dar un refuerzo a los estudiantes del programa Ingeniería de Sistemas, para los núcleos temáticos de lógica y algoritmia, así como los de programación I, tomando como eje central el proceso de aprendizaje que han venido desarrollando, que por diferentes aspectos en algunas ocasiones ha generado un tipo de retraso en el desarrollo de estas asignaturas y sus consiguientes.

Por esta razón se pretendió contribuir a los estudiantes con el refuerzo de conceptos básicos de la programación como variables, constantes, instrucciones, sentencias, y la algoritmia; estableciendo el uso de este videojuego como parte de una estrategia pedagógica.

El videojuego se desarrolló en la plataforma de Unity, haciendo uso del lenguaje C# para facilitar el uso de scripts, además de una base de datos donde se almacena la información característica del jugador para su respectivo proceso en el juego. La dinámica del juego trata de un ambiente aventurero donde se manejan una serie de obstáculos acompañados de unas preguntas que le permitirán al jugador avanzar en el juego de una manera llamativa durante todo el proceso; a medida que se va avanzando lógicamente se aumenta el grado de dificultad.

Palabras Clave:

Algoritmia – Aprendizaje – Estrategia – Estructura – Metodología – Videojuego.

ABSTRACT

The development of a videogame as a methodological strategy to algorithms and programming learning, in the Universidad de Cundinamarca, has been made under a SCRUM methodology, designed for an iterative and incremental process under environments based on agile software development.

This videogame was created for helping the Systems Engineering students in the thematic cores of “logic and algorithmics” and “programming” subjects, by reducing the difficulties that they have in relation to the learning process, and therefore, making them get behind in the career.

In this way, it was pretended to contribute to the students reinforcing the basic concepts of programming, just like: variables, constants, instructions, sentences and algorithms; establishing the use of this videogame as part of a pedagogic strategy.

The videogame was developed in Unity, using the C# language to ease the writing of scripts, and in addition. The game mechanic is about a building where there are two levels. In the first floor, the player moves forward through rooms answering questions. The second floor is similar, but instead of rooms, the player should jump over different platforms, where the questions are, making high the difficulty.

Keywords:

Algorithmics – Learning – Strategy – Structure – Methodology - Videogame

TABLA DE CONTENIDO

1. PROBLEMA. _____	11
1.1. Planteamiento del problema. _____	11
1.2. Formulación del problema. _____	12
2. OBJETIVOS. _____	13
2.1. Objetivo general. _____	13
2.2. Objetivos específicos. _____	13
3. ALCANCES Y LIMITACIONES. _____	14
3.1. Alcances. _____	14
3.2. Limitaciones. _____	14
4. JUSTIFICACIÓN. _____	15
5. LINEA DE INVESTIGACIÓN. _____	17
6. MARCO TEÓRICO. _____	18
6.1. Marco referencial. _____	18
6.2. Marco conceptual. _____	21
6.3. Margo ingenieril. _____	24
7. METODOLOGÍA. _____	30
8. DESARROLLO DEL PROYECTO . _____	33
8.1. Desarrollo de la metodología. _____	33
8.2 Costo del proyecto. _____	57
9. PRUEBAS. _____	58
10. CONCLUSIONES. _____	62
11. RECOMENDACIONES. _____	63
12. PROYECCIONES. _____	64

LISTA DE FIGURAS

<i>Ilustración 1. Aplicación encuesta a los estudiantes de Lógica de programación y Programación 1 de la universidad de Cundinamarca extensión chía.</i>	25
<i>Ilustración 2. Creación del terreno del juego en Unity.</i>	27
<i>Ilustración 3. Selección de la textura que se aplicara al terreno del juego.</i>	27
<i>Ilustración 44. Autenticación del usuario.</i>	61
<i>Ilustración 45.Registro de un nuevo usuario.</i>	61
<i>Ilustración 47. Puntuación en cero.</i>	62
<i>Ilustración 45. Representación de respuestas a la pregunta 1 de la encuesta.</i>	72
<i>Ilustración 49. Adición de texturas para el terreno.</i>	75
<i>Ilustración 53. Adición de más texturas.</i>	77
<i>Ilustración 54. Inserción de los árboles para el terreno.</i>	77
<i>Ilustración 55. Incorporación de obstáculos al terreno.</i>	78
<i>Ilustración 56. Utilización de más Asset.</i>	78
<i>Ilustración 57. Uso de Asset Rocks Free Pack.</i>	78
<i>Ilustración 58. Uso del Asset Block Creator Lite.</i>	79
<i>Ilustración 59. Uso del Asset Cartoon Temple Building Kit Lite.</i>	79
<i>Ilustración 64. Diseño de los botones para el Login.</i>	81
<i>Ilustración 65. Base de datos del juego.</i>	82
<i>Ilustración 66. Creación de las preguntas.</i>	82
<i>Ilustración 67. Botones con la respuesta.</i>	83
<i>Ilustración 68. Diseño de la estructura del juego.</i>	91
<i>Ilustración 69. Preguntas a desarrollar para aumentar el puntaje.</i>	91

LISTA DE TABLAS

Tabla 1. CU01 Inicio del Juego. _____	35
Tabla 2. CU02 Menú Registrarse. _____	36
Tabla 3. CU03 Menú de Inicio de Sesión. _____	37
Tabla 4. CU04 Del Juego. _____	38
Tabla 5. CU05 Salir del Juego. _____	39

LISTA DE ANEXOS

Anexo A. Encuesta Dificultades En La Lógica De Programación. _____	68
Anexo B. Elaboración del videojuego. _____	71
Anexo C. Artículo Científico. _____	86

CAPITULO 1

INTRODUCCIÓN

En la Universidad de Cundinamarca extensión chía para el programa de Ingeniería de Sistemas, se pretende implementar un videojuego como estrategia metodológica para apoyar el proceso de aprendizaje sobre conceptos básicos de programación y algoritmia tanto para los estudiantes de las asignaturas de lógica y algoritmia como de programación I, con este videojuego se busca mitigar las dificultades que se presentan en el desarrollo de las asignaturas mencionadas antes debido a las falencias que presentan algunos estudiantes por diferentes aspectos que van desde la falta de compromiso con el núcleo académico, una mala interpretación de los contenidos o el uso de metodologías de aprendizaje-enseñanza que hacen de este tipo de materias una experiencia traumática que no causa el efecto positivo esperado al momento de poner en práctica lo visto en clases, interrumpiendo con la culminación satisfactoria de las asignaturas.

Es por esto que se materializó la idea de crear un videojuego que incluyera conceptos básicos de programación, que les facilite a los estudiantes su desarrollo académico; este documento es el resultado del trabajo realizado en el último periodo en cuanto al análisis, diseño e implementación de un videojuego como estrategia metodológica, para poder contribuir en el programa de Ingeniería de Sistemas de la Universidad de Cundinamarca extensión chía con una aplicación que preste un servicio en el proceso de aprendizaje.

Al finalizar este proceso el resultado esperado es que la Universidad habilite este método para que sea incorporado en el desarrollo del aprendizaje de los estudiantes y de esta forma contribuir en la práctica de este tipo de materias haciéndolo de una manera más dinámica y divertida para los mismos reforzando las bases obtenidas, así mismo generar un material de apoyo a los docentes que imparten estas materias y puedan contar con el respaldo de nuevas herramientas en la institución. Además de aportar a la universidad el uso de estos nuevos mecanismos de ayuda.

1. PROBLEMA

1.1 Planteamiento del problema

La programación y la algoritmia son dos temas básicos para los estudiantes de las carreras de Ingeniería de Sistemas y ciencias de la computación, convirtiendo a la capacidad de crear soluciones mediante el diseño, desarrollo e implementación de estos mismos una de las cualidades más buscadas por el mercado profesional actual, es por esto que muchos de los núcleos temáticos de las universidades poseen una fuerte carga educativa orientada a cubrir estas habilidades, sin embargo, siendo estas materias la base fundamental dentro de la carrera, se hace necesario hallar un método que resulte para los estudiantes más sencillo su aprendizaje.

Resaltando la importancia que tienen estos temas no solo en un contexto educativo sino en la vida cotidiana, ya que permite resolver problemas, generando un pensamiento analítico que se desarrolla de manera específica como una capacidad como programadores y como profesionales, aumentando el grado de análisis para llevar a cabo procesos y resolver asertivamente diferentes tipos de problemas.

En las asignaturas relacionadas con el desarrollo de software, el no poder entender la lógica computacional de uno o varios lenguajes, así como la lógica proposicional y booleana, conlleva a no poder desarrollar soluciones a los problemas dados, estancando a los alumnos, con el fin de identificar los motivos que pueden generar dificultades en el proceso de aprendizaje de la lógica y programación, se realizó una breve encuesta con los alumnos de los núcleos temáticos Lógica & Algoritmia y Programación I correspondientes a las jornadas diurna y nocturna de la Universidad de Cundinamarca extensión chía en el mes de febrero de 2018, con una muestra de 60 estudiantes. (Esta encuesta se encuentra en el anexo I), se denota que muchos alumnos afirman no contar con buenas bases por diversos aspectos, así como la falta de interés ante metodologías ya definidas; lo que les implica llevar un proceso de aprendizaje en los estudiantes más lento, impidiendo de esta manera que el estudiante avance en la carrera de una manera proactiva y muchas veces cronológica, por ello se diseñó este proyecto ya que si no se actúa de forma correctiva a tiempo el futuro profesional al momento

de enfrentarse al mundo laboral puede sentirse frustrado al no manejar la capacidad requerida en este tipo de áreas que lo van ayudar no solo en su crecimiento profesional si no personal al desenvolverse con mayor confianza en este campo. Además de verse afectada su parte económica ya que si se está invirtiendo en su formación y no se ve reflejado al momento de ponerlo en práctica se genera un sentimiento de desperdicio no solo económico, también de tiempo por no obtener un resultado adecuado y así generando un descontento y poca credibilidad con lo que hace, reduciendo de esta manera su oferta en el campo laboral.

1.2 Formulación del problema

¿Cómo mediante el desarrollo de un videojuego se puede apoyar el aprendizaje de las asignaturas de algoritmia y programación, en los estudiantes de Ingeniería de sistemas pertenecientes a la Universidad de Cundinamarca Sede Chía?

2. OBJETIVOS

2.1 Objetivo general

Desarrollar un videojuego sobre conceptos básicos de programación y algoritmia para emplearlo como estrategia metodológica en el proceso de aprendizaje de los estudiantes de ingeniería de sistemas de la Universidad de Cundinamarca extensión Chía.

2.2 Objetivos específicos

- Establecer las funcionalidades bajo las cuales se desarrollará el videojuego. (El jugador pueda perder o ganar, se evidencie el puntaje a medida que el jugador responda correctamente, introducción a controles)
- Desarrollar la perspectiva del videojuego en primera persona.
- Codificar el videojuego para equipos de cómputo con sistema operativo Windows, basado en el diseño estructural previamente establecido para el aprendizaje de los conceptos básicos de la programación como variables, constantes, instrucciones, sentencias y la algoritmia en los estudiantes de ingeniería de sistemas de la universidad de Cundinamarca extensión chía.
- Evaluar el aplicativo con una población de pruebas (testers) compuesta por estudiantes que estén inscritos en las asignaturas de 'Lógica y Algoritmia' y Programación I, que verifiquen la funcionalidad y estabilidad del sistema, así como su utilidad para el aprendizaje de los temas de las asignaturas mencionadas.

3. ALCANCES Y LIMITACIONES

3.1 Alcances

El desarrollo y pruebas de la herramienta se practicó bajo la plataforma de Unity 3D y aplica para el sistema operativo Windows en sus versiones 7, 8, 8.1 y 10, el mismo será aplicado al programa de Ingeniería de Sistemas de la Universidad De Cundinamarca extensión Chía, como proyecto piloto en las asignaturas de lógica & algoritmia y Programación I con miras a mejorar su capacidad lógica para tener mayor eficiencia a la hora de resolver y analizar problemas en la asignatura de algoritmia, además de contribuir al mejoramiento del aprendizaje de programación, en base a conceptos como variables, constantes, instrucciones, ciclos, arreglos y sentencias, actuando como una herramienta de apoyo tanto a estudiantes como a docentes empleando esta herramienta como material de enseñanza complementario en su práctica académica.

3.2 Limitaciones

- Compatibilidad del sistema operativo con el aplicativo.
- Capacidad de almacenamiento del equipo.
- Controladores necesarios para el buen funcionamiento del aplicativo.
- Autorización previa por parte de los docentes de las asignaturas de Programación I y Lógica & algoritmia para la implementación del videojuego en el aula.
- Aprobación por parte de la dirección del programa de ingeniería de sistemas de la universidad de Cundinamarca extensión chía, para implementar el aplicativo como método de aprendizaje.

4. JUSTIFICACIÓN

Como se evidencio por medio de la encuesta mencionada anteriormente en el planteamiento del problema, aplicada tanto a los estudiantes de lógica y algoritmia como a los alumnos de programación I de la Universidad de Cundinamarca extensión chía, se encontró que el mayor problema al momento del aprendizaje de la lógica de programación radica en muchas circunstancias porque los estudiantes no adquirieron buenas bases, debido a aspectos como su falta de compromiso o por su desacuerdo frente a las metodologías ya estandarizadas dentro del núcleo académico; es por esto que se busca implementar un método más llamativo para los estudiantes, por el cual puedan adquirir un mejor nivel de conocimiento frente a la materia, que sea consistente y les permita mitigar los vacíos que puedan tener con respecto al tema, herramientas interactivas.

Para este proceso se recurrió al concepto de Gamificación que consiste en el uso de estrategias, modelos, dinámicas, mecánicas y elementos propios de los juegos en contextos ajenos a éstos con el propósito de transmitir un mensaje, unos contenidos o de cambiar un comportamiento a través de una experiencia lúdica que propicie la motivación junto con la implicación y la diversión (Francisco J. Gallego, 2014), al explorar el campo de los videojuegos se encuentra que tienen la capacidad de transmitir a los participantes mucha información de forma óptima, por ello se escogió este concepto de Gamificación que involucra el uso de la experiencia y métodos con los que cuentan estos y así contrarrestar la desmotivación por parte del estudiante siendo una opción eficaz para evitar el abandono temprano de la actividad educativa y apoyar el proceso de aprendizaje del estudiante.

Lo que se desea con esta práctica es motivar el aprendizaje de los estudiantes creando situaciones de estudio que sean divertidas, generando que la información quede fijada en el cerebro; proporcionando de esta manera una mejor comunicación, estimulando la participación de los estudiantes y lo que antes generaba poco interés por que lo veían aburrido sean actividades llamativas promoviendo la perseverancia al momento del aprendizaje. De esta manera sería una herramienta muy útil para los docentes siendo un sistema novedoso, fomentando el trabajo en las aulas para mejorar el desempeño de los alumnos, por ello se

debe emplear este modelo porque está pensado directamente en los problemas que presentan los estudiantes de este tipo de asignaturas impartidas en la universidad y que se ha incluido los factores que pueden retroalimentar y a su vez mejorar el contraste de estas, mientras otro tipo de videojuego ya existente en el mercado no cuenta con la suficiente apropiación del problema en particular ya que en su momento se diseñaron pensando en otro tipo de estrategia.

El uso de videojuegos como herramienta de aprendizaje busca enseñar y reforzar no solo conocimientos, sino también habilidades como lo pueden ser la resolución de problemas, la colaboración y hasta la misma comunicación, por medio de la motivación. Utilizando mecanismos que ayuden a incentivar a los estudiantes a participar en ellos.

5. LINEA DE INVESTIGACIÓN

Software, Sistemas Emergentes y Nuevas Tecnologías.

Conjunto de programas, subprogramas y menús que se elaboran a manera de aplicaciones y/o paquetes para cumplir un fin específico. (Universidad de Cundinamarca, 2009)

CAPITULO 2

6. MARCO TEÓRICO

6.1 Marco Referencial

Para el desarrollo del proyecto, se tomaron ciertas características de otros videojuegos como base, esto con el objetivo de disminuir el margen de error del proyecto, para que este tenga buenas bases de conocimientos y metodologías de juego, logrando que este sea de utilidad para el aprendizaje de algoritmia y la programación en estudiantes de las asignaturas de lógica & algoritmia y Programación I de la universidad de Cundinamarca sede Chía. Partiendo del hecho que el juego lo que hace es promover la interacción, el descubrimiento y el aprendizaje a través de prueba-error; ayudando a retener el conocimiento, la concentración y el desempeño a un alto nivel de ejecución.

Un estudio sobre Gamificación en educación superior, que analiza las consecuencias que han tenido estas experiencias en el proceso de aprendizaje de los estudiantes, concluye con una mejora sustancial de la captación de conocimientos por parte del alumno (Udima, 2017) este es el resultado expuesto por la doctora Sonia Pamplona Roche en un congreso internacional sobre aprendizaje, innovación y competitividad, analizando en la ponencia los efectos que tiene la Gamificación.

Un proyecto muy interesante, el cual es Videojuego 3D para el estudio de procesos cognitivos, realizado por Sergio Primo Galán (2014), el cual refleja en su trabajo que desarrolla un videojuego desde sus inicios para poder examinar y estudiar la cognición y el razonamiento de los participantes mientras jugaban el videojuego. El diseño implementado para el juego se realizó en función de los requisitos que aprobó la entidad académica en la cual desarrollo su carrera como profesional.

- **Codecombat**

Consiste en un juego de rol online, desarrollado en HTML5 y CoffeeScript donde se puede aprender los fundamentos de la programación en diferentes lenguajes, mientras se va jugando y escribiendo código desde el navegador; el juego cuenta con un ambiente de diferentes pistas musicales, además de la opción de asignar complementos al protagonista, como armas, armaduras y mascotas, al igual que cuenta con diversas herramientas que generan una mejor experiencia. Siendo una plataforma para que los estudiantes aprendan sobre la ciencia de la computación mientras hacen uso de un juego real. (CodeCombat, 2018)

Desde 2013, CodeCombat ha crecido de un mero juego de bocetos a un juego vivo y próspero, Más de 5.000.000 Total de jugadores Han comenzado su viaje de programación a través de CodeCombat se puede aplicar al proyecto la idea de movilizar al avatar y que este ejecute acciones en el videojuego, por medio del aprendizaje y la aplicación de comandos simples, que faciliten el aprendizaje de estos y su implementación por parte de los estudiantes (CodeCombat, 2013).

- **Modelo metodológico para el desarrollo de videojuegos 3D Softserebii**

Es un proyecto desarrollado por Juan Carlos Díaz Ulloa y Jhonny Castillo Gordo, - estudiantes de Ingeniería de Sistemas de la Universidad de Cundinamarca, publicado en el año 2012 donde muestran el modelo con las actividades y procedimientos que se deben realizar al momento de la preparación, realización y finalización de la estructura para la implementación de un videojuego en 3D. Partiendo de este proyecto se implementa unas ideas en la creación de un diseño que sea estándar al momento de realizar un videojuego en forma organizada, eficiente y eficaz, permitiendo desarrollar y enfocar las ideas de manera adecuada sin desviarse del objetivo inicial. (Castillo & Diaz, 2012) Esto se realizó después de una debida investigación que arrojó las deficiencias en cuanto a la capacitación y la documentación para modelos de animación y videojuegos en 3D.

- **Videojuego Para La Enseñanza Del Cuerpo Humano Mediante Guante Interactivo.**

Es un proyecto desarrollado por Víctor Manuel Pinzón Rodríguez estudiante de ingeniería de sistemas de la universidad de Cundinamarca extensión chía en el segundo periodo del año 2018, se trata de un videojuego sobre el cuerpo humano dirigido a un público infantil que consiste en 6 niveles iniciaba por decir en el sistema digestivo donde se tiene que completar las partes que lo integran por medio de un guante interactivo si lo hace de forma correcta la parte que se va colocando queda fija y avanza al siguiente nivel que corresponde al sistema respiratorio sino se devuelve de donde se tomó, la dinámica del juego consiste en ubicar de forma correcta las partes del cuerpo por medio del guante, se desarrolló en Unity y las figuras fueron hechas en Blender.

- **Hack n slash**

Inspirado por la serie Zelda, Hack n Slash enseña la idea de aprender la programación reprograming las propiedades de un objeto en el juego. (SLASH, 2014).

De Hack n Slash se ha tomado la idea de permitir al jugador, reprogramar comandos conforme vea conveniente a la situación y de acuerdo con los conocimientos de algoritmia y programación que vaya adquiriendo, para que su desarrollo en el juego sea cada vez mejor y más fácil, además de incentivar al estudiante a analizar las situaciones que se le presenten y poder tomar la mejor decisión para poder seguir avanzando. Lo cual va a generar en los estudiantes un pensamiento claro y preciso para la resolución de problemas de manera óptima y sistemática.

- **Production and deployment of educational videogames as assessable learning objects**

En este artículo los autores proponen que los videojuegos son educativos y que pueden generar un aprendizaje evaluable. La generalización de los Objetos de Aprendizaje basados en juegos como material de aprendizaje serio requiere su integración en la infraestructura de e-learning preexistente (sistemas y cursos) y la inclusión de procedimientos de evaluación

que tengan en cuenta el juego. (Centro de Estudios Superiores Felipe II, 2006). De lo cual se puede tomar para el proyecto, que por cada logro u objetivo que logre un estudiante obtendrá puntos, lo cual el profesor podrá tomar como referencia para poder evaluar al estudiante con respecto a los temas que se desean evaluar.

- **Codingame**

CodinGame es un juego que propone al usuario distintos ejercicios y retos de programación y ofrece una respuesta visual cuando son resueltos. (Carballo, 2015), el cual parte de la premisa “Si mezclas los juegos con el aprendizaje se obtiene una experiencia muy motivadora” (Desmoulins, 2015).

Se hace mucha referencia a incentivar al jugador con premios o logros cada vez que este logre superar un reto o alcanzar un objetivo, esto se hace con el fin de que el estudiante se vea motivado a seguir avanzando y progresando más en el juego y por ende en su aprendizaje, para alcanzar más metas.

6.2 Marco Conceptual

Con el fin de hacer más fácil el proceso de realización y comprensión del presente proyecto es indispensable relacionar un glosario de los términos claves que hacen parte fundamental para el desarrollo eficaz del mismo.

Análisis: en él se realiza un examen detallado de un objeto o situación para identificar cuáles son sus características, cualidades o su estado, y de esta manera sacar conclusiones, se puede hacer llevando por separado las partes que la constituyen (RAE.ES, 2017).

Aprendizaje: proceso en el cual se modifican y adquieren habilidades, destrezas, conocimientos, conductas o valores como resultado de un estudio, experiencia, instrucción, razonamiento u observación de un conjunto de cosas o circunstancias (Johnson, 2001).

Diseño: es la actividad creativa o de ingenio para dar soluciones, innovar o crear nuevas cosas para contribuir a la sociedad trabajando en intervenciones concretas, garantizando que las tecnologías se emplean pensando en las personas. (Encyclopædia Britannica, 2017).

Enterprise Architect: en esta herramienta de diseño, se cubre el desarrollo del software desde la elaboración de diagramas de requerimientos, modelos de diseño y demás que comprenden las etapas de elaboración del proyecto. (Sparx, 2018).

Estrategias: se elabora un plan que especifica una serie de pasos o de conceptos que tienen como finalidad sacar adelante un determinado objetivo, teniendo una planificación estratégica o táctica, que implica el control de múltiples unidades. (Encyclopædia Britannica, 2017).

Gamificación: Es un tipo de aprendizaje que transpone la mecánica de los juegos al ámbito educativo-profesional con el fin de conseguir mejores resultados, la gamificación es usada para para absorber, como, por ejemplo, algunos conocimientos, mejorar habilidades, o bien recompensar acciones concretas, entre otros muchos objetivos (IIEMD, 2017).

Ingeniería: estudio y aplicación, por especialistas, de las diversas ramas de la tecnología. (Johnson, 2001).

MySQL: es el sistema de gestión de bases de datos relacional desarrollado bajo licencia dual, es decir pública. Siendo bastante común su aplicación en diferentes plataformas como Windows o Linux, en aplicaciones web ligadas a PHP. Entre otras cosas ofrece un sistema de contraseñas y privilegios, soporta una gran cantidad de datos, permite escoger múltiples motores de almacenamiento por cada tabla que se maneje, además de otro tipo de características que hacen parte de este administrador de bases de datos. Contando con un conjunto completo de funciones avanzadas, herramientas de gestión y soporte técnico para alcanzar los niveles más altos de escalabilidad, seguridad, confiabilidad y tiempo de actividad (Oracle, 2018).

Nuevas Tecnologías: hace referencia a los últimos desarrollos tecnológicos y sus aplicaciones, centrándose en procesos de comunicación en las áreas de informática, video y telecomunicaciones. Avanzando en los diseños de sistemas operativos y programación, mejorando procesos de grabación, velocidad, circuitos, en los sistemas de cableado permitiendo el aumento de cantidad de información, además de desarrollos educativos. (Fernandez, 2016)

Plataforma: sistema base para hacer funcionar determinados módulos de hardware o de software con los que es compatible, definido por estándares donde se determinan sus arquitecturas. (RAE.ES, 2017)

Programación: proceso donde se diseña, codifica, depura y mantiene el código fuente de programas de computadora, desarrollando aplicaciones de todo tipo regido por reglas, instrucciones o comandos, mediante una secuencia de datos. Automatizando tareas, cumpliendo requisitos del programa, diseñando una arquitectura, descomponiendo sus partes, centrada en pasos de planificación y diseño de programas. (Johnson, 2001)

Sistemas Emergentes: son un tipo de organismo que sostiene una gran capacidad para generar conductas o procesos innovadores, pudiéndose adaptar a los cambios bruscos de mejor forma que los modelos jerárquicos o más rígidos. (Johnson, 2001)

Software: es un conjunto de programas, instrucciones y reglas informáticas que permiten ejecutar distintas tareas en una computadora. (RAE.ES, 2017)

Técnica: Procedimiento o conjunto de estos centrados en la forma de hacer. Se refiere al trabajo con herramientas, materiales e instrumentos. La técnica requiere tanto destrezas manuales como intelectuales, frecuentemente integra el uso de herramientas y saberes muy variados. (Garcia, 2005)

Unity: es un motor de videojuego multiplataforma, con soporte de compilación con diferentes tipos de plataformas. Puede usarse en conjunto con otros entornos gráficos como

lo son Blender, Cinema 4, Adobe Photoshop entre otros. Cuenta con un soporte para mapeado de relieve, de reflejos, por paralaje, de sombras, etc. (Technologies, 2018)

Videojuego: Un videojuego o juego de vídeo es un software creado para el entretenimiento en general y basado en la interacción entre una o varias personas, mediante un aparato electrónico que ejecuta dicho videojuego; este dispositivo electrónico puede ser una computadora, una máquina arcade, una videoconsola, un dispositivo handheld (un teléfono móvil, por ejemplo) los cuales son conocidos como "plataformas". (Wikitendo, 2011)

Xampp: es el entorno más popular de desarrollo con PHP, de software libre, para la gestión de base de datos. (Apache, 2018)

6.3 Marco Ingenieril

Para dar inicio al desarrollo del aplicativo, se pensó primero en hallar que dificultades se presentaban en los estudiantes al momento de aplicar la lógica en la programación, para ello se realizó una encuesta a los estudiantes de lógica de programación, y programación I, bajo la plataforma **e-encuesta.com** que es un sistema para la creación de encuestas online por medio de plantillas editables o desde cero, utilizando cierta gama de interfaces que permiten personalizar a nuestro gusto el cuestionario que deseamos desarrollar con diferentes tipos de preguntas además de contar con una fácil distribución ya que se pueden enviar por email, publicándola en un blog o simplemente compartiéndola en una red social. También arroja resultados en tiempo real por medio de informes automáticos de frecuencias.

ro | <https://manager.e-encuesta.com/v2/survey/1296936/preview>

Dificultades En La Lógica De Programación

1.

Para programar se requiere de una combinación de actitudes, conocimiento y un ambiente adecuado, pero no siempre a todos les agrada la idea de irse por esta rama. De acuerdo a esto buscamos identificar por medio de esta breve encuesta cual es el posible inconveniente que se presenta en los estudiantes al momento de pasar por el proceso de aprendizaje en las clases de Algoritmia, Programación I y Programación II, para poder implementar una herramienta que mitigue las dificultades. Esta encuesta es realizada por Andrés Cano y Vanessa Sanchez estudiantes de Ingeniería de Sistemas de Décimo semestre, agradecemos su colaboración.

1 Que tan difícil le parece la lógica de programación en una escala de 1 a 5; siendo 1 el nivel más bajo y 5 el nivel más alto? (*)

1

2

3

4

5

2 cuál cree usted qué es el principal motivo por el cual los estudiantes presentan dificultades en la parte de la lógica de programación? (*)

Falta de interés

No le gusta

Ilustración 1. Aplicación encuesta a los estudiantes de Lógica de programación y Programación I de la universidad de Cundinamarca extensión chía.

Al momento de crear la encuesta se escogió una interfaz de fácil manejo y llamativa para el público, se plantearon 3 preguntas cerradas con respecto al grado de dificultad en la lógica de programación sus posibles generadores y si esto conlleva a la deserción de la carrera, una vez creada se compartió en diferentes redes sociales con el grupo de personas que eran de nuestro interés para la investigación y por ultimo bajo la misma plataforma se hizo el análisis de los resultados.

Después de obtener unos resultados concretos se establecieron los objetivos del proyecto y se definió la estrategia o modelo que se iba a seguir para el desarrollo del proyecto. Consultando fuentes que sirvieran como guía para llevar a cabo este proceso. Con el propósito de desarrollar este videojuego como una alternativa más didáctica y de mayor interés para los estudiantes, se aplicó el concepto y los recursos de la gamificación, para que este juego pueda cubrir los conceptos faltantes y que este permita adquirir más conocimientos de una manera dinámica y divertida, algo que permita a los estudiantes incentivarse a aprender y retener más la información por medio de buenas experiencias al adquirir el conocimiento y recibir recompensas. Basados en esto el videojuego se desarrolló en 3D utilizando un motor de videojuego multiplataforma llamado Unity, el cual fue creado por Unity Technologies. El cual está disponible como plataforma de desarrollo para Microsoft

Windows, OS X, Linux, que en este caso fue desarrollado sobre el sistema operativo Windows 10. Los lenguajes que maneja Unity son; C#, JavaScript y boo. Un aspecto a tener en cuenta con respecto a estos lenguajes, es que Boo ha sido discontinuado a efectos prácticos en las nuevas versiones de Unity, ya que no pueden crearse scripts en este lenguaje directamente en el editor. (Android, 2015), En este caso, el desarrollo del aplicativo fue con el lenguaje C#, ya que es el lenguaje que prima el propio motor de juegos Unity y facilita el uso de scripts, creados por terceros en este desarrollo lo que genera un soporte especial.

Se menciona también que los videojuegos mejoran la toma de decisiones, Investigaciones anteriores mostraron que las personas que juegan videojuegos de acción tienen tiempos de reacción más rápidos que los que no juegan (Hsu, 2010). Los científicos hallaron que los jugadores son mejores en la toma de decisiones rápidas y precisas, basándose en la información que extraen de su entorno. Los investigadores usaron una serie de experimentos de decisión para compararlos (Granic, Lobel, & Engels, 2014). Los jugadores jugaban 5 horas semanales a juegos como el Grand Theft Auto: San Andreas o Halo 2, lo que permitió afirmar que la experiencia con la práctica de videojuegos como estrategia metodológica para el aprendizaje aplicado a los estudiantes de ingeniería de sistemas de la universidad de Cundinamarca extensión chía, les permite desarrollar habilidades para toma de decisiones de manera clara y precisa, lo cual es vital para un futuro ingeniero, al verse en situaciones de toma de decisiones para desarrollo y diseño de un proyecto.

Por medio del videojuego los estudiantes además de desarrollar habilidades, refuerzan los conocimientos adquiridos permitiéndoles un mejor desempeño en la carrera.

Unity permite en la creación del juego crear escenas para las diferentes secciones que va contener el juego, como lo son menús, niveles, obstáculos y lo esencial en el diseño.

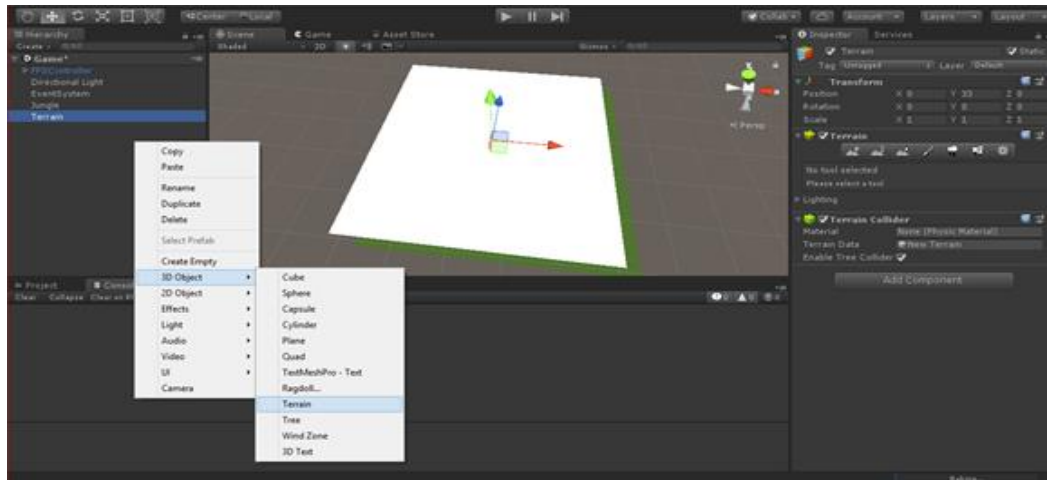


Ilustración 2. Creación del terreno del juego en Unity.

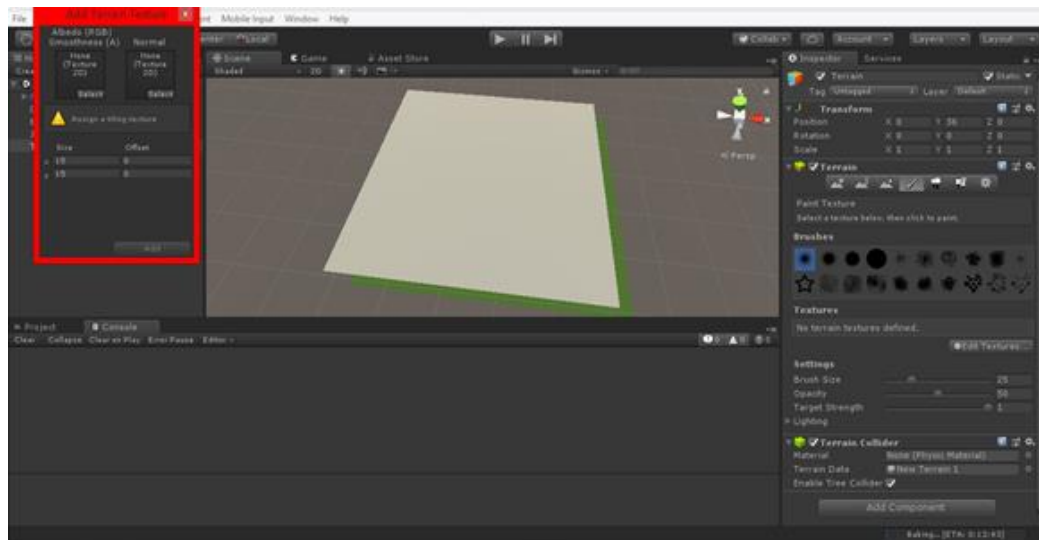


Ilustración 3. Selección de la textura que se aplicara al terreno del juego.

Bajo esta plataforma se llevó a cabo todo el diseño del videojuego como el terreno, los personajes, el ambiente en el cual se desenvolvería la historia del juego por su practicidad y diversidad en cuanto a las herramientas que ofrece para la elaboración de este tipo de proyectos, además de permitir el enlace e interacción con otras plataformas como lo es el caso de MySQL donde se creó la base de datos, se procedió a instalar Asset Server, el cual crea un admin y su respectiva password. Esta contraseña es requerida para administrar el Asset Server dentro de Unity. Para lo cual debe conectarse al Asset Server como el administrador antes de crear el proyecto y los usuarios en Unity.

Cada servidor puede contener varios Proyectos, y cada Usuario puede tener permiso a uno o más proyectos. Los Proyectos por lo general son ortogonales, y únicos en la colección de

Asset. Es mejor pensar en “un proyecto es igual a un juego”. Se debe crear la base de datos en MySQL, utilizando un servidor local, para poder trabajar de manera off-line, que en este caso va a ser Xampp. Para este caso desde el computador se genera un servidor, para que en este se pueda montar la base de datos, y así acceder a ella por medio de Unity. Creando la base de datos, con los campos requeridos para poder almacenar la información requerida de cada jugador.

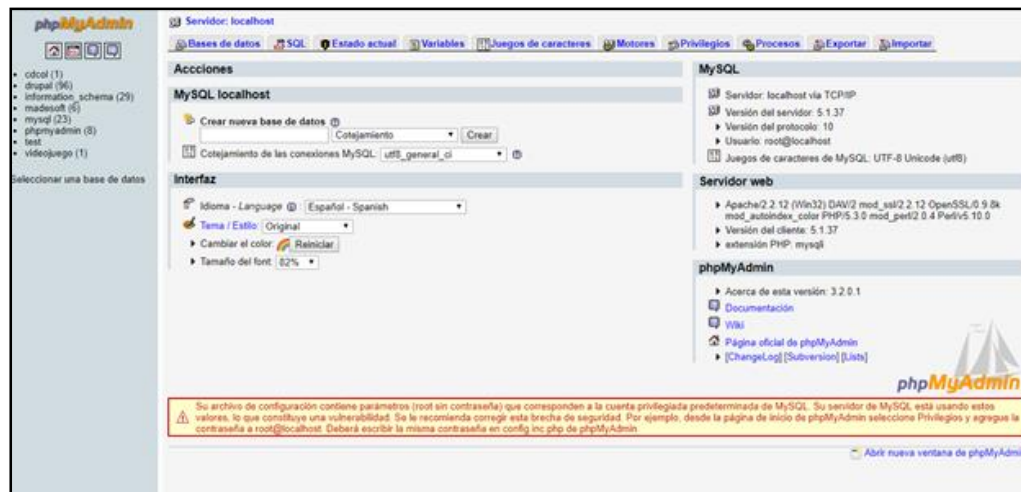


Ilustración 4. Acceso a la base de datos del juego.

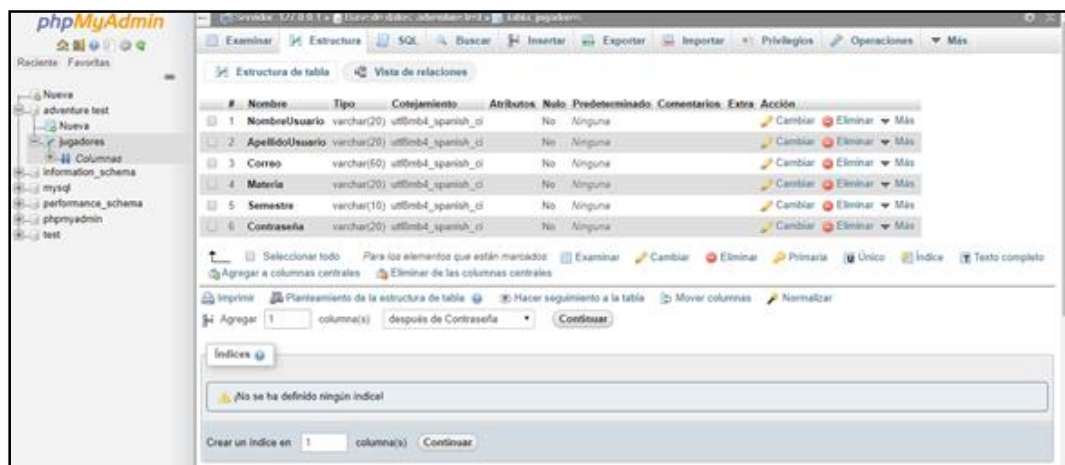


Ilustración 5. Base de datos- Videojuego llevando incorporados los campos con la información necesaria para evidenciar el desarrollo del juego.

Para el diseño de los diferentes diagramas se utilizó Enterprise Architect herramienta grafica multiusuario diseñada para la ayuda de creación de sistemas robustos, usando reportes y documentación de alta calidad. Proporciona una trazabilidad completa desde requisitos,

análisis y modelos de diseño hasta la implementación y el despliegue. La verificación efectiva, la validación y el análisis de impacto inmediato son posibles a lo largo de todo el ciclo de vida, utilizando capacidades tales como la Matriz de relaciones de Enterprise Architect y la Vista de jerarquía. (McMillan, 2017)

Cuenta con unas características de administración de requisitos que se pueden usar para:

- Definir un modelo organizado de requisitos jerárquicos
- Rastrear la implementación de los requisitos del sistema para modelar elementos
- Busque e informe sobre los requisitos
- Realizar un análisis de impacto de los cambios propuestos a los requisitos

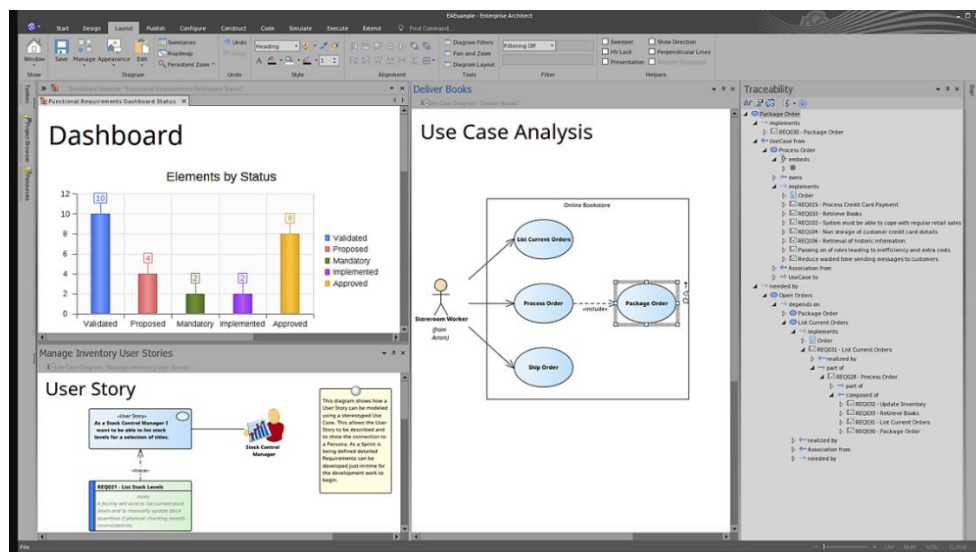


Ilustración 6. Modelo jerárquico de requisitos en Enterprise Architect (McMillan, 2017)

Enterprise también permite personalizar el código fuente generado de acuerdo a las especificaciones del proyecto.

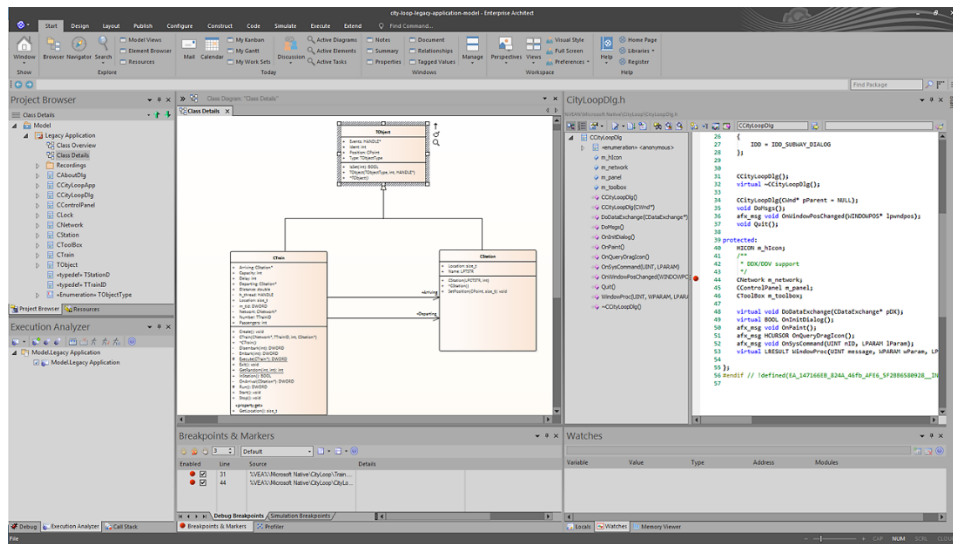


Ilustración 7. Modelos de entidad relación en Enterprise Architect (McMillan, 2017)

En Enterprise Architect en la parte de modelado de bases de datos, modela las tablas de la base lo que son columnas, claves, claves foráneas, para la parte de las relaciones usa UML y perfiles de modelado de datos incorporado. También ofrece la vista jerárquica de las interconexiones entre elementos en la administración de los requisitos, permitiendo en esta parte agregar o anidar estas relaciones. Y cuenta con diversos formatos de imágenes para los diagramas.

CAPITULO 3

7. METODOLÓGIA

La Metodología empleada en el proyecto fue SCRUM, que está diseñada para la gestión y desarrollo de software basada en procesos iterativos e incrementales, se maneja un esquema basado en reuniones en distintos órdenes, tiempos y bosquejos para llevar un control de los avances del proyecto.

SCRUM adopta una aproximación pragmática, aceptando que el problema no puede ser completamente entendido o definido, y centrándose en maximizar la capacidad del equipo de entregar rápidamente y responder a requisitos emergentes ya que se pueden generar cambios no contemplados inicialmente. El proceso que maneja SCRUM puede ser visto desde una pila de productos, en un retraso que puede presentarse, el tiempo que se llevara realizar el proyecto y finalmente el incremento de trabajo del software en pocas palabras un modelo de referencia, el cual se encarga de definir un conjunto de prácticas y roles, que pueden tomarse como un punto de partida para definir el proceso que se va a desarrollar durante el proyecto.

Fases:

- 1. Planificación:** como primer paso se definió las funcionalidades del juego, el público al cual está dirigido, el tiempo que tardaría todo el proceso.
- 2. Desarrollo:** una vez definido el sprint que se iba a llevar a cabo se revisaba los riesgos que se podrían presentar para no generar retrasos.
- 3. Revisión:** cuando se finalizaba una tarea se hacía una retroalimentación para mirar que aspectos debían mejorarse o si requería alguna modificación importante.
- 4. Retroalimentación:** se llevó a cabo unas reuniones con el director del proyecto para mostrarle los resultados que se iban obteniendo y así mismo hacer las correcciones pertinentes que fueran requeridas.
 - Se definen los requerimientos funcionales y no funcionales del proyecto.
 - Se define como va a estar diseñado el juego:

Público objetivo.

El público al cual está dirigido el juego son los estudiantes de lógica y algoritmia, y programación I del programa Ingeniería de Sistemas de la Universidad de Cundinamarca extensión chía, los docentes que imparten estas materias.

Características.

El juego consta de un menú de registro y otro de autenticación de usuario, la dinámica del juego trata de vencer a un ejército enemigo sobrepasando unos obstáculos y a medida que los va derrotando activa un toquen con una pregunta que al ir respondiendo de forma correcta va generando un puntaje, el jugador puede perder o ganar. Existe una introducción a los controles para que el jugador tenga unas instrucciones claras sobre los movimientos que puede realizar dentro del juego.

Personajes.

El personaje principal es en perspectiva de primera persona el diseño del personaje se limitó a crear una mano y una espada con la que peleara contra los personajes secundarios, los personajes secundarios son un ejército de orcos al cual se tiene que enfrentar para avanzar en el juego.

Herramientas de Desarrollo.

Para el desarrollo del videojuego se utilizaron las siguientes herramientas:

Software

Unity 3D: Es un motor diseñado para el modelado de videojuegos bajo licencia o para uso gratuito, que permite crear juegos multiplataforma, además de permitir la integración con diferentes lenguajes para el manejo de eventos y acciones, en este caso se implementaron scripts con C#.

MySQL: Es uno de los motores de bases de datos más robustos y utilizado a nivel mundial, bajo este se desarrolló la base de datos que el juego utilizará para almacenar los datos del jugador y los contenidos del juego.

Hardware

El desarrollo del videojuego se realizó en 2 Portátiles con las siguientes características: Procesador Core i5 (2.6GHz), Memoria RAM de 6GB, Tarjeta de Video incorporada (Intel(R) HD Graphics).

- Se establece un cronograma bajo el cual se desarrolla el proyecto.
- Diseño de módulos que van a contener dentro del videojuego.

Se contextualizo que tipo de escenario y objetos iban dentro del juego tomando a consideración las limitaciones técnicas del hardware sobre el cual correrá éste, los sonidos que se reproducirían; para ello se hizo uso de elementos sonoros tales como, sonidos ambientales, efectos sonoros y música. Se recreó como los entes virtuales interactúan dentro del juego, definiendo las reglas de éste. Se escogió el lenguaje de programación bajo el cual se describió su funcionamiento estático y dinámico.



Ilustración 8. Se realizó la creación de los diferentes módulos que llevaría el juego.

```

1 using System;
2 using UnityEngine;
3 using UnityEngine.InputSystem;
4
5 namespace UnityStandaAssets.Characters.ThirdPerson
6 {
7     [RequireComponent(typeof(ThirdPersonCharacter))]
8     public class ThirdPersonCharacterControl : MonoBehaviour
9     {
10         private ThirdPersonCharacter m_Character; // A reference to the ThirdPersonCharacter on the object
11         private Transform m_Camera; // A reference to the main camera in the scene's transform
12         private Vector3 m_CameraForward; // The current forward direction of the camera
13         private Vector3 m_Move;
14         private bool m_Dump; // the world-relative desired move direction, calculated from the joystick and user in
15
16         private void Start()
17         {
18             // get the transform of the main camera:
19             if (Camera.main != null)
20             {
21                 m_Camera = Camera.main.transform;
22             }
23             else
24             {
25                 Debug.LogWarning(
26                     "Warning: no main camera found. Third person character needs a Camera tagged 'MainCamera', for camera-relative control.
27                     // we use self-relative controls in this case, which probably isn't what the user wants, but hey, we warned them!"
28                 );
29             }
30
31             // get the third person character (this should never be null due to require component)
32             m_Character = GetComponent<ThirdPersonCharacter>();
33         }
34     }

```

Ilustración 9. Script para el control de movimiento del personaje en el juego.

- Se realizan las pruebas correspondientes para verificar si antes de implementarlo con los estudiantes requiere ajustes para su puesta en marcha.

8. DESARROLLO DEL PROYECTO

8.1 Desarrollo de la metodología

Como se mencionó previamente en la metodología, se estableció qué clase de videojuego se quería desarrollar, en este caso el enfoque es a un videojuego de tipo aventura con un personaje en primera persona enfrentando un enemigo, al iniciar el juego se evidencia cómo funcionan los controles del juego con respecto a las actividades que realizan los enemigos se les dio una inteligencia artificial para que ejecutaran ciertos roles en el transcurso del juego. Cuando el jugador mata al enemigo aparece el toquen donde está la pregunta para que pueda seguir avanzando.

Como segundo paso se realizó una encuesta para analizar cuáles eran las falencias que se presentaban al momento de adquirir el conocimiento en las materias de lógica de programación y programación I donde se pudo llegar a la conclusión de que los conceptos de la lógica para programar y la programación son denotados como temas con un nivel de dificultad intermedio para los estudiantes de la Universidad de Cundinamarca extensión Chía. También gracias a la encuesta realizada a los estudiantes se pudo encontrar que la dificultad para el aprendizaje de dichos conocimientos está relacionada con el proceso para adquirir dicha información, tales como falta de concentración o interés por parte del estudiante, uso de metodologías inadecuadas o que resulten complicadas y tediosas para los estudiantes, gracias a esto se pudo delimitar los conceptos necesarios para ser implementados en el videojuego para que estos puedan ser adquiridos de una manera más dinámica y generar en los estudiantes interés hacia las materias. Por último, se desarrollaron los diagramas para tener claro cómo iba a ser el desarrollo del videojuego, para minimizar errores en cuanto a navegación y funcionalidad del mismo además de la creación del entorno del juego que es lo que se presenta a continuación.

Modelo de Requisitos.

- Diagrama de Flujo

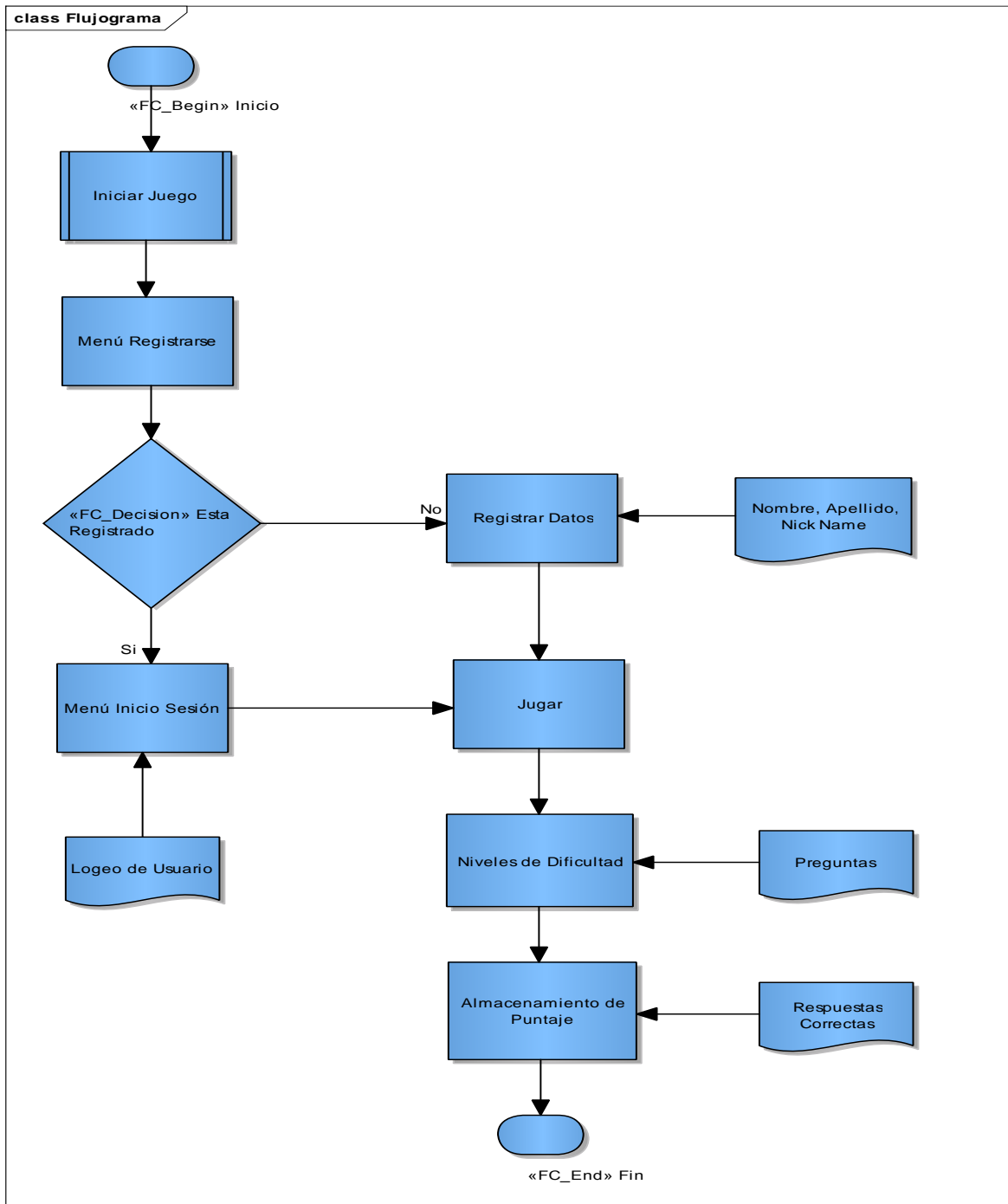


Ilustración 10. Diagrama de flujo del sistema que contiene el juego.

- **Modelo de Casos de Uso**

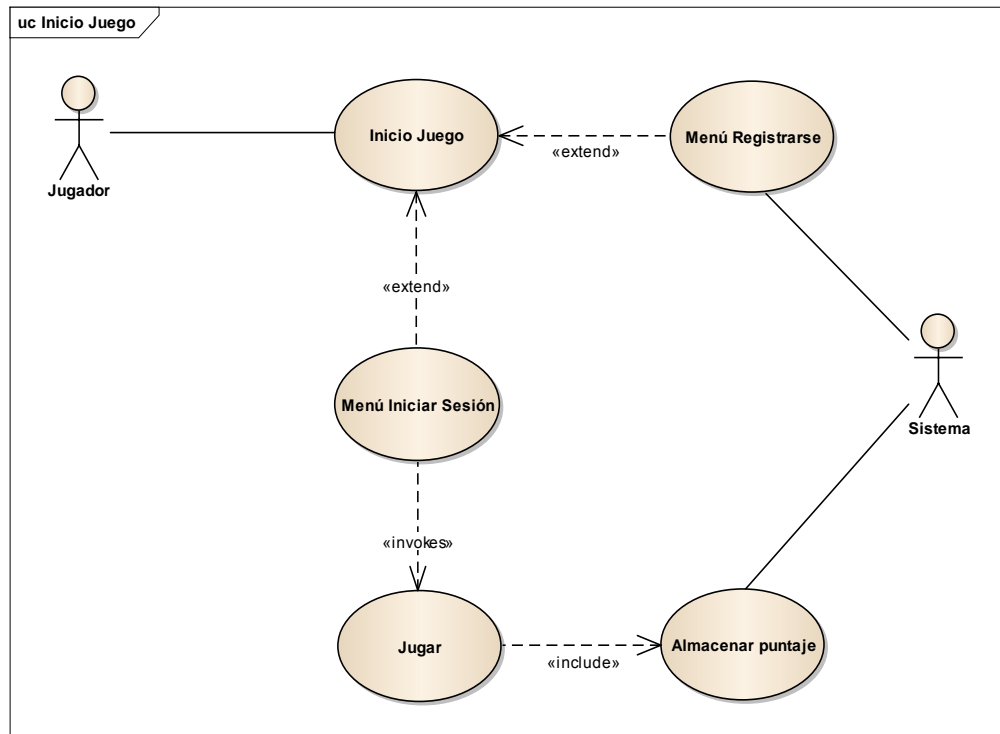


Ilustración 11. CU01 Inicio de Juego.

Tabla 1. CU01 Inicio del Juego.

Caso de Uso: Inicio Juego	
Actor: Jugador	
Descripción de los procesos	Alternativas
El jugador inicia el juego	
Comentarios	
El videojuego debe estar instalado, cuando el jugador inicia el juego aparece la pantalla inicial	

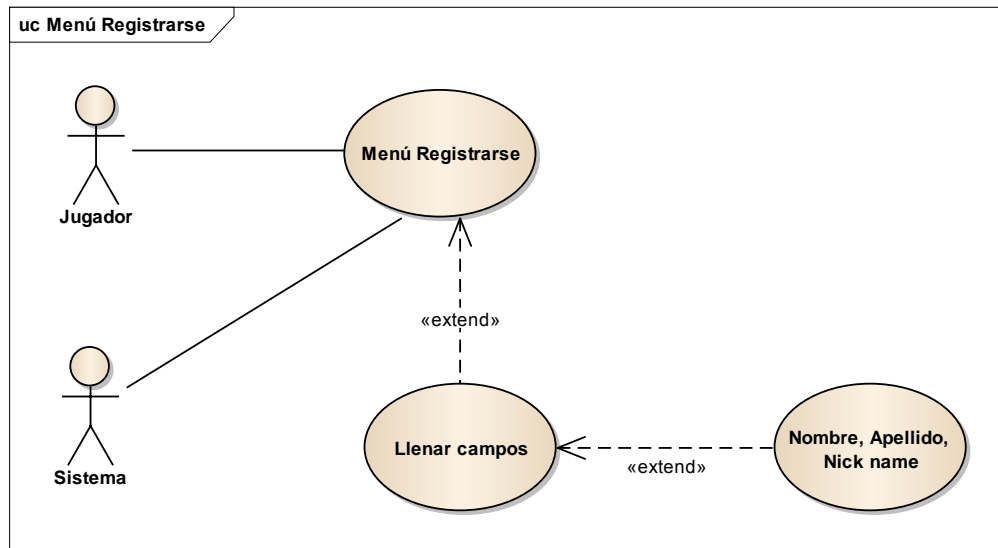


Ilustración 12. CU02 Menú Registrarse.

Tabla 2. CU02 Menú Registrarse.

Caso de Uso: Menú Registrarse	
Actor: Jugador	
Descripción de los procesos	Alternativas
El jugador se encuentra en el menú de registro donde debe llenar unos campos	Si ya está registrado ingresa directamente al menú de inicio de sesión
Comentarios	
El jugador debe estar en el menú de registro para ingresar sus datos para posteriormente jugar	

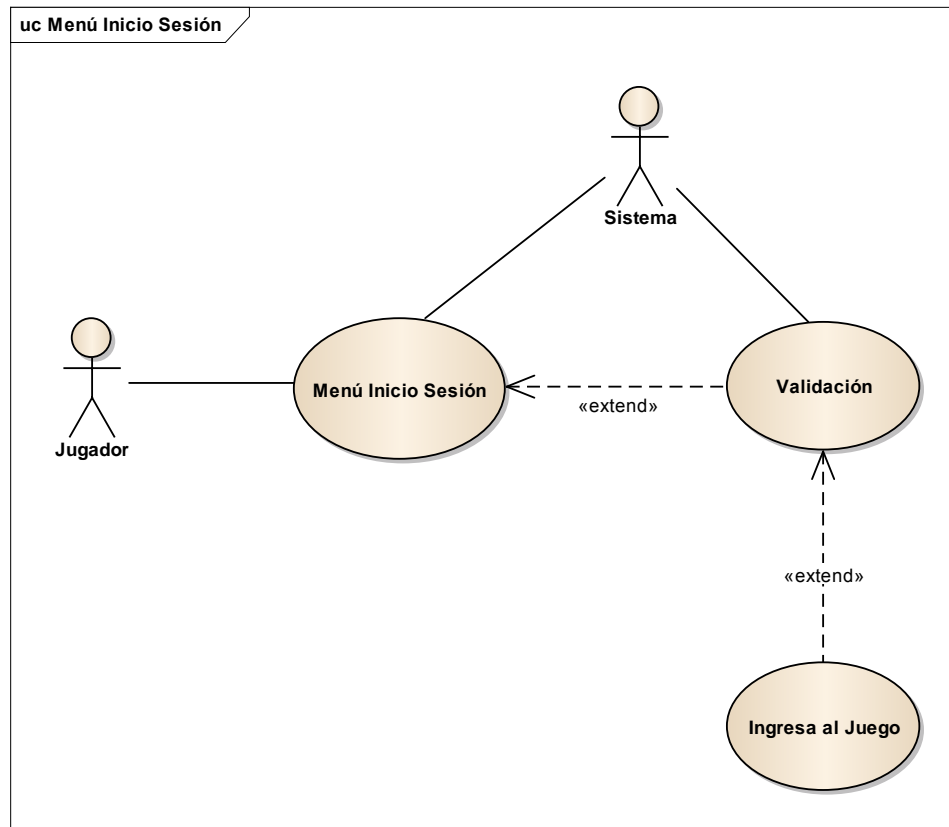


Ilustración 13. CU03 Menú de Inicio de Sesión.

Tabla 3. CU03 Menú de Inicio de Sesión.

Caso de Uso: Menú Iniciar Sesión	
Actor: Jugador	
Descripción de los procesos	Alternativas
El jugador hace el proceso de autenticación e inicia el juego	
Comentarios	
El jugador debe estar en el menú de inicio de sesión para acceder al juego	

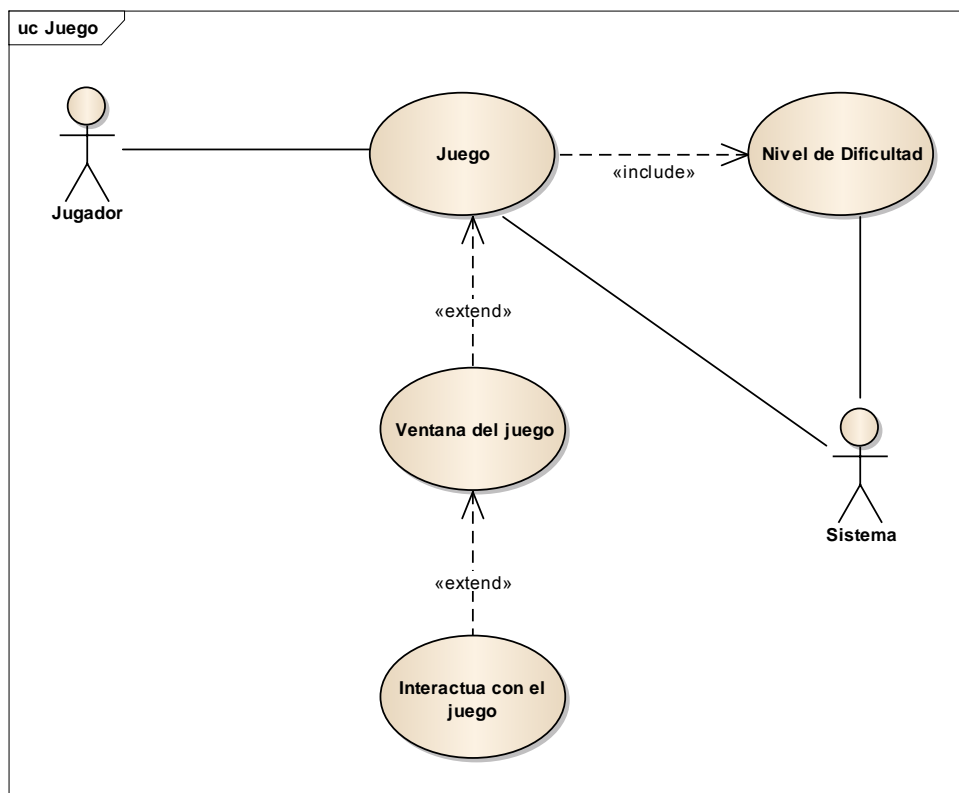


Ilustración 14. CU04 Juego.

Tabla 4. CU04 Del Juego.

Caso de Uso: Jugar	
Actor: Jugador	
Descripción de los procesos	Alternativas
El jugador comienza el juego	
Comentarios	
El jugador comienza a jugar pulsando las teclas para que la cámara se mueva en el entorno	

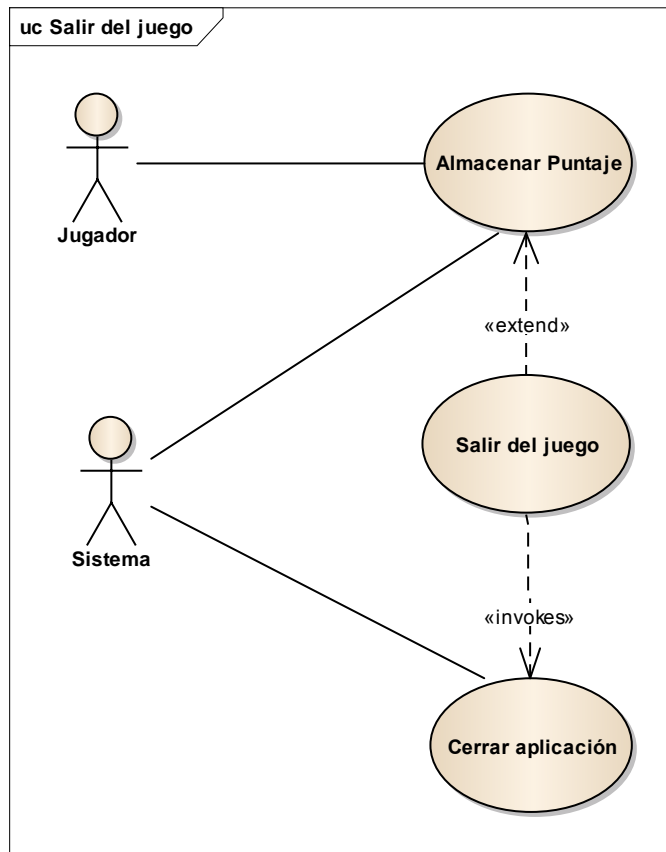


Ilustración 15. CU05 Salir del Juego.

Tabla 5. CU05 Salir del Juego.

Caso de Uso: Almacenar Puntaje	
Actor: Jugador	
Descripción de los procesos	Alternativas
Por respuesta correcta se asigna un puntaje	
Comentarios	
A medida que el jugador va respondiendo de forma correcta aparece un puntaje	

- **Requisitos Funcionales**

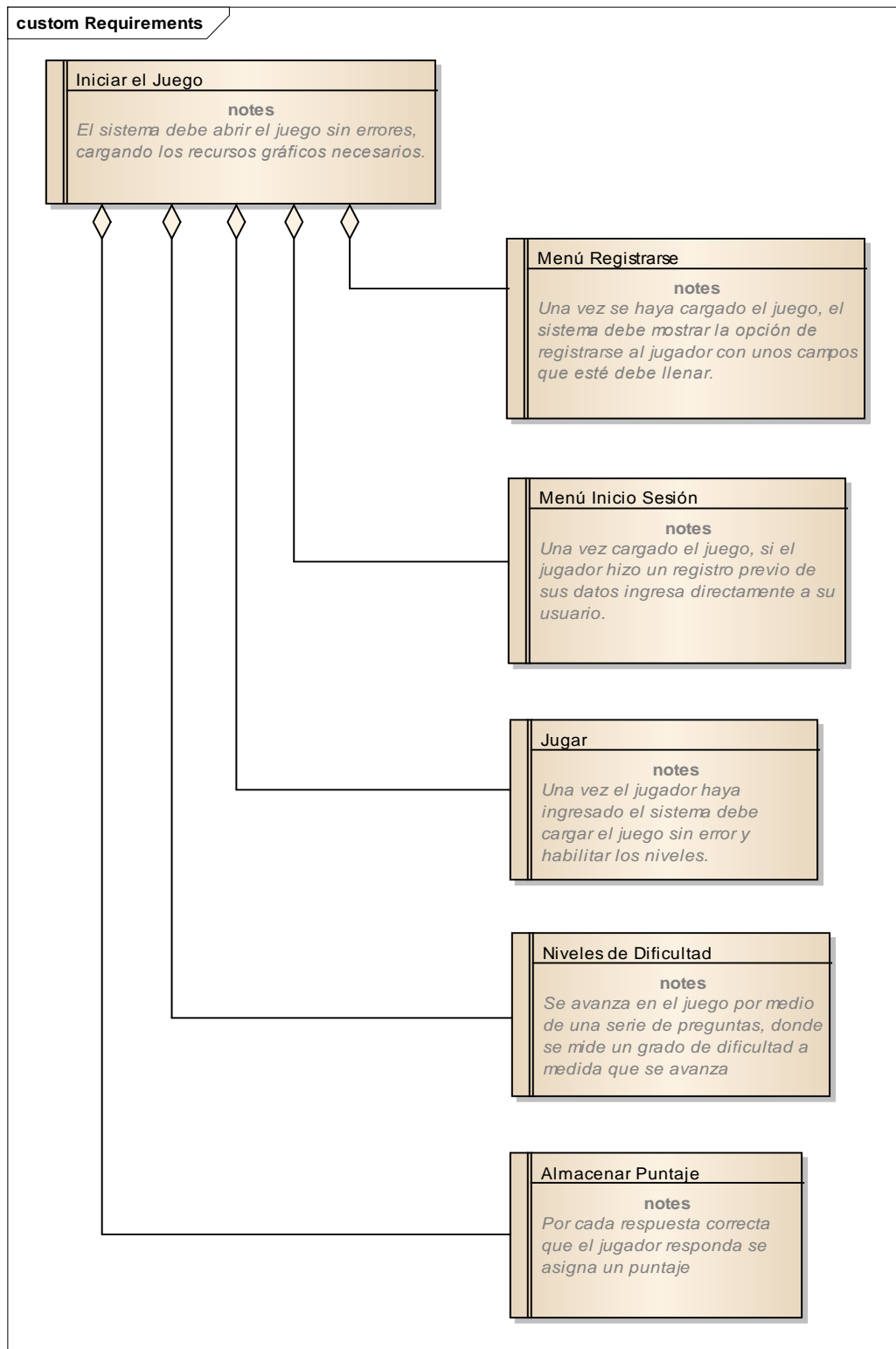


Ilustración 16. Diagrama de los requisitos funcionales del sistema que rige el juego.

- **Requisitos No Funcionales**

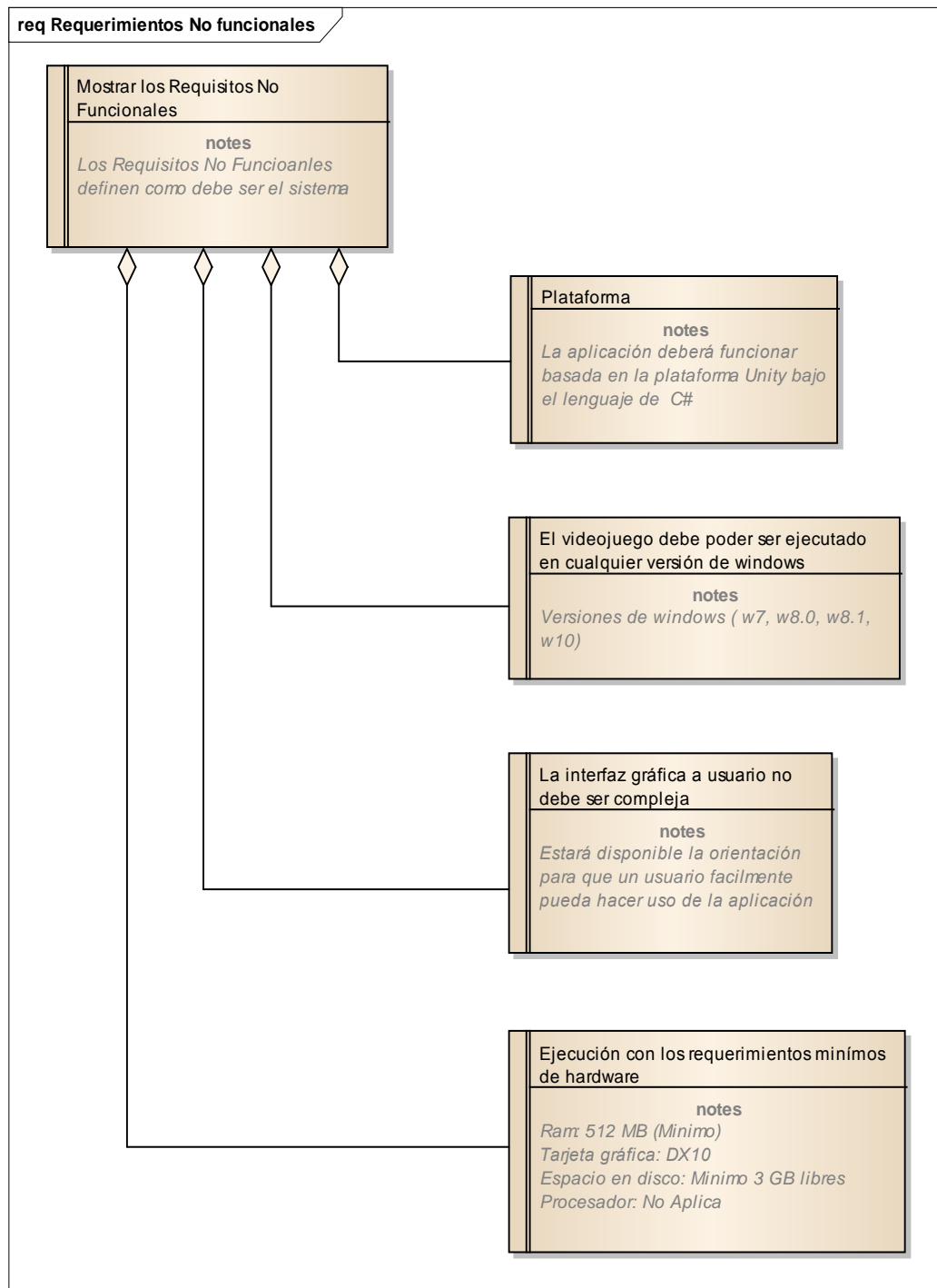


Ilustración 17. Diagrama de los requisitos No funcionales del sistema por los que se rige el juego.

- **Restricciones**

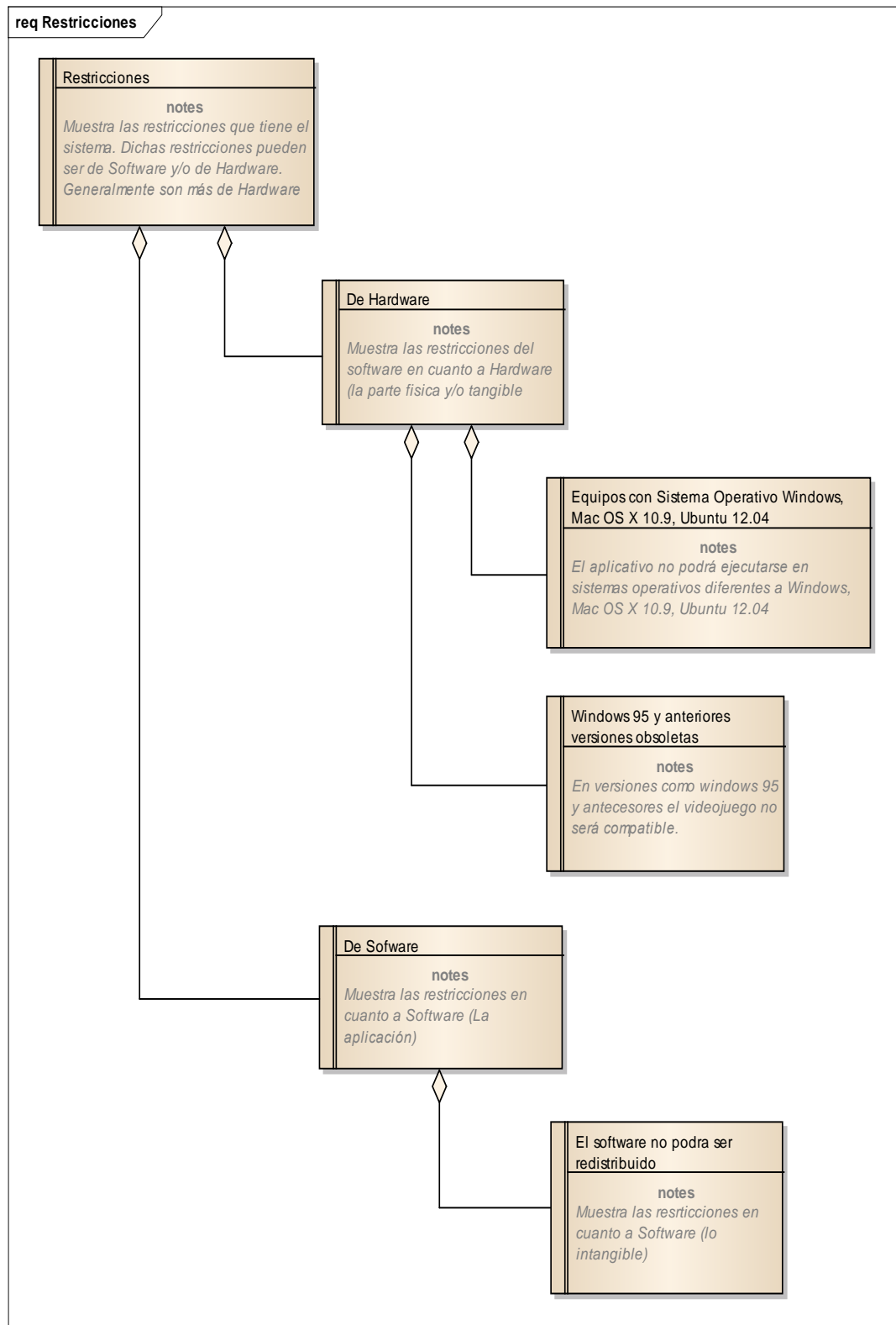


Ilustración 18. Diagrama de las restricciones que contiene el sistema.

Modelo de Análisis

- **Arquitectura de Clases**

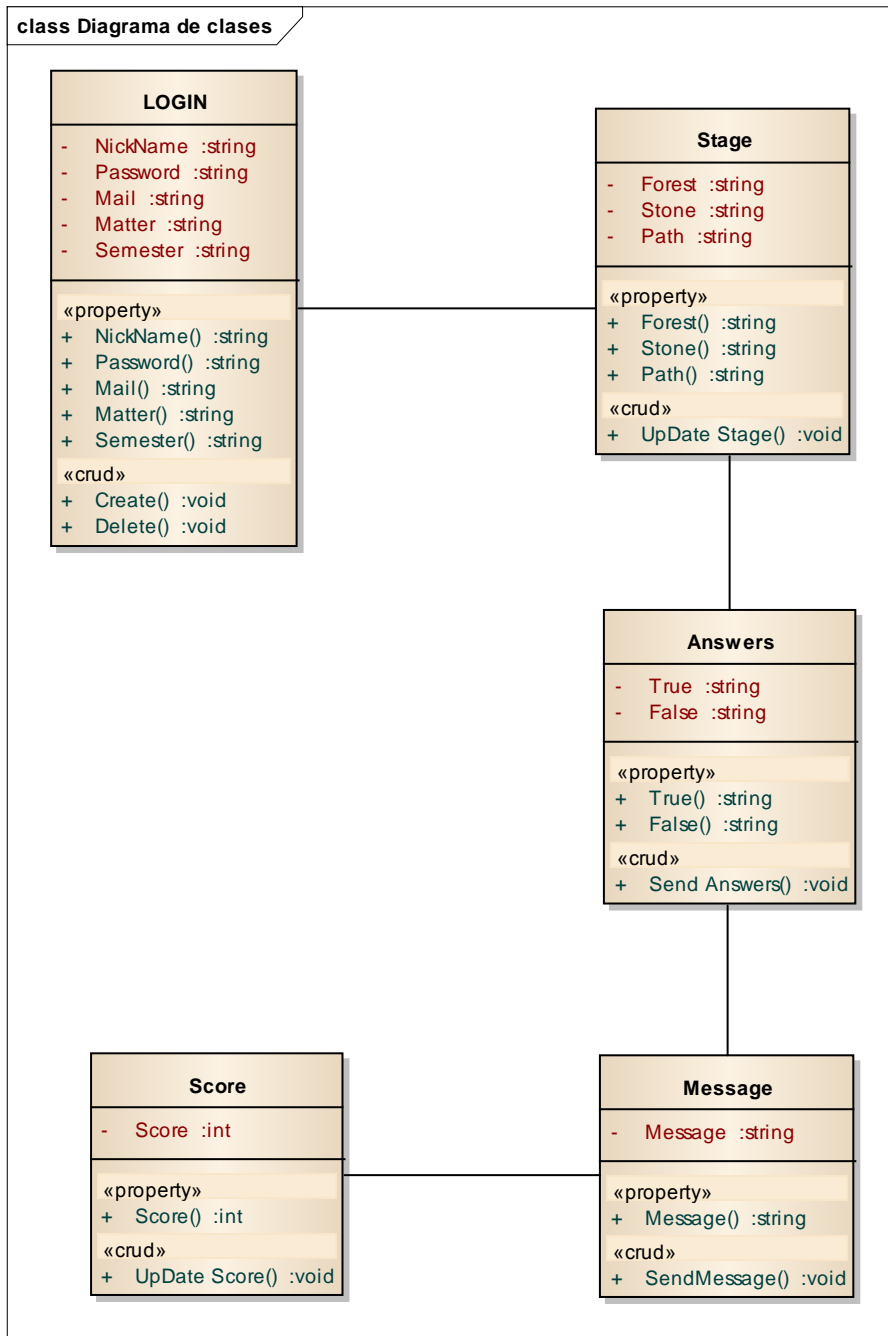


Ilustración 19. Diagrama de Clases.

- **Diagramas de Secuencia**

Dentro del siguiente diagrama de secuencia, se describe el inicio del juego donde muestra la opción de registrar usuario o si ya se encuentra registrado iniciar directamente sesión.

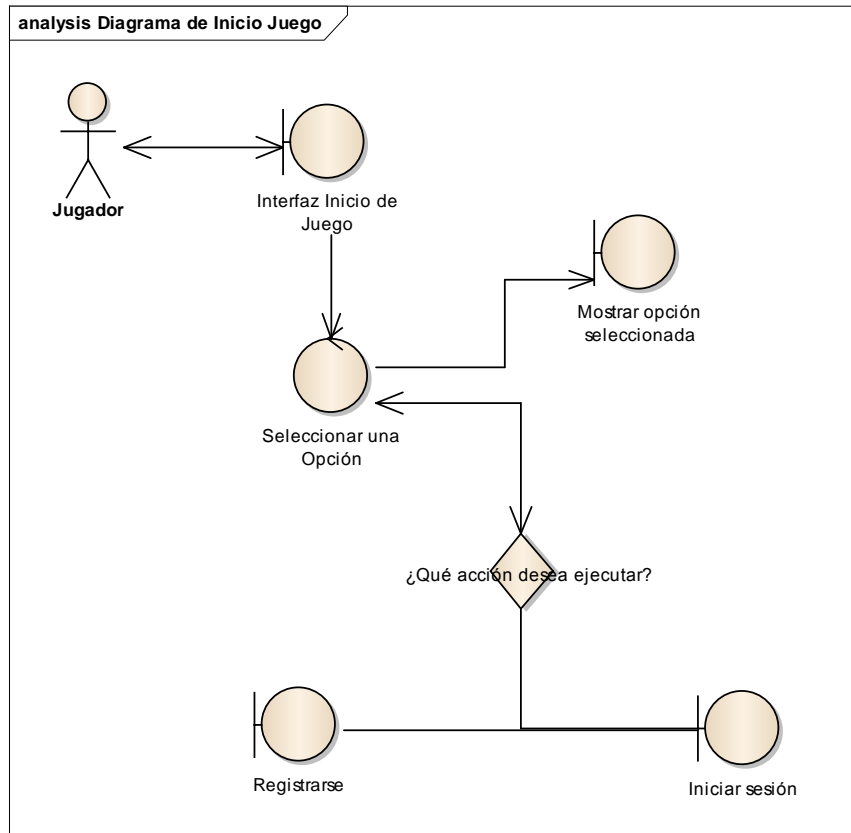


Ilustración 20. Diagrama de inicio de juego.

En la ilustración 20, se observa como el jugador debe hacer el registro de sus datos, llenando un formulario creando su usuario y contraseña, para guardarlos en el sistema.

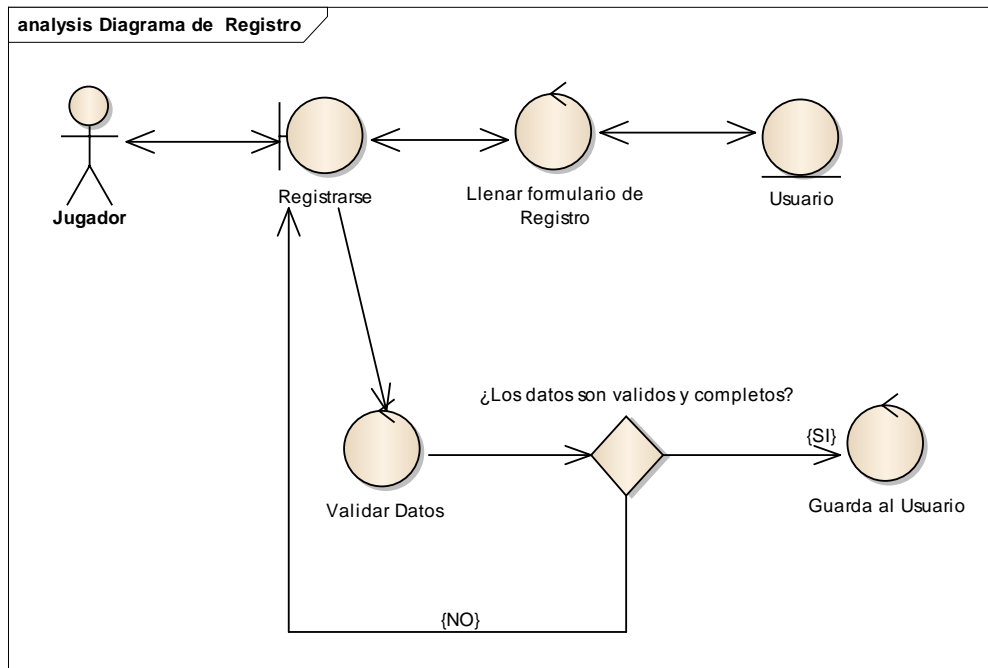


Ilustración 21. Diagrama de registro de usuario.

En la ilustración 21, se describe la validación de usuario y contraseña que debe realizar el jugador para acceder a la aplicación.

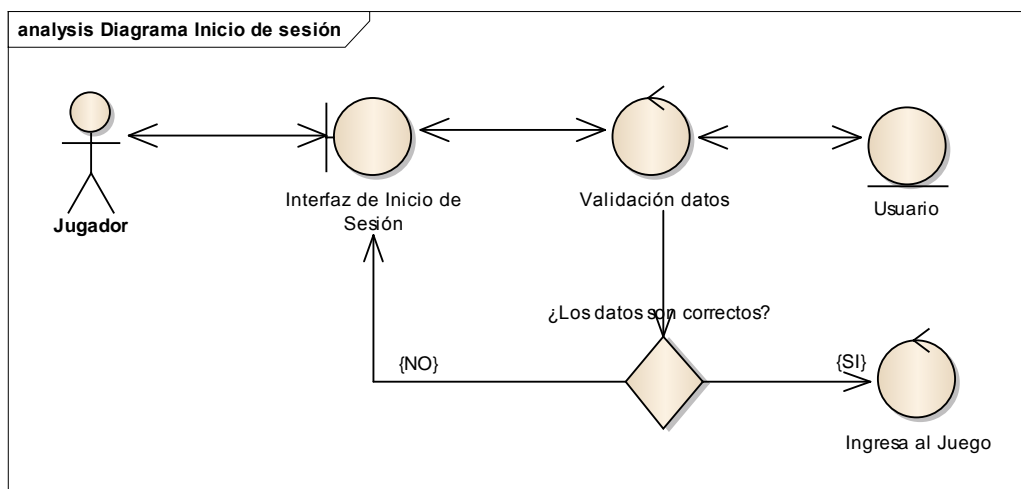


Ilustración 22. Diagrama de inicio de sesión.

En la ilustración 22, se observa como el jugador interactúa en el juego superando obstáculos y respondiendo preguntas, las cuales si son contestadas de forma correcta el sistema aumentan el puntaje del jugador.

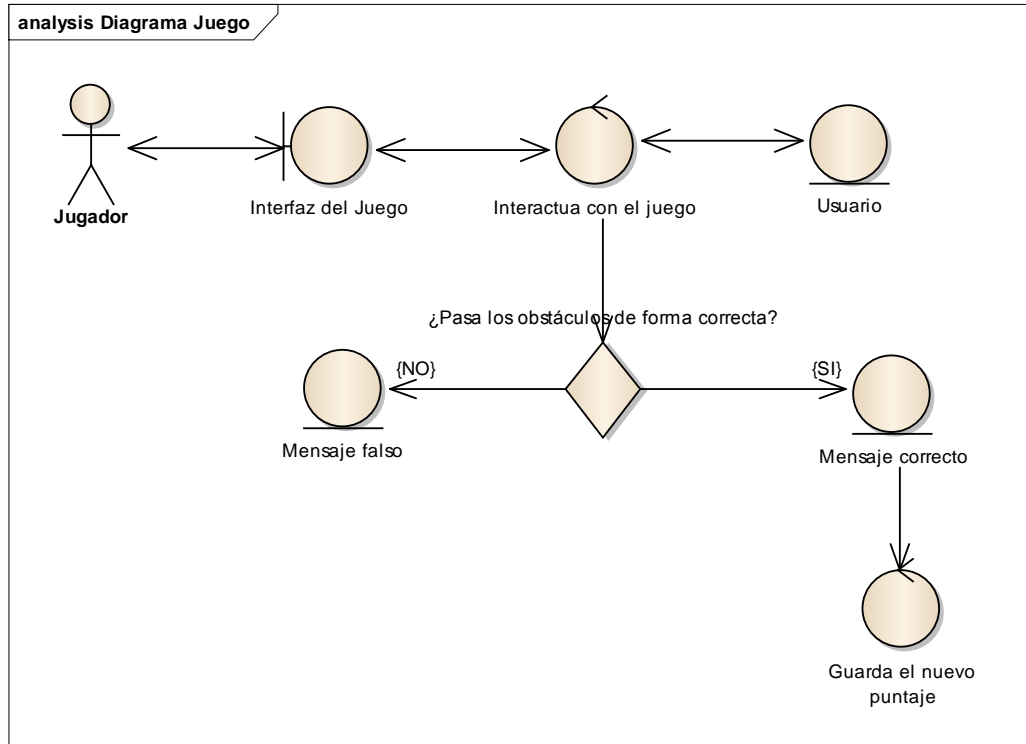


Ilustración 23. Diagrama del juego.

- **Mapa de Navegación.**



Ilustración 24. Mapa de Navegación del juego.

Se inicia por la creación del terreno, a continuación se da click en el recuadro edit textures donde nos arroja otro cuadro, en el cual se selecciona la textura que se desea para el terreno. Para el color principal se seleccionó un verde, el cual está guardado en el Asset Nature.

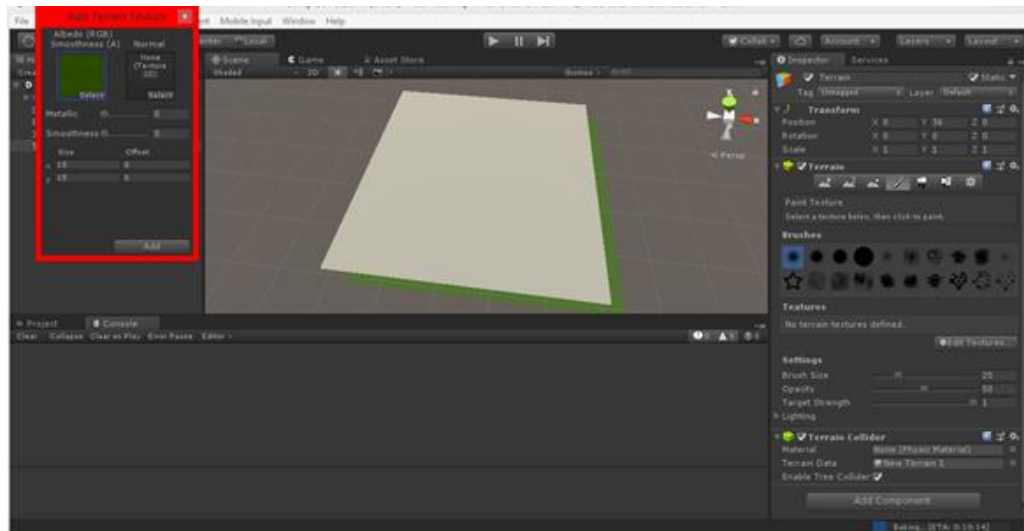


Ilustración 25. Selección del color que llevara el terreno del juego.

El cual al dar click en adherir, pintara automáticamente todo el terreno.

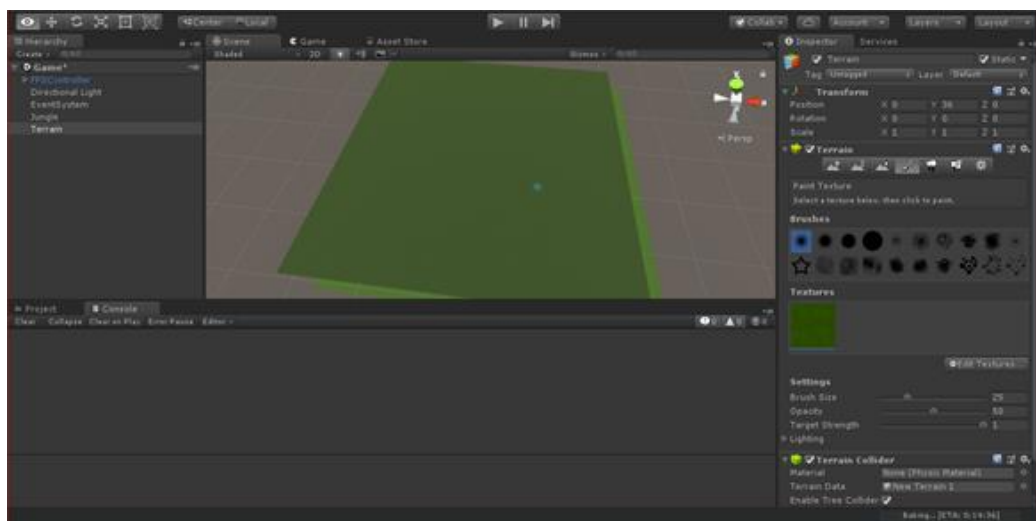


Ilustración 26. Pintado automático del terreno del juego.

Para el color secundario del terreno, el cual será el de los caminos, se escogió un marrón, para simular la tierra, seleccionando el tamaño del pincel para hacer los caminos.



Ilustración 27. Simulación de los caminos del Terreno del juego.

Se coloca el first person controller, el cual va a ser el personaje de este videojuego

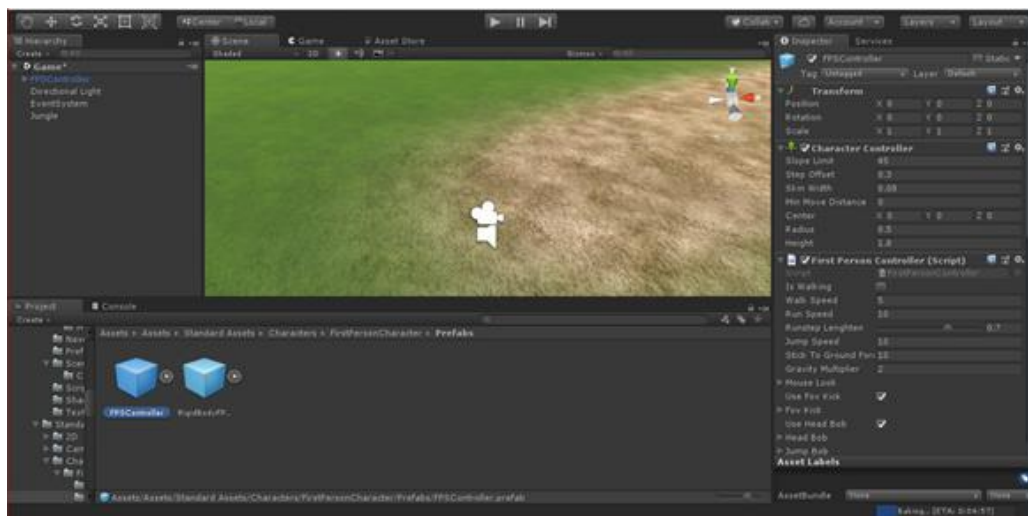


Ilustración 28. Selección personaje del juego para el desarrollo de las pruebas.

Dando una perspectiva en primera persona y ya tiene los controles de manejo por defecto (los que se han definido inicialmente, estos pueden ser modificados posteriormente por el jugador).

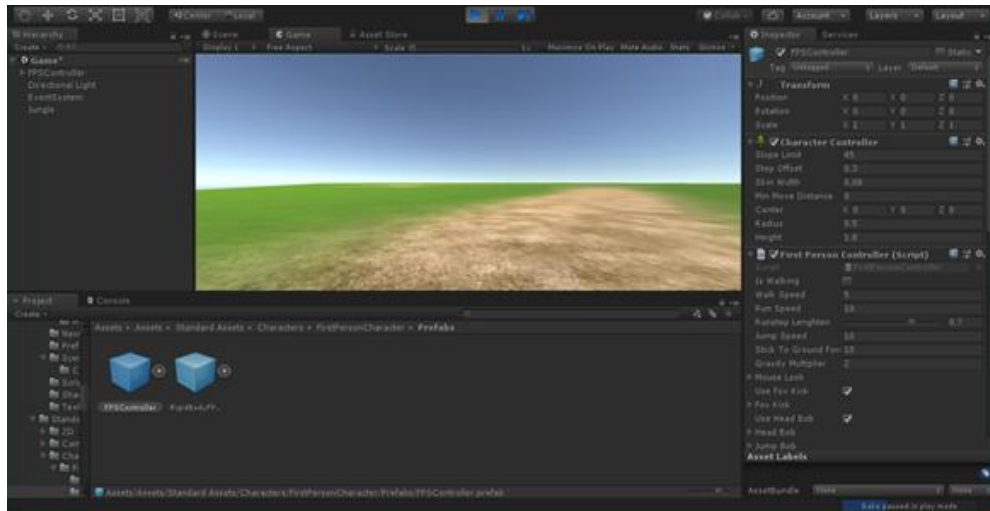


Ilustración 29. Controles de manejo del juego en el terreno para el desarrollo de las pruebas.

Para la creación de la naturaleza como césped, Flores, arboles entre otros, se va a utilizar el Asset Nature Starter kit 2, que son recursos para desarrollar videojuegos donde contiene 4 modelos de árboles, 6 modelos de arbustos y 2 de hierbas o pastos. (Unity, 2017)

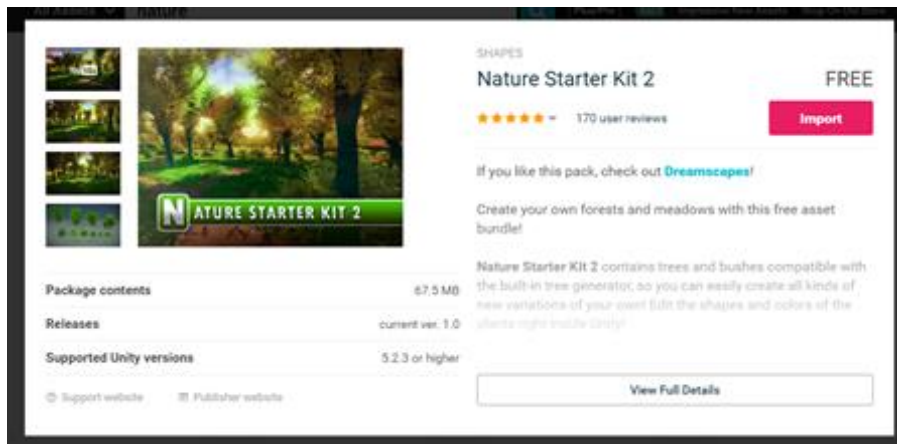


Ilustración 30. Selección del Nature Starter que se incorporara para el diseño del entorno del juego.

Para las rocas se usó el Asset Rocks free pack, que contiene 6 modelos de texturas de rocas gratis.

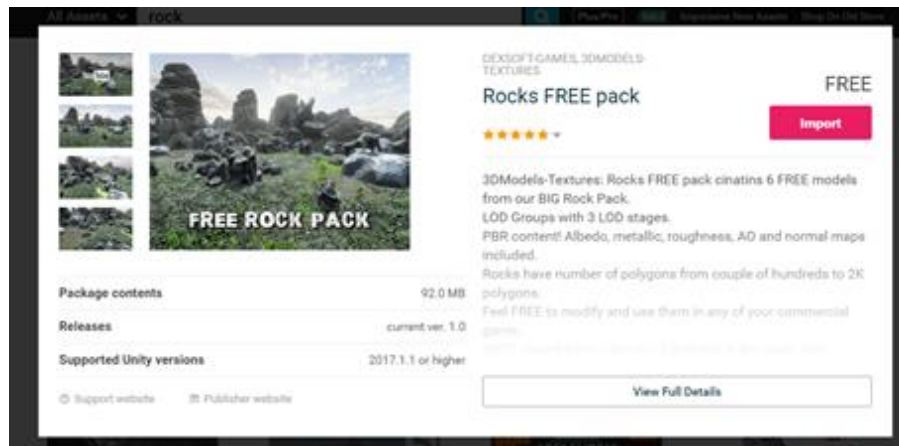


Ilustración 31. Selección del Rocks FREE Pack para dar la textura de las rocas que llevara el juego.

Para la creación del césped se selecciona el terreno, y se da click en pain details, en seguida en edit details y Add Grass texture arrojándonos un cuadro emergente el cual permite seleccionar la textura del pasto, la cual ya se encuentra guardada en el Asset Nature que se importó previamente. Se hicieron dos texturas de pasto y una de flores a la cual se le varía los colores. Para implementar estas texturas de pasto en el terreno, tenemos herramientas de pincel, así como las que se utilizaron para hacer el camino, además de establecer que tan grande es el área de implementación del pasto y que tan saturado de esta textura se desea.

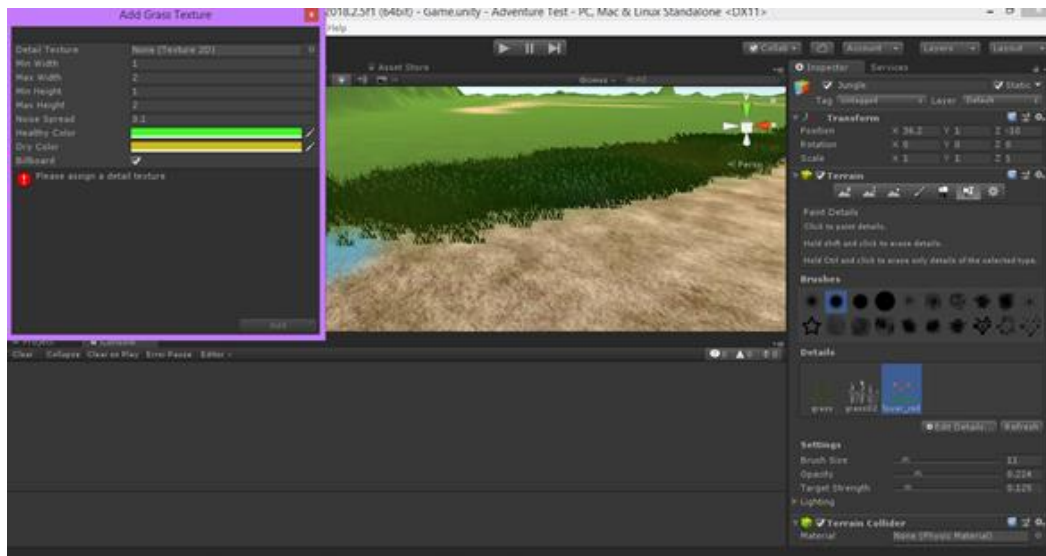


Ilustración 32. Implementación de la textura que lleva el pasto en el terreno del juego.

Para agregar los árboles, los cuales también se encuentran en el Asset Nature, se selecciona el terreno y se da click en la opción Paint Tree, luego se selecciona para editarla arrojándonos una ventana emergente donde se puede colocar cualquiera de los árboles o

arbustos que ya están listos, de la misma manera que se colocó el césped, se procede a implementar los árboles. Para así obtener este resultado grafico en 3d.

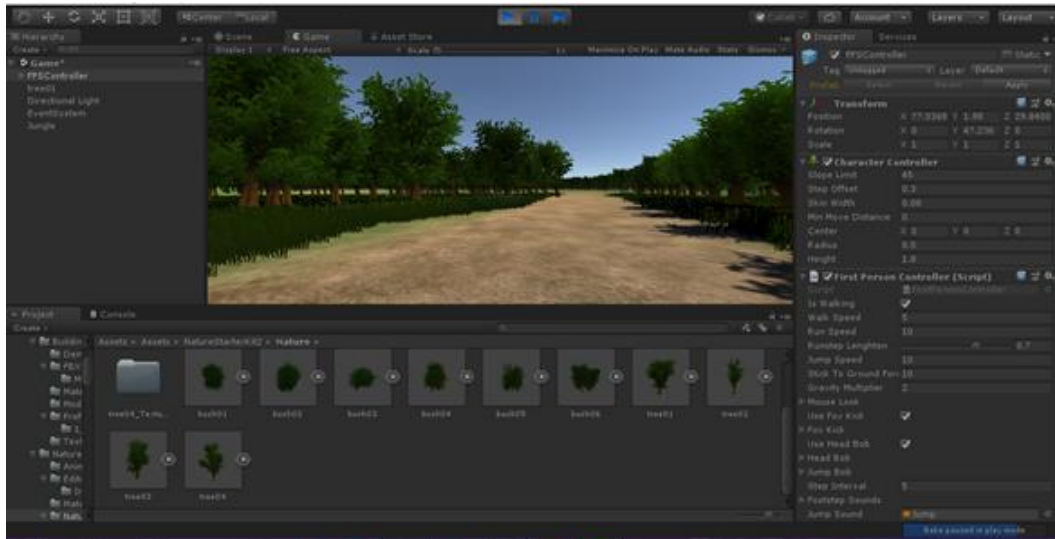


Ilustración 33. Implementación de los árboles que irán incorporados en el ambiente del terreno.

Se seleccionan todos los objetos creados, para poder deshabilitar la opción de Mesh Render, la cual permite hacer los objetos visibles o no. Se deshabilita, para que los objetos no sean visibles, pero que aun así el personaje se pueda colisionar con ellos, lo que le permite ver a su alrededor, pero que no se le permita salir del camino.



Ilustración 34. Mesh Render para lograr el efecto que el personaje no salga del camino.



Ilustración 35. Enemigo

A continuación de esto se crea el login; donde se emplea un canvas, se procede a colocar los Input Field los cuales son campos de texto, que en este caso permitirán recolectar la información de los jugadores, implementando el label, para referenciar cual es la casilla de Nombre del usuario y cuál es la casilla para ingresar su contraseña.

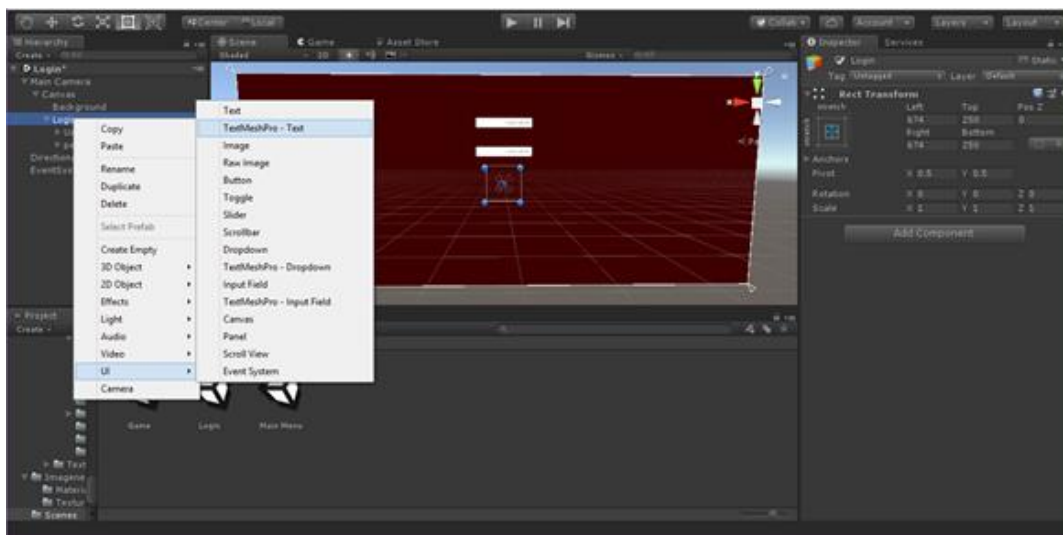


Ilustración 36. Creación login y contraseña para que el jugador pueda ingresar al juego.

Una vez terminados los campos para el ingreso de la información y colocados los label que delimitan para que es cada campo, se procede a implementar los botones, donde una vez esté listo el menú del login, se procede a crear el menú de registro, para lo cual se desactiva temporalmente el de login.

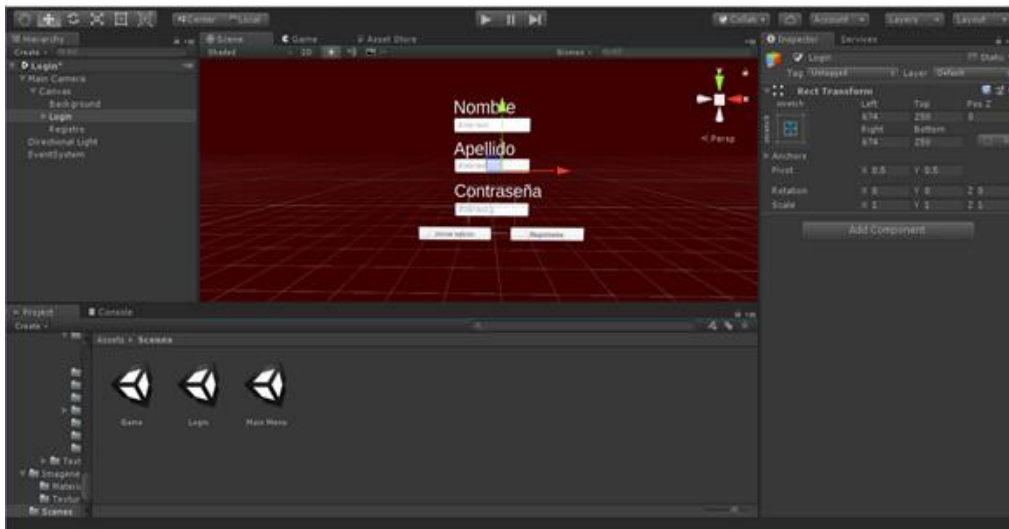


Ilustración 37. Autenticación del jugador.

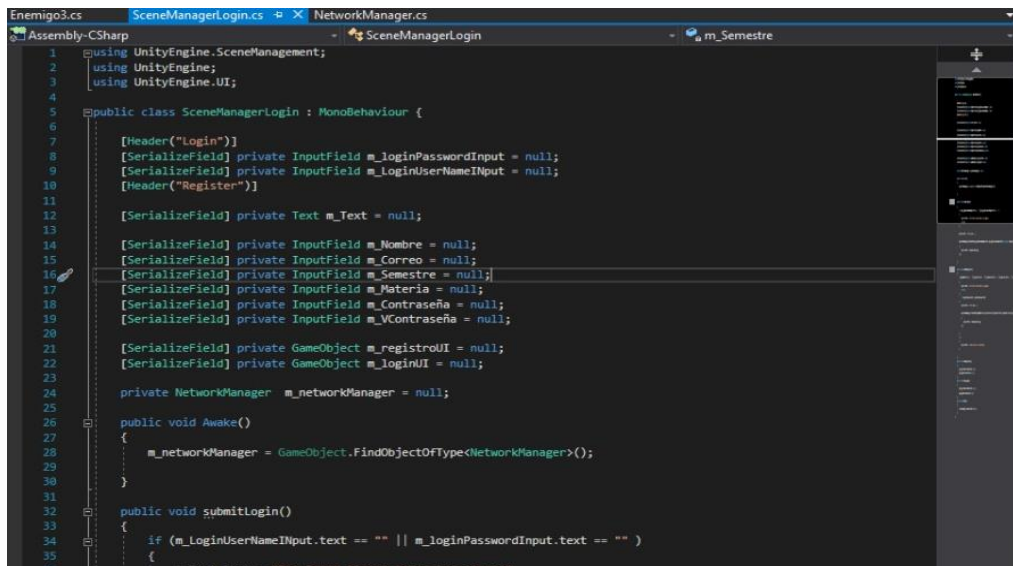


Ilustración 38. Requerimientos para llenar la base de datos.


```

34     if (m_LoginUserNameInput.text == "" || m_loginPasswordInput.text == "")
35     {
36         m_Text.text = "Por Favor llenar todos los campos";
37         return;
38     }
39
40     m_Text.text = "Procesando...";
41
42     m_networkManager.CheckUser(m_loginUserNameInput.text, m_loginPasswordInput.text, delegate (Response response)
43     {
44         m_Text.text = response.message;
45     });
46
47 }
48
49 public void submitRegister()
50 {
51     if (m_Nombre.text == "" || m_Correo.text == "" || m_Semestre.text == "" || m_Materia.text == "" || m_Contraseña.text == "" || m_VContraseña.text == "")
52     {
53         m_Text.text = "Por Favor llenar todos los campos";
54         return;
55     }
56     if (m_Contraseña.text == m_VContraseña.text)
57     {
58         m_Text.text = "Procesando...";
59
60         m_networkManager.CreateUser(m_Nombre.text, m_Correo.text, m_Semestre.text, m_Materia.text, m_Contraseña.text, m_VContraseña.text, delegate (Response response)
61         {
62             m_Text.text = response.message;
63         });
64     }
65 }
66

```

Ilustración 39. Requerimientos para llenar la base de datos.

```

1  using System.Collections;
2  using System;
3  using System.Collections.Generic;
4  using UnityEngine;
5
6  public class NetworkManager : MonoBehaviour
7  {
8
9      public void CreateUser(string Nombre, string Correo, string Semestre, string Materia, string Contraseña, Action<Response> response)
10     {
11         StartCoroutine(CO_CreateUser(Nombre, Correo, Semestre, Materia, Contraseña, response));
12     }
13     private IEnumerator CO_CreateUser(string Nombre, string Correo, string Semestre, string Materia, string Contraseña, Action<Response> response)
14     {
15         WWWForm form = new WWWForm();
16         form.AddField("Nombre", Nombre);
17         form.AddField("Correo", Correo);
18         form.AddField("Semestre", Semestre);
19         form.AddField("Materia", Materia);
20         form.AddField("Contraseña", Contraseña);
21
22
23         //WWW w = new WWW("https://localhost/Game/createUser.php", form);
24         WWW w = new WWW("https://localhost/Game/createUser.php");
25
26
27         Debug.Log(w.text);
28         response( JsonUtility.FromJson<Response>(w.text) );
29         yield return w;
30     }
31 }
32 public void CheckUser(string Correo, string Contraseña, Action<Response> response)
33 {

```

Ilustración 40. Código para conectar la base de datos.

```

1 <?php
2 include "dbConnection.php";
3
4 /*$Correo = $_POST['Correo'];
5 $Semestre = $_POST['Semestre'];
6 $Materia = $_POST['Materia'];
7 $Contraseña = hash("sha256", $_POST['Contraseña']);*/
8
9 $Nombre = 'andres';
10 $Correo = 'Correo';
11 $Semestre = 'Semestre';
12 $Materia = 'Materia';
13 $Contraseña = hash("sha256", '123');
14
15 $sql = "SELECT Nombre From usuarios WHERE Nombre = '$Nombre'";
16 $result = $pdo->query($sql);
17
18 if($result->rowCount() > 0)
19 {
20     $data = array('done' => false, 'message' => "Error! Nombre de Usuario ya existe");
21     Header('content-Type: application/json');
22     echo json_encode($data);
23     exit();
24 }
25 else {
26     $sql = "SELECT Correo From usuarios WHERE Correo = '$Correo'";
27     $result = $pdo->query($sql);

```

Ilustración 41. Conexión a la base de datos.

```

1 <?php
2
3
4 try {
5     $pdo = new PDO('mysql:host=localhost;dbname=Game', 'Boss', '1234');
6     $pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
7     $pdo->exec('SET NAMES "utf8" ');
8
9 }
10 catch (PDOException $e)
11 {
12     echo "ERROR CONECTING TO DATABASE " . $e->getMessage();
13     exit();
14 }
15 }
16
17
18 ?>
19

```

Ilustración 42. Conexión a la base de datos con atom.

```

1 <?php
2 include "dbConnection.php";
3
4 /*$Correo = $_POST['Correo'];
5 $Semestre = $_POST['Semestre'];
6 $Materia = $_POST['Materia'];
7 $Contraseña = hash("sha256", $_POST['Contraseña']);*/
8
9 $Nombre = 'andres';
10 $Correo = 'Correo';
11 $Semestre = 'Semestre';
12 $Materia = 'Materia';
13 $Contraseña = hash("sha256", '123');
14
15 $sql = "SELECT Nombre From usuarios WHERE Nombre = '$Nombre'";
16 $result = $pdo->query($sql);
17
18 if($result->rowCount() > 0)
19 {
20     $data = array('done' => false, 'message' => "Error! Nombre de Usuario ya existe");
21     Header('content-Type: application/json');
22     echo json_encode($data);
23     exit();
24 }
25 else {
26     $sql = "SELECT Correo From usuarios WHERE Correo = '$Correo'";
27     $result = $pdo->query($sql);

```

Ilustración 43. Llena y almacena información a la base de datos.

8.2 Costo del proyecto

Recurso	Costo	
	Mensual (Cm)	Total=Cm*No de meses
Papelería	\$12.000	\$144.000
Transporte	\$50.000	\$600.000
Internet	\$35.000	\$420.000
Hardware	\$20.000	\$240.000
Imprevistos	\$20.000	\$240.000
Otros	\$15.000	\$180.000
	Total	\$1.824.000

Como se puede observar en la tabla se presentan los gastos mensuales que se requirieron para el desarrollo del proyecto, como lo son elementos de papelería e impresiones, el uso de transportes, el coste de la navegación en internet para realizar las consultas, temas de hardware y medios masivos de almacenamiento y una reserva para ser usada en gastos que surgían de imprevisto. Estos gastos se acoplaron a lo largo de 1 año que fue el transcurso de tiempo en el cual fue desarrollado el proyecto para finalmente hacer una sumatoria de los gastos realizados en el año y así arrojar un total de \$1.824.000 los cuales fueron invertidos en el desarrollo del proyecto a lo largo del año.

9. PRUEBAS

9.1 Pruebas de diseño

- A medida que se desarrolló el juego se iban realizando las correcciones o cambios necesarios respecto a la escena, ya que por medio del compilado automático del código de Unity al ejecutarlo permitía ver los errores existentes.
- Con respecto al entorno también se realizaron las pruebas pertinentes empezando con el personaje explorando su desplazamiento por este, observando si los objetos estaban bien diseñados o no, tales como los muros o estructuras por los cuales debe atravesar el personaje si la ubicación de los objetos era la correcta o si era necesario añadir más componentes. A medida que se iban modificando o creando cosas en el script se compilaba para no ir acumulando errores de código.
- En cuanto a la interfaz las pruebas se hacían “a ojo” repasando continuamente que los elementos se situaran en la posición indicada.

9.2 Pruebas generales

Con la ayuda de una muestra de 16 estudiantes de la asignatura Programación I se comprobó desde un punto objetivo que el juego:

- Al momento en el que se abre, cargara debidamente la interfaz gráfica. Observando la impresión que causa si es visualmente llamativa o no para el usuario, además que estuvieran habilitadas las funciones que esta aplicación debe cumplir.
- Se comprobó si el sistema al momento de iniciar el juego estaba solicitando la autenticación del usuario, y de no estar registrado informe que es inexistente en la base de datos.

Ilustración 44. Autenticación del usuario.

- En el momento en el que el usuario registra sus datos, se observó que aparecieran guardados en la base de datos del juego.

Ilustración 55.Registro de un nuevo usuario.

Ilustración 46. Guardado en la base de datos.

- Al momento de jugar si el usuario contesta correctamente las preguntas, se verifico que el contador aumentara el puntaje según lo indicado.

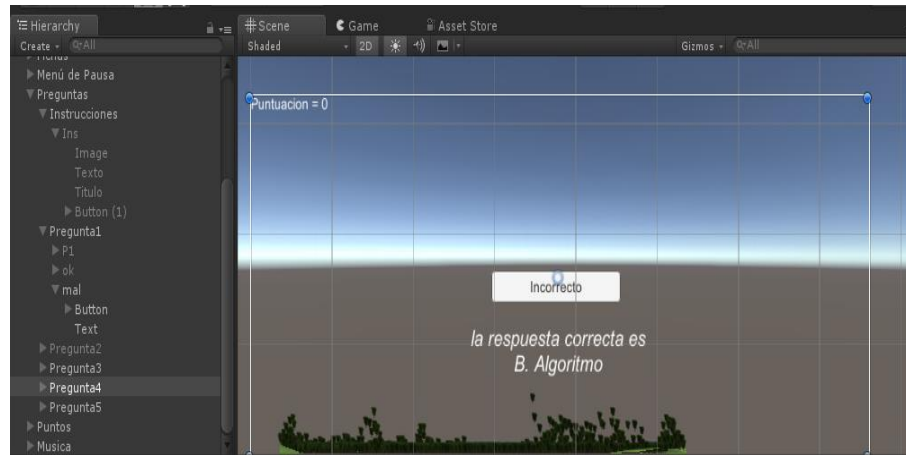


Ilustración 67. Puntuación en cero.

- Finalmente, por medio de una encuesta se midió el grado de satisfacción de los jugadores con respecto al juego y si consideraban necesario o no que fuera incorporado a la metodología como una herramienta de apoyo.

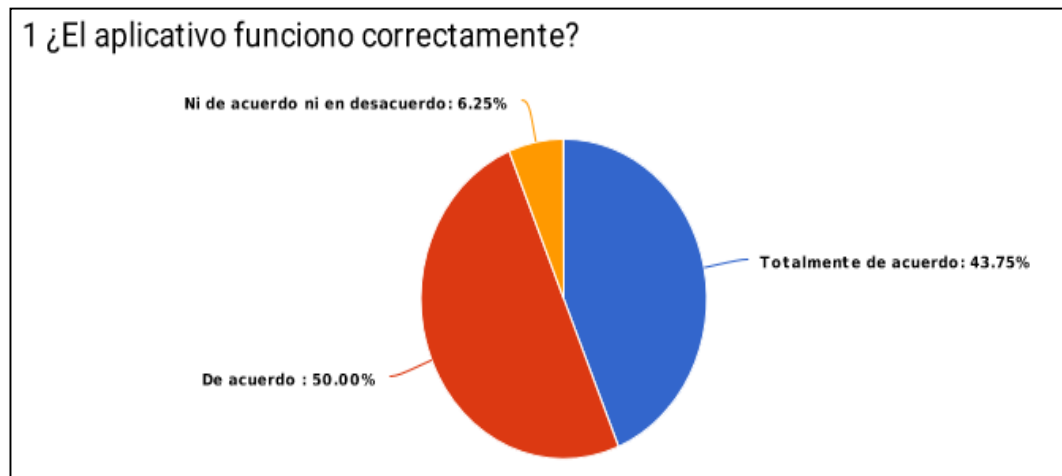


Ilustración 78. Encuesta sobre el juego.

2 ¿Considera que el aplicativo refuerza sus conocimientos con respecto a la lógica de programación?

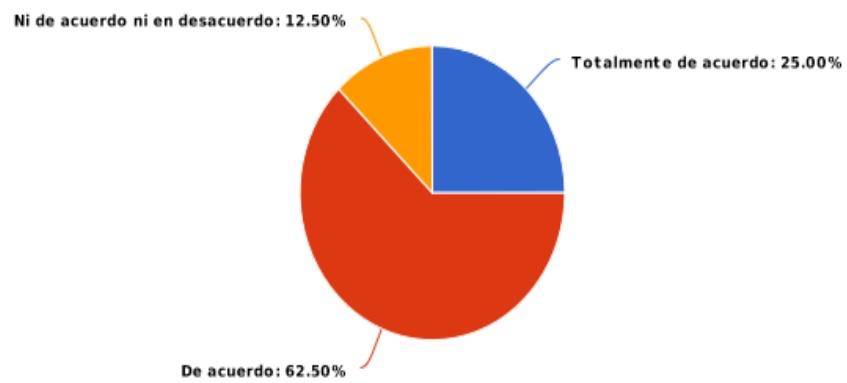


Ilustración 89.2 Encuesta del juego.

3 ¿Cómo evalúa nuestro aplicativo?

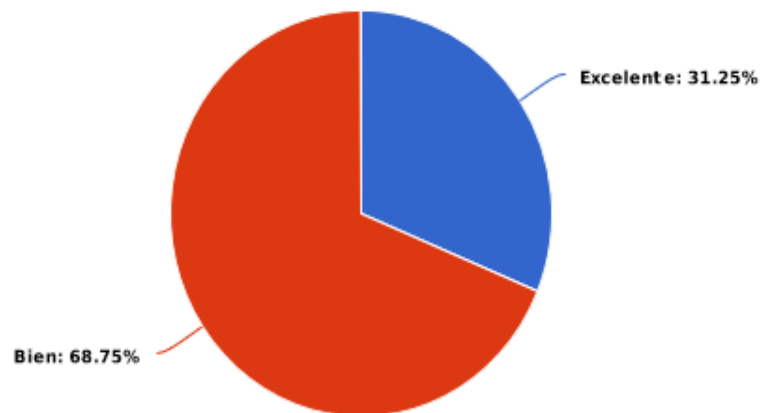


Ilustración 50. 3 encuesta del juego.

4 ¿Considera que en los núcleos temáticos de "lógica y algoritmia" y "programación I" se debe implementar este tipo de herramienta metodológica para reforzar los conocimientos de los estudiantes?

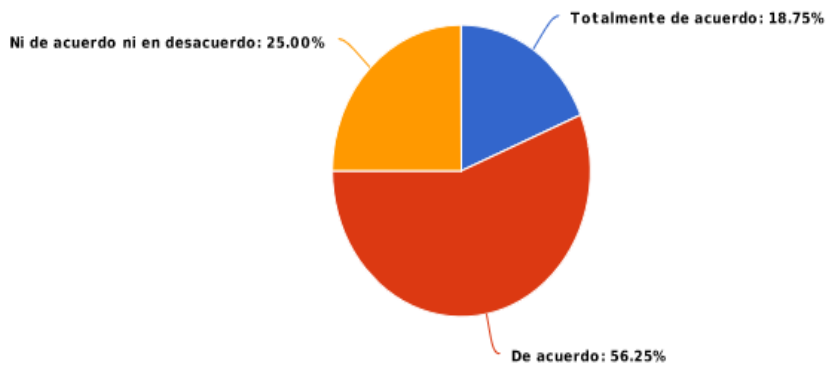


Ilustración 51. 4 encuesta del juego.

CAPITULO 4

9. CONCLUSIONES

Por medio de la implementación de herramientas de obtención de información en fuentes primarias, como la encuesta realizada a los estudiantes de “lógica y algoritmia” y “programación I”, junto con la búsqueda sistémica de información en fuentes indexadas, se pudo establecer el modelo bajo el cual se diseñó el videojuego para ayudar a reducir los vacíos con respecto a conceptos básicos de programación y algoritmos que se presentaban.

Con el desarrollo del videojuego, los componentes artísticos que este exige, y las necesidades educativas detectadas, es posible gestionar un sistema de alto nivel de interacción, que permita generar entretenimiento y medios de enseñanza no tradicionales basados en tecnologías disponibles, esto según el tema central desarrolle y refuerce los conocimientos adquiridos en base a una materia.

El género en perspectiva de primera persona brinda una experiencia más personal al no tener un personaje visible llevando a cabo las acciones que desarrolla el jugador en el ambiente virtual.

Con respecto a la mecánica del juego se notó cierta inconformidad al inicio con los métodos instruccionales dentro del juego, ya que al parecer los objetivos no eran del todo claros y la secuencia de tareas se rompía en algún punto del juego, como las siguientes:

- El jugador puede perder o ganar dependiendo de su destreza en el desarrollo del juego.
- Se genera la autenticación del jugador.
- La creación de un videojuego exige un profundo conocimiento en el género, ya que si no se cuenta con éste no es posible conocer las limitantes de desarrollo a múltiples niveles, ni los requerimientos exigidos por el público en la actualidad.

Es importante señalar que lo que se busca mediante el desarrollo de este proyecto, es fomentar el valor de la información, que, si no se puede adquirir de una manera o se genera gran dificultad para poder adquirirla, se puede analizar e implementar una nueva metodología. Trabajando en sincronía alumno y docente para poder alcanzar el principal objetivo de la enseñanza, el cual es adquirir los conocimientos de un tema en específico.

10. RECOMENDACIONES

Los estudiantes de los siguientes semestres interactúen aportando mejoras al videojuego en cuanto a estrategias y dinámicas que permitan contribuir de manera más acertada en el progreso académico.

Se realicen adaptaciones del videojuego por parte de la comunidad educativa en sistemas operativos diferentes al propuesto inicialmente.

11. PROYECCIONES

- Que este proyecto sea implementado en la Universidad de Cundinamarca reforzando posibles vacíos con respecto al tema de la lógica de programación.
- Realizar versiones enfocadas a las necesidades educativas que tengan interés de adquirir este videojuego, agregando un valor monetario.
- El videojuego fue diseñado para que los estudiantes de ingeniería de sistemas puedan realizar basándose dentro del contexto desarrollado niveles posteriores del mismo.

REFERENCIAS

- Android, A. (02 de 11 de 2015). *Academia Android*. Recuperado el 18 de 02 de 2018, de <https://academiaandroid.com/scripts-lenguajes-programacion-unity/>
- Apache. (2018). *Apache Friends*. Obtenido de <https://www.apachefriends.org/es/index.html>
- blogspot. (27 de mayo de 2011). *Informatica*. Obtenido de <http://informaticallc.blogspot.com/2011/05/sistema-aplicativo.html>
- Camargo, M. d. (2012). Educación Superior No Culmina Sus Estudios. *Semana*.
- Carballo, J. (11 de Noviembre de 2015). *computer hoy.com*. Obtenido de <http://computerhoy.com/noticias/internet/codingame-plataforma-aprender-programar-jugando-36703>
- Castillo, j. g., & Diaz, u. j. (2012). *sistema de gestión bibliotecaria*. Obtenido de sistema de gestión bibliotecaria: <http://catalogo.ucundinamarca.edu.co:2676/cgi-bin/koha/opac-detail.pl?biblionumber=24105>
- Centro de Estudios Superiores Felipe II, U. C. (10 de 2006). *Engieering Village*. Recuperado el 15 de 2 de 2018, de https://e-biblioteca.ucundinamarca.edu.co:2148/search/doc/abstract.url?&pageType=quickSearch&usageZone=resultlist&usageOrigin=searchresults&searchtype=Quick&SEARCHID=81aa5239M7ce7M4367Mb3f2Ma66b1ed14f76&DOCINDEX=1&ignore_docid=cpx_30c22110f06810d25M73292
- CodeCombat, I. (2018). *CODE COMBAT*. Obtenido de CODE COMBAT: <https://codecombat.com/>
- Combat, C. (2013). *CODECOMBAT*. Recuperado el 10 de 2 de 2018, de <https://codecombat.com/about>
- Comite, t. d. (2009). *Grupo de Investigación - S@R@*. Obtenido de Grupo de Investigación - S@R@: [Guias%20Trabajos%20Grado.pdf](#)
- company, C. S. (s.f.). *CODE SCHOOL*. Obtenido de <https://www.codeschool.com/about>
- Desmoulins, F. (11 de noviembre de 2015). *Computer Hoy.com*. Obtenido de Computer Hoy.com: <https://computerhoy.com/noticias/internet/codingame-plataforma-aprender-programar-jugando-36703>

- Encyclopædia Britannica, I. (2017). *ENCYCLOPEDIA BRITANNICA*. Obtenido de
 ENCYCLOPEDIA BRITANNICA:
<https://www.britannica.com/search?query=+design>
- Fernandez, s. (2016). *Previa*. Obtenido de <https://previa.uclm.es/>
- Francisco J. Gallego, R. M. (11 de julio de 2014). *Gamificar una propuesta docente*.
 Obtenido de Gamificar una propuesta docente:
[https://rua.ua.es/dspace/bitstream/10045/39195/1/Gamificacio%CC%81n%20\(defin%CC%81n\).pdf](https://rua.ua.es/dspace/bitstream/10045/39195/1/Gamificacio%CC%81n%20(defin%CC%81n).pdf)
- Garcia, V. (14 de Enero de 2005). *EcuRed*. Obtenido de
<https://www.ecured.cu/T%C3%A9cnica>
- IIEMD. (26 de Enero de 2017). *IIEMD*. Recuperado el 14 de Noviembre de 2018, de IIEMD:
<https://iiemd.com/gamificacion/que-es-gamificacion>
- Johnson, S. (abril de 2001). *Sistemas Emergentes O qué tienen en común hormigas, neuronas, ciudades y software*. Obtenido de
https://scielo.conicyt.cl/scielo.php?script=sci_arttext&pid=S0250-71612008000100008
- McMillan, K. (2017). *sparx systems* . Obtenido de sparx systems :
<http://sparxsystems.com/products/ea/>
- Nacional, M. d. (2009). *Deserción Estudiantil*. Recuperado el 20 de 2 de 2018, de
https://www.mineducacion.gov.co/sistemasdeinformacion/1735/articles-254702_libro_desercion.pdf
- Oracle, C. (2018). *MySQL*. Obtenido de MySQL: <https://www.mysql.com/ProcesosdeSoftware>. (14 de Abril de 2016). *Metodología RUP*. Obtenido de
<https://procesosdesoftware.wikispaces.com/METODOLOGIA+RUP#discussion>
- RAE.ES. (2017). *REAL ACADEMIA ESPAÑOLA* . Obtenido de
<http://dle.rae.es/?id=2Vga9Gy>
- Research, M. (s.f.). *Code Hunt*. Obtenido de <https://www.codehunt.com/about.aspx>
- SLASH, H. '. (09 de SEP de 2014). *STEAM*. Recuperado el 15 de 2 de 2018, de
http://store.steampowered.com/app/246070/Hack_n_Slash/?l=spanish
- Sparx, S. (2018). *sparxsystems*. Obtenido de sparxsystems: <https://sparxsystems.com/>

SUM. (2009). *SUM*. Recuperado el 12 de 3 de 2018, de Online:
<http://www.gemserk.com/sum>

Technologies, U. (2018). *Unity*. Obtenido de <https://unity3d.com/es>

Udima. (6 de 10 de 2017). Obtenido de UNIVERSIDAD A DISTANCIA DE MADRID:
<https://www.udima.es/es/estudio-gamificacion-educacion-superior-udima-sonia-pamplona.html>

Unity, c. (2017). *Unity Tridevent*. Obtenido de Unity Tridevent:
<http://unity.tridevent.com/nature-starter-kit-2/>

Wikitendo. (10 de Octubre de 2011). *Nintendo Wiki*. Recuperado el 14 de Noviembre de 2018, de Nintendo Wiki: [http://es.nintendo.wikia.com/wiki/Videojuego_\(concepto\)](http://es.nintendo.wikia.com/wiki/Videojuego_(concepto))

ANEXO A

Encuesta Dificultades En La Lógica De Programación

1 Que tan difícil le parece la lógica de programación en una escala de 1 a 5; siendo 1 el nivel más bajo y 5 el nivel más alto?

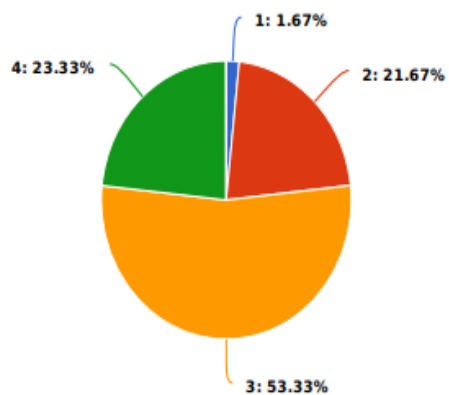


Ilustración 9. Representación de respuestas a la pregunta 1 de la encuesta.

Tabla 1

Respuesta	Porcentaje	Cantidad
1	1.67%	1
2	21.67%	13
3	53.33%	32
4	23.33%	14
5	0.00%	0
Total de respuestas		60

2 cuál cree usted qué es el principal motivo por el cual los estudiantes presentan dificultades en la parte de la lógica de programación?

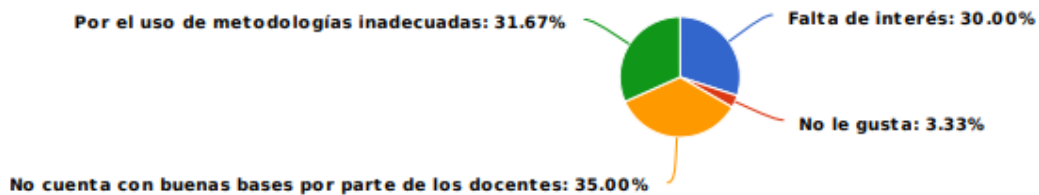


Ilustración 38. Respuestas a la pregunta 2 de la encuesta

Respuesta	Porcentaje	Cantidad
Falta de interés	30.00%	18
No le gusta	3.33%	2
No cuenta con buenas bases por parte de los docentes	35.00%	21
Por el uso de metodologías inadecuadas	31.67%	19
Total de respuestas		60

3 Si usted reprobara en el curso de lógica de programación esto lo haría sentir que debe retirarse o cambiar de carrera?

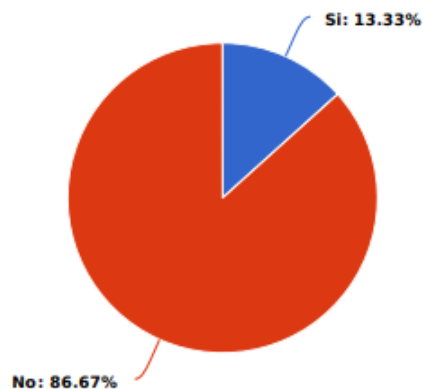




Ilustración 39. Representación de las respuestas a la pregunta 3 de la encuesta

Respuesta	Porcentaje	Cantidad
Si	13.33% 	8
No	86.67% 	52
Total de respuestas		60

ANEXO B

Elaboración del Videojuego

Creación del terreno

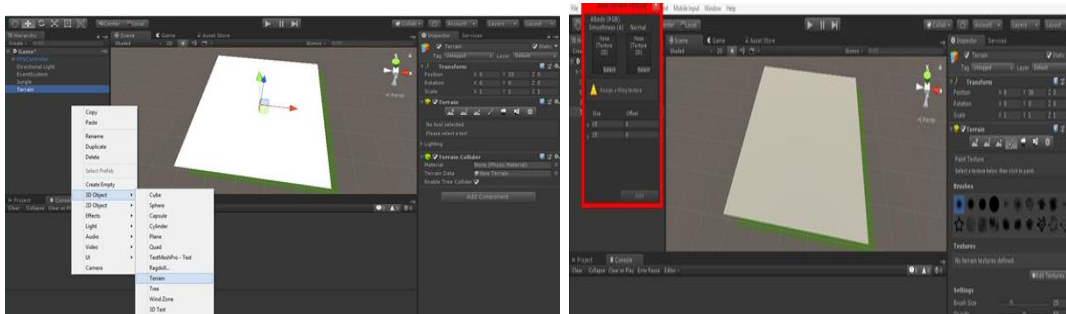


Ilustración 48. Primer paso en la creación del terreno.

Se da click en el recuadro edit textures apareciendo otro cuadro, en el cual se agrega la textura que se desea para el terreno, para el color principal se utilizó un verde, el cual está guardado en el Asset Nature.

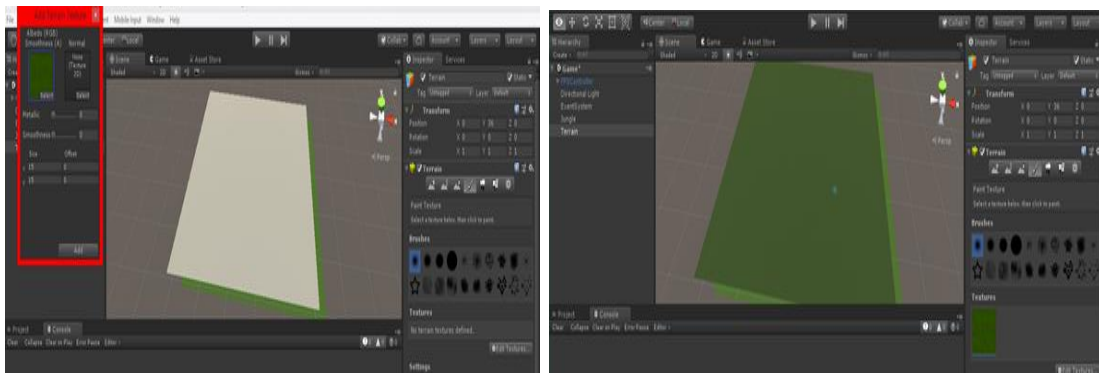


Ilustración 109. Adición de texturas para el terreno.

Para el color secundario del terreno, el cual será el de los caminos, se seleccionó un marrón, para simular la tierra, se selecciona el tamaño del pincel para hacer los caminos.

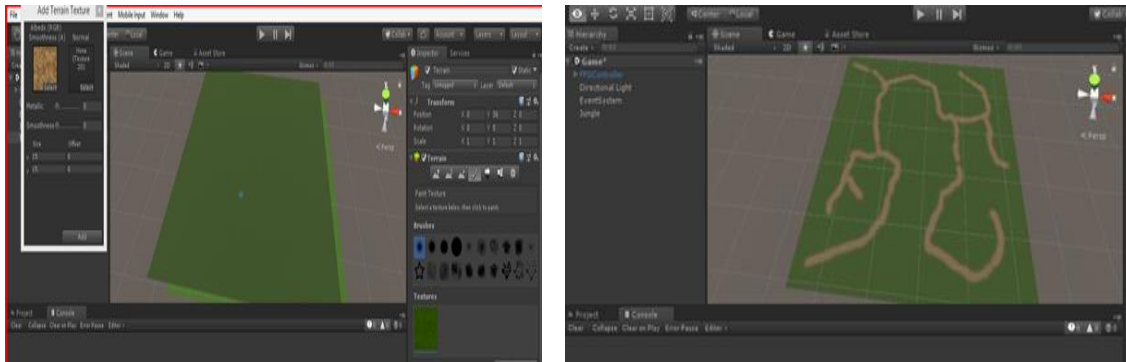


Ilustración 50. Inserción de caminos.

Se coloca el first person controller, el cual va a ser el personaje de este videojuego. Dando una perspectiva en primera persona y ya tiene los controles de manejo por default.

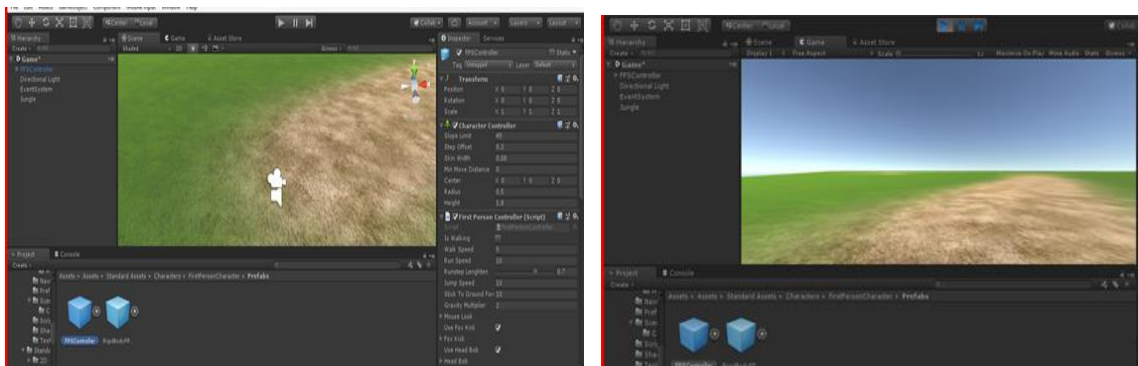


Ilustración 51. Ubicación del personaje en el terreno.

Para la creación de la naturaleza como césped, Flores, arboles entre otros, se utilizó el Asset Nature Starter kit 2 y para las rocas se usa el Asset Rocks free pack.

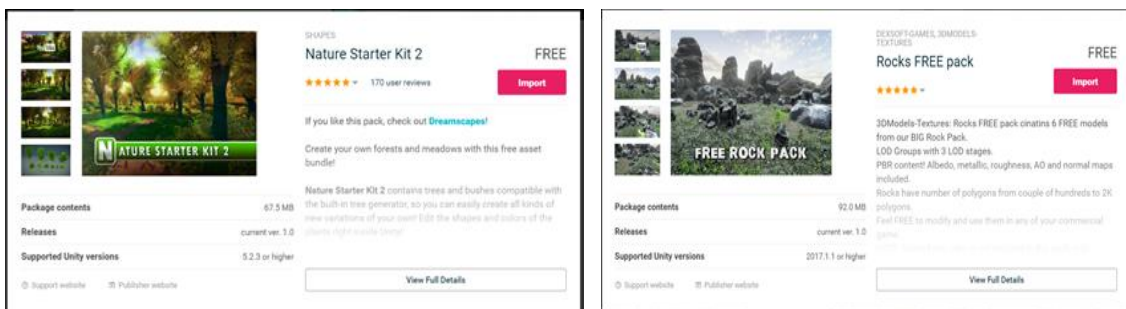


Ilustración 52. Importación de los Asset utilizados en el diseño del terreno.

Para la creación del césped se selecciona el terreno, y se da click en pain details, después en edit details y Add Grass texture Saldrá un cuadro emergente el cual permite seleccionar la textura del pasto, la cual ya se encuentra guardada en el Asset Nature que se importó

previamente. Se hicieron dos texturas de pasto y una de flores a la cual se le varía los colores. Para implementar estas texturas de pasto en el terreno, tenemos herramientas de pincel, así como las que se utilizaron para hacer el camino, además de establecer que tan grande es el área de implementación del pasto y que tan saturado de esta textura se desea.

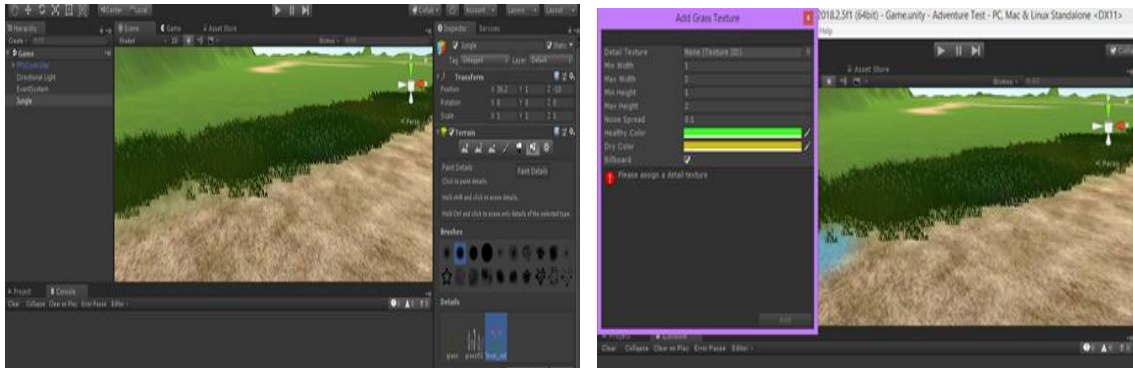


Ilustración 113. Adición de más texturas.

Para agregar los árboles, los cuales también se encuentran en el Asset Nature, se selecciona el terreno y se da click en la opción Paint tres, seleccionando después la opción edit trees, add tree. Y de la misma manera que se colocó el césped, se procede a implementar los árboles, para así obtener este resultado grafico en 3d.



Ilustración 124. Inserción de los árboles para el terreno.

Para evitar que el personaje se salga del camino, se toma un objeto cualquiera, que en este caso es una caja, se coloca por todo el camino para que el personaje colisione y no pueda salir del camino. Se seleccionan todos los objetos creados, para poder deshabilitar la opción de mesh Render, la cual permite hacer los objetos visibles o no, se deshabilita esta opción, para que los objetos no sean visibles, pero que aun así se pueda colisionar con ellos, lo que permite ver al personaje su alrededor, pero que no se le permita salir del camino.

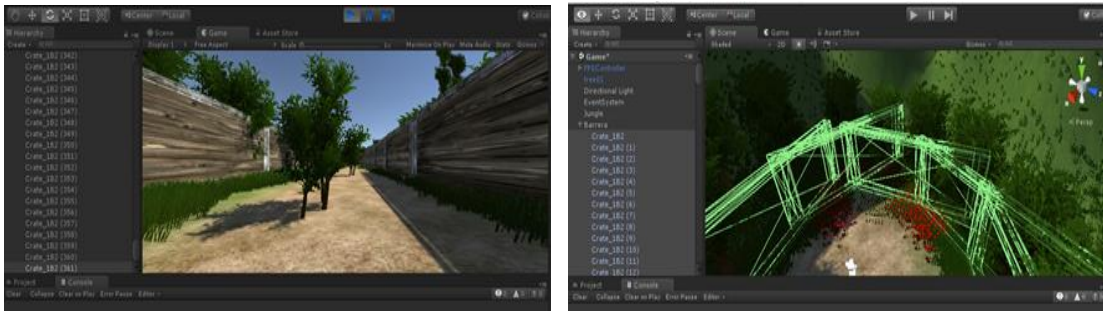


Ilustración 135. Incorporación de obstáculos al terreno.

Para los obstáculos que se van a presentar en el camino y poner un poco más de ambientación en la pista para que el juego sea más atractivo y dinámico, se utilizaron 4 Asset los cuales son:

Ultimate fantasy creator lite, el cual cuenta con prefabs de madera, alusivos a campamentos.

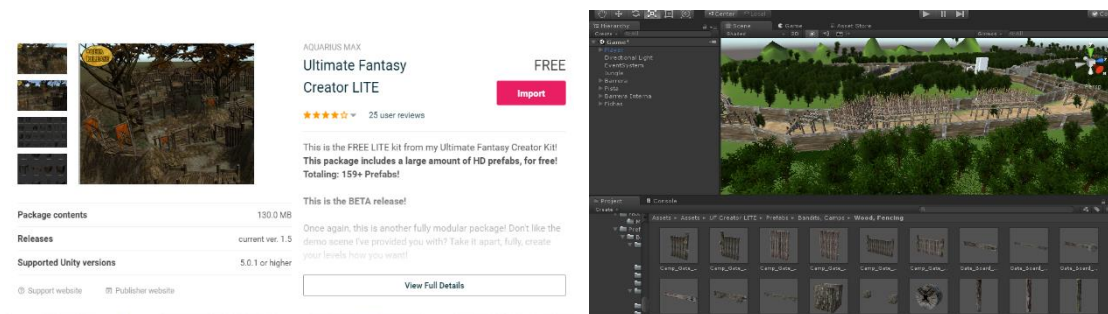


Ilustración 146. Utilización de más Asset.

Rocks free pack el cual brinda una variedad de rocas, pero toca agregarles el componente de mesh collider, ya que, si solo se colocan así, estas rocas se pueden atravesar.

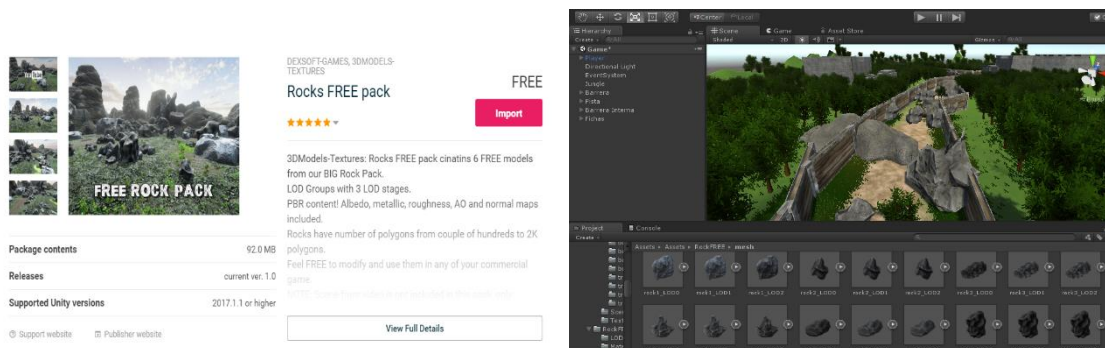


Ilustración 157. Uso de Asset Rocks Free Pack.

Block creator lite el cual ofrece prefabs de bloques para poder realizar estructuras

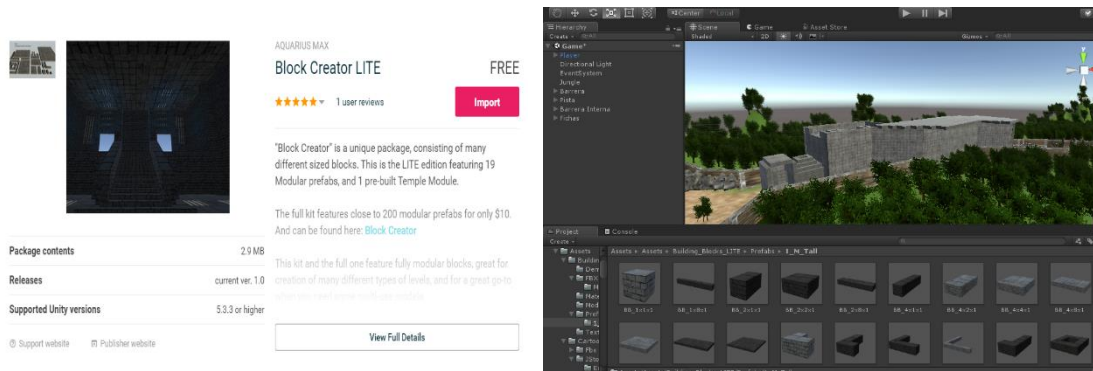


Ilustración 168. Uso del Asset Block Creator Lite.

Cartoon temple building kit lite, a estos prefabs también toca agregarles el componente de mesh collider, para que exista una colisión entre el personaje y los objetos.

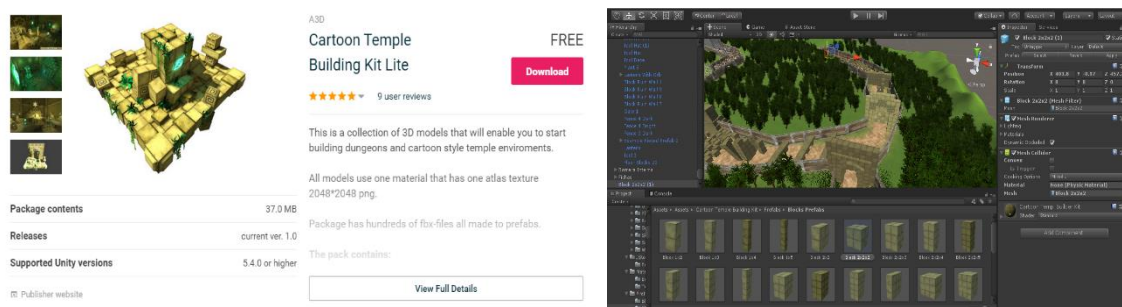


Ilustración 179. Uso del Asset Cartoon Temple Building Kit Lite.

Para la creación del menú de pausa, se crea un nuevo canvas o lienzo, se selecciona la opción de escena, en la opción 2d luego se presiona la letra 'f' para tener visión del canvas en el cual se van a colocar los elementos de la interfaz de usuario.

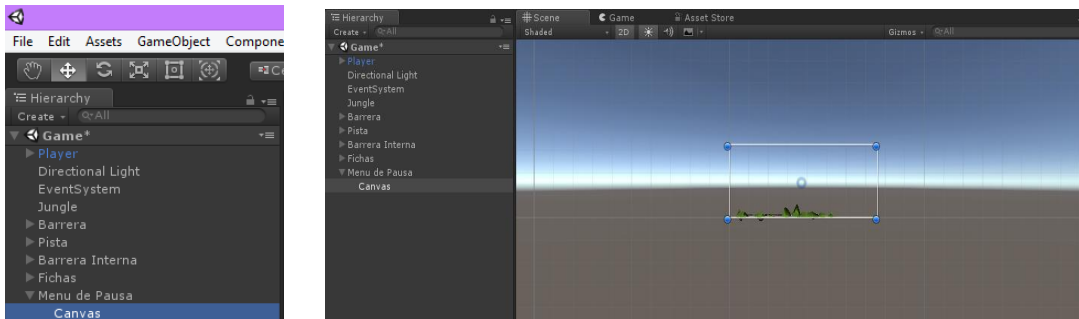


Ilustración 60. Creación del menú de pausa.

Enseguida se crea un panel y se le da un tono negro transparente, para así poder dar alusión de que el juego se encuentra en un estado de pausa.

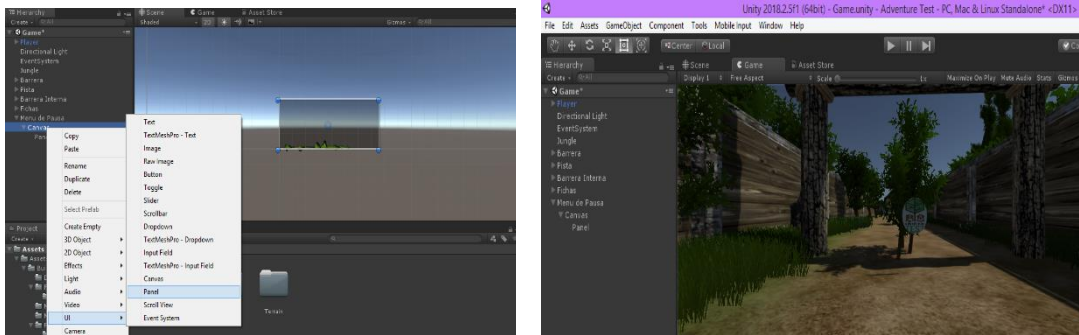


Ilustración 61. Opción de pausa.

Se agrega un botón dentro del panel 'Pause Menú'; Se selecciona el botón, se deshabilita la opción de Image (Script), Luego se despliega el botón, se selecciona el texto, se cambia el estilo de letra, el tamaño, el color a blanco y se reescribe el texto por reanudar. Se añade el componente de sombra y finalmente obtener como resultado grafico la reanudación del juego.

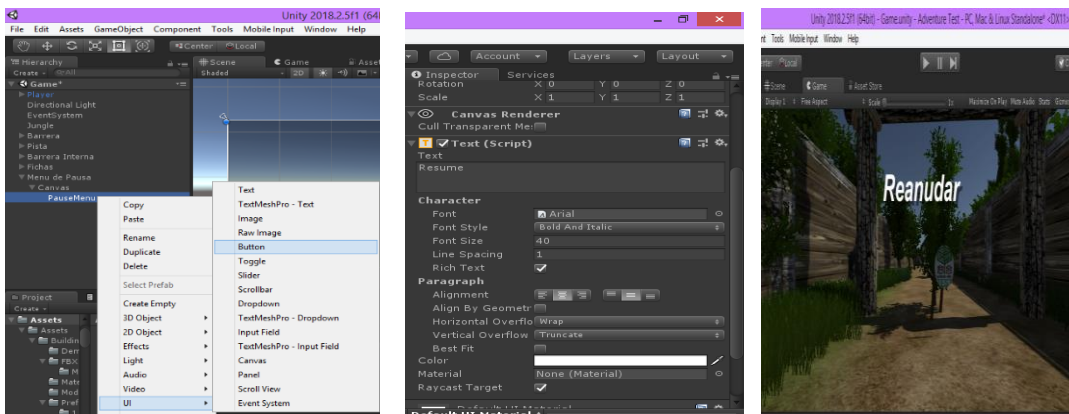


Ilustración 62. Reanudación del juego.

LOGIN Y BD

Para crear el menú de login se debe crear un canvas, se procede a colocar los input Field los cuales son campos de texto, que en este caso permitirán recolectar la información de los jugadores

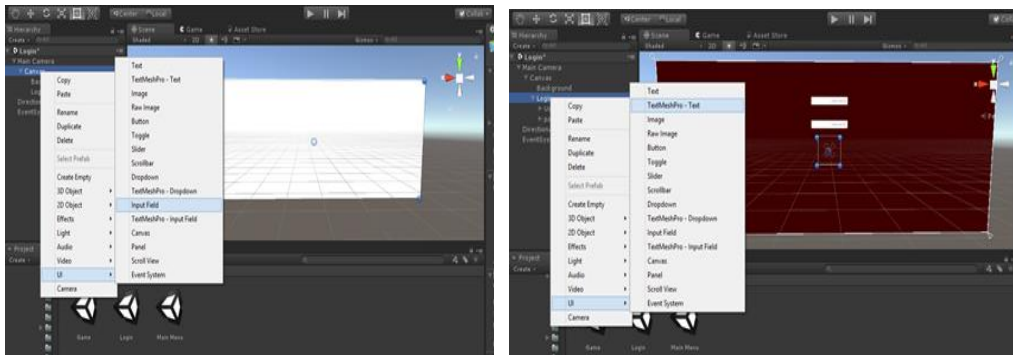


Ilustración 63. Creación del Login.

Se implementa el label, para referenciar cual es la casilla de Nombre del usuario y cuál es la casilla para ingresar su contraseña, una vez terminados los campos para el ingreso de la información y colocados los label que delimitan para que es cada campo, se procede a implementar los botones.

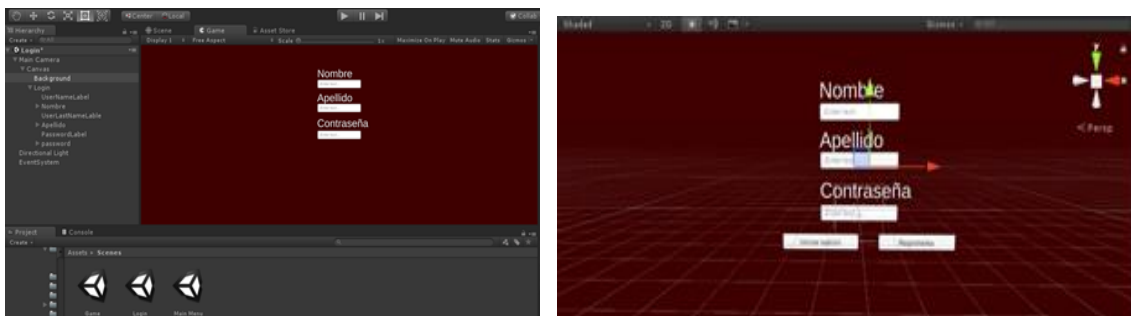


Ilustración 184. Diseño de los botones para el Login.

Una vez esté listo el menú del login, se procede a crear el menú de registro, para lo cual se desactiva temporalmente el de login; creamos la base de datos, con los campos requeridos para poder almacenar la información necesaria de cada jugador.

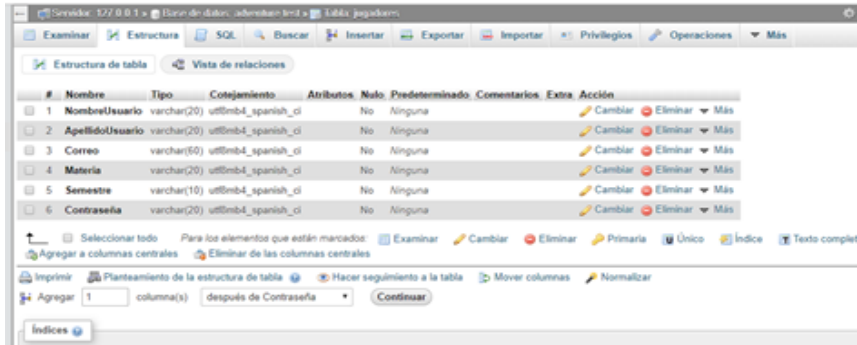


Ilustración 195. Base de datos del juego.

Preguntas

Para la creación de las preguntas, se crea un canvas, del mismo modo de como se creó para la implementación del menú de pausa. Después se coloca un texto en la parte superior izquierda, este texto va a ser usado en el script 'moneda', para que muestre la puntuación actual del jugador; se coloca el título y la pregunta con la creación de textos, se crean los botones, los cuales, van a tener dos funciones gracias al script 'moneda', los cuales son correcto o incorrecto.

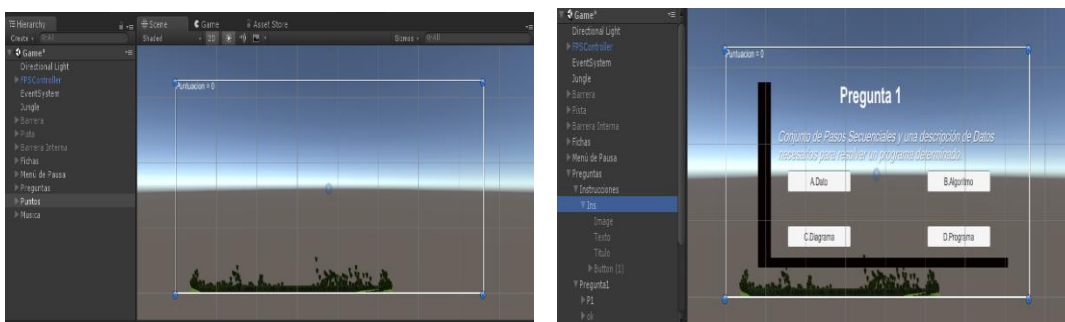


Ilustración 206. Creación de las preguntas.

Si la respuesta es correcta, el puntaje del texto aumentará en 5 puntos y aparecerá este botón, que dice correcto el cual permitirá salir de la pregunta y continuar con el juego, si por el contrario es incorrecta, el contador de puntos no aumentará, además de esto aparecerá un texto informando al jugador cuál era la respuesta correcta y aparecerá un botón informando que la respuesta fue incorrecta y al igual que el botón de correcto, este al pulsarse reanudará el juego.

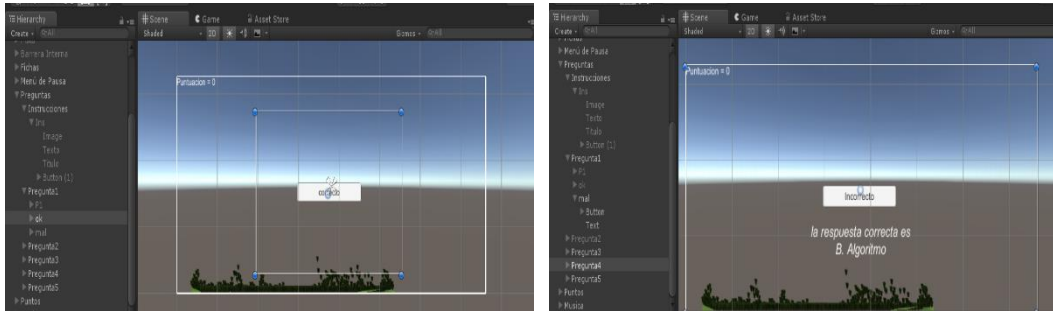


Ilustración 217. Botones con la respuesta.

Para las preguntas que aparecen en el juego se clasificaron de la siguiente manera:

Conceptos básicos

1. Conjunto de Pasos Secuenciales y una descripción de Datos necesarios para resolver un programa determinado.

Rta/ Algoritmo

2. Cifra, letra o palabra que se suministra al programa como entrada, el cual puede ser almacenado en un determinado formato.

Rta/ Dato

3. ¿Los Algoritmos se pueden representar de forma gráfica?

Rta/ Si. Se llaman diagramas de flujo

4. Corresponde a un área reservada en la memoria, a la cual se le puede cambiar su contenido a lo largo de la ejecución de un programa.

Rta/ Variable

5. Es un valor que no puede ser alterado o modificado durante la ejecución de un programa.

Rta/ Constante

6. Es una aplicación o programa que traduce un programa fuente en un programa objeto que puede ser interpretado y ejecutado por el computador.

Rta/ Compilador

7. Es la representación gráfica de un algoritmo o pseudocódigo mediante símbolos convencionales que muestran las acciones, procesos y flujo de la información para resolver un problema planteado.

Rta/ diagrama de flujo

8. ¿Todo algoritmo debe tener un fin?

Rta/ Si

9. Es un conjunto de palabras, letras, números y símbolos especiales con reglas sintácticas definidas que se utilizan para dar instrucciones al computador.

Rta/ Lenguaje de Programación

10. ¿Los operadores lógicos son representados con signos en el lenguaje de programación?

Rta/Si

Operadores Aritméticos

11. ¿Qué función cumple el código?

```
int numero1 = 0;
int numero2 = 0;

System.out.println("Introduce el primer número:");
numero1 = reader.nextInt();

System.out.println("Introduce el segundo número:");
numero2 = reader.nextInt();
resultado = numero1+numero2;

System.out.println("La suma es " + numero1 + " + " + numero2 + " = " + resultado);
```

Rta/ Suma

12. ¿Qué función cumple el código?

```
int numero1 = 0;
int numero2 = 0;

System.out.println("Introduce el primer número:");
numero1 = reader.nextInt();

System.out.println("Introduce el segundo número:");
numero2 = reader.nextInt();
resultado = numero1-numero2;

System.out.println("La resta es " + numero1 + " - " + numero2 + " = " + resultado);
```

Rta/ Resta

13. ¿Qué función cumple el código?

```
numero1 = Integer.parseInt(br.readLine());
numero2 = Integer.parseInt(br.readLine());

try{
    System.out.println("Introduce el primer número:");
    numero1 = Integer.parseInt(br.readLine());
    System.out.println("Introduce el segundo número:");
    numero2 = Integer.parseInt(br.readLine());
} catch (IOException ioe){
    ioe.printStackTrace();
}
resultado = numero1*numero2;
System.out.println("La multiplicación es " + numero1 + " x " + numero2 + " = " + resultado);
```

Rta/ Multiplicación

14. ¿Qué función cumple el código?

```
1. float numero1 = 0;
2. float numero2 = 0;
3. float resultado;

Scanner reader = new Scanner(System.in);
System.out.println("Introduce el primer número:");
numero1 = reader.nextFloat();

System.out.println("Introduce el segundo número:");
numero2 = reader.nextFloat();
resultado = numero1/numero2;
System.out.println("La división es " + numero1 + " / " + numero2 + " = " + resultado);
```

Rta/ División

15. La siguiente línea de código es para el operador

System.out.println(10 % 2);

Rta/ mod

Operadores de Comparación

16. Símbolo del operador lógico 'Es Igual':

Rta/ ==

17. Símbolo del operador lógico 'Es distinto':

Rta/ !=

18. Símbolo del operador lógico 'Menor':

Rta/ <

19. Símbolo del operador lógico 'Mayor':

Rta/ >

20. Símbolo del operador lógico ‘Menor o igual’:

Rta/ <=

21. Símbolo del operador lógico ‘Mayor o igual’:

Rta/ >=

Operadores Lógicos

22. Símbolo del operador lógico ‘And (Y)’

Rta/ &&

23. Símbolo del operador lógico ‘or (O)’

Rta/ ||

24. Símbolo del operador lógico ‘Not (No)’

Rta/ !

Variables

25. Son datos que no necesitan métodos o invocaciones para ser credos

Rta/ datos primitivos

26. Son datos que necesitan métodos o invocaciones para ser credos

Rta/ Datos Objeto

27. La variable ‘byte’ recibe datos de tipo:

Rta/ Entero

28. La variable ‘short’ recibe datos de tipo:

Rta/ Entero

29. La variable ‘Int’ recibe datos de tipo:

Rta/ Entero

30. La variable ‘Long’ recibe datos de tipo:

Rta/ Entero

31. Si las variables Int, byte, short y long aceptan valores enteros, ¿que los diferencia?

Rta/ Capacidad de almacenamiento

32. La variable ‘Float’ recibe datos de tipo:

Rta/ Decimal Simple

33. La variable ‘Float’ recibe datos de tipo:

Rta/ Decimal Doble

34. La variable 'Char' recibe datos de tipo:

Rta/ Carácter simple

35. ¿Si se quiere recibir un valor que sea verdadero o falso, que variable se utiliza?

Rta/ Boolean

36. Serie de elementos o formación tipo vector o matriz. Lo consideraremos un objeto especial que carece de métodos.

Rta/ Array

Clases predeterminadas

37. Plantilla que sirve para crear un objeto.

Rta/ Clase

38. Clase que manejar cadenas de caracteres de cualquier longitud finita

Rta/ Clase String

39. public final class Math {
 public static final double E = 2.7182818284590452354;
 public static final double PI = 3.14159265358979323846;
 } //Teniendo en cuenta el código, la clase math acepta valores de tipo:

Rta/ doble

Arreglos

40. Es una estructura de datos que nos permite almacenar un conjunto de datos de un mismo tipo.

Rta/ Array

41. ¿Cuántos elementos posee este array? int[] edad = {45, 23, 11, 9};

Rta/ 4

42. double[] estatura = {1.73, 1.67, 1.56};

43. ¿Este array está escrito de manera correcta? String[] nombre = {María, Gerson};

Rta/ No, los valores de tipo string deben estar encerrados entre comillas

String[] nombre = {"María", "Gerson"};

44. ¿Este array está escrito de manera correcta? char[] sexo = {'m', 'f'};

Rta/ Si, los caracteres a diferencia del tipo string van encerrados entre comilla sencilla

Ciclos

```
45. f (inicialización ; condición ; acción ){
    bloque de acciones;
} //¿este ciclo permite evaluar condiciones?
```

Rta/ si, El Ciclo for permite evaluar la condición, si esta es verdadera entra al ciclo y ejecuta el o los estatutos

46. Que ciclo permite verifica la condición, mientras sea verdadera se sigue ejecutando; al momento de ser falsa termina el ciclo.

Rta/ Do/While

```
47. public void contarHasta(int numero) {
    System.out.println("Cuenta hasta: " + numero);
    for (int i = 1; i <= numero; i++) {
        System.out.println(i)
    }
} //¿Este ciclo for se va a ejecutar de manera correcta?
```

Rta/ No, la línea 4 no está cerrándose con ‘;’

```
48. public void numeroPar(int numero) {
    System.out.println("Numero par menor que: " + numero);
    int n;
    do {
        n = (int) (Math.random() * numero);
        System.out.println(n);
    } while (n % 2 != 0);
    System.out.println("Y el numero par elegido es: " + n);
} //¿Este ciclo do/while se va a ejecutar de manera correcta?
```

Rta/ Si

```
49. public int leerNumero() {
    Scanner sc = new Scanner(System.in);
    int numero = -1;
```



```
while (numero <= 0) {  
    System.out.println("Introduce un numero positivo: ");  
    numero = sc.nextInt();  
}  
sc.close();  
return numero;  
} //¿Este ciclo while se va a ejecutar de manera correcta?
```

Rta/ No, en la línea 9 se está cerrando con ':' por lo tanto el retorno no se va a efectuar y el código no va a correr

ANEXO C

Artículo Científico

Refuerzo de conocimientos por medio de los videojuegos

Andrés Felipe Cano Rodríguez, Leidy Vanessa Sánchez Núñez. Universidad de Cundinamarca, Extensión Chía, Facultad de Ingeniería de sistemas.

I. RESUMEN

En este artículo se exponen las cualidades y beneficios que representan la implementación de un videojuego como nueva metodología de aprendizaje, que en este caso fue enfocado a los temas de la algoritmia y la lógica para programar. Este por medio de una encuesta que se realizó a los estudiantes de lógica de programación y programación I de la Universidad de Cundinamarca, extensión Chía, en la cual se buscó analizar el nivel de aceptación de la implementación de una nueva metodología por parte de los estudiantes e indagar cuales son los problemas más frecuentes que se presentan en el proceso de aprendizaje de estos temas. Con lo cual se analizó qué tipo de características debe poseer este videojuego para reducir estos problemas y generar un mayor aporte al proceso de aprendizaje, por parte de los estudiantes en esta área del conocimiento.

II. PALABRAS CLAVE

Videojuego, Aprendizaje, Algoritmia, Lógica para programar, Metodología.

III. ABSTRACT

In this article we expose the qualities and benefits that represent the implementation of a video game as a new learning methodology, which in this case was focused on the topics of algorithm and logic to program. This was done through a survey that was carried out to the students of programming logic and programming of the University of Cundinamarca, extension Chía, in which it was sought to analyze the level of acceptance of the implementation of a new methodology by the students and to investigate which are the most frequent problems that arise in the learning process of these topics. With which it was analyzed what type of characteristics must have this video game to reduce these problems and generate a greater contribution to the learning process, on the part of the students in this area of knowledge.

IV. KEYWORDS

Videogame, Learning, Algorithmic, Logic to program, Methodology.

V. INTRODUCCIÓN

El aprendizaje es un proceso en el cual se adquieren habilidades, conocimientos, actitudes y

valores, que se desarrollan por medio de la enseñanza o la experiencia [1]. Este proceso de enseñanza es empleado por los profesores, por medio de unas metodologías, las cuales son un conjunto de procesos que permiten influir en la percepción del estudiante, para poder adquirir conocimientos nuevos [2]. Existen ocasiones donde los estudiantes por diferentes aspectos que van desde la falta de compromiso con el núcleo académico, en algunos casos por una mala interpretación de metodologías que hacen de este tipo de materias una experiencia irrelevante que no causa ningún efecto positivo al momento de ponerla en práctica, se planteó el desarrollo de un videojuego que ayude a reforzar los temas vistos en este tipo de materias. Teniendo en cuenta este proceso y los posibles problemas que se pueden presentar, se realiza una encuesta a los estudiantes de lógica de programación y programación I de la Universidad de Cundinamarca, extensión Chía, con el fin de analizar si, en primer lugar estos temas les resultan de interés, buscar si hay posibles problemas, como falta de bases en estos temas, y contemplar que tan aceptada y viable sería la implementación de una nueva herramienta metodológica que les resulte más llamativa, para que puedan adquirir un mejor nivel de conocimiento con respecto a estos temas, que esta sea consistente y permita mitigar los problemas que se presentan en este proceso de aprendizaje.

VI. MÉTODOS Y MATERIALES

Esta nueva metodología trata de la implementación de un videojuego educativo con el propósito mejorar el proceso de aprendizaje sobre conceptos básicos de programación y algoritmia. El videojuego fue elaborado en Unity, donde se recreó un escenario de estrategia aventurera, en el que el

jugador tuvo que aprobar una serie de obstáculos, que a su vez estaban acompañados de una serie de preguntas dentro de las cuales se clasifican desde el manejo de conceptos básicos, variables, operadores lógicos hasta sentencias. Dentro del juego a medida que el participante responde acertadamente aumenta una puntuación asignada para que él pueda evidenciar que tan bueno es en este tipo de materias y que debe reforzar.



Ilustración 228. Diseño de la estructura del juego.

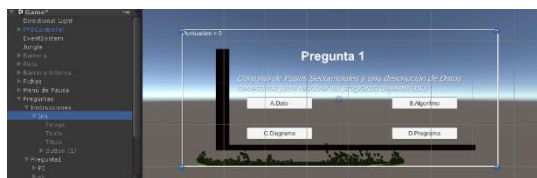


Ilustración 239. Preguntas a desarrollar para aumentar el puntaje.

Se tomó como población a los estudiantes de ingeniería de sistemas de la Universidad de Cundinamarca extensión Chía, a los cuales se les realizó una encuesta para contemplar cuales son los problemas que se presentan en el proceso de aprendizaje de la algoritmia y la lógica para programar. Esto se planteó en base a que de los videojuegos se pueden obtener varios beneficios tales como;

Al jugar se genera un nuevo ambiente, un ambiente más didáctico y colectivo que permite al estudiante estar en confort y desarrollarse de manera más creativa, olvidándose de los prejuicios y las dudas, potenciando el desarrollo de sus capacidades y fijando de mejor forma los

conocimientos. Haciéndolo ver de manera más objetiva y generando un pensamiento más lógico para la solución de problemas [3].

Se tomó en cuenta un caso que ocurrió en el 2017, el cual fue desarrollado por la doctora Sonia Pamplona Roche, profesora del Grado en Ingeniería Informática en la Universidad a Distancia de Madrid (UDIMA), la cual presentó este trabajo académico durante su participación en el IV Congreso Internacional sobre Aprendizaje, Innovación y Competitividad (CINAIC 2017), que tuvo lugar hasta este viernes en la sede de Zaragoza. En donde se analizó la experiencia de la gamificación en el proceso de aprendizaje de los estudiantes. Los resultados mostraron tres efectos principales: “diversión, motivación y mejora del aprendizaje”, así lo explicó a UDIMA Media la doctora Pamplona Roche.

VII. RESULTADOS

De la encuesta que se realizó a los estudiantes de lógica de programación y programación I de la Universidad de Cundinamarca, extensión Chía, se tomaron los datos más relevantes para plantear como sería el desarrollo del juego. Fue contestada por 27 estudiantes arrojando la siguiente información:

2 ¿Usted considera que se puede adquirir conocimientos a través de los videojuegos?

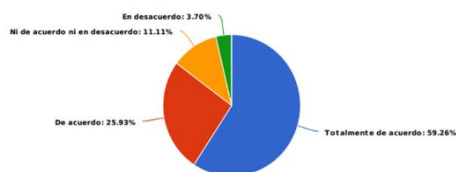


Fig.2 En la cual se buscó obtener información por parte del estudiante, de si este puede adquirir conocimiento por medio de los videojuegos. A lo cual más del 70% de los estudiantes encuestados afirmaron estar de acuerdo, lo cual afirmó las

teorías planteadas previamente, que se puede obtener conocimientos por medio de los videojuegos.

5 ¿Considera que los videojuegos son una buena estrategia para obtener una mayor participación por parte de los estudiantes en la programación?

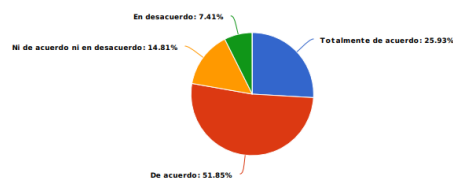


Fig.5 En esta encuesta se analizó la motivación que recibirían los estudiantes para su proceso de aprendizaje por medio de la implementación de un videojuego.

8 ¿Considera que de esta manera se pueda reducir el grado de dificultad en la lógica de programación?

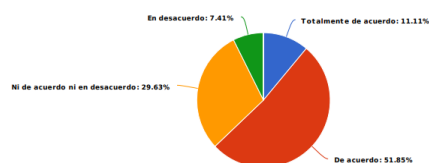


Fig.8 Se analizó que implementar esta metodología podría minimizar el grado de dificultad para el proceso de aprendizaje de este tema.

Por ende gracias a la encuesta que se aplicó, se pudo concluir que si se presentan problemas en el proceso de aprendizaje y las bases establecidas en este tema, que los estudiantes requerían de la implementación de una nueva metodología de enseñanza, la cual se planteó como un videojuego educativo para reforzar el aprendizaje y mitigar o resolver los problemas que se presentaron.

VIII. DISCUSIÓN

De acuerdo con los resultados obtenidos podemos evidenciar que el desarrollo de un videojuego como estrategia metodológica, ayuda a mitigar los problemas al momento del aprendizaje en la materia, reforzando la pedagogía implementada por los docentes. Captando un mayor interés por parte de los estudiantes, haciendo de este método un proceso dinámico e interesante de aplicar.

Partiendo de la idea de adoptar esta estrategia en los estudiantes de los primeros semestres de la carrera ingeniería de sistemas en la Universidad de Cundinamarca extensión chía, podemos comparar el grado de interés y participación en las materias de programación y algoritmia con respecto a metodologías anteriormente impuestas por los docentes como proceso de formación, para verificar la efectividad de esta nueva herramienta. Esto contando previamente con la aprobación y aceptación por parte del cuerpo docente y directivo. Una vez hecho efectivo este proceso evaluaremos el tipo de mejoras pertinentes o la adición de módulos de ser requerido.

IX. CONCLUSIONES

- Se comprobó que había problemas de aprendizaje con respecto a la algoritmia y la programación por parte de los estudiantes.
- Se evidencio que es necesario incorporar nuevas herramientas metodológicas que ayuden a reforzar el tema del aprendizaje.
- Se concluyó que por medio de los videojuegos los estudiantes, puedan adquirir un mayor interés por la materia y de esta manera mejorar su rendimiento ampliando su conocimiento.

X. RECONOCIMIENTOS

Agradecemos a los docentes Misael Perilla y Jairo Márquez por su colaboración y guía en el desarrollo de este proyecto. Haciendo posible llevar a cabo un trabajo muy interesante sobre cómo se puede ayudar a mejorar u reforzar la

metodología de aprendizaje en los estudiantes y docentes.

XI. REFERENCIAS BIBLIOGRÁFICAS

- [1] <https://definicion.de/aprendizaje/>
 [2] <https://es.slideshare.net/GerardoPerez33/metodologa-de-la-enseanza-69508399>
 [3] <http://noticias.universia.es/educacion/noticia/2016/09/09/1143448/beneficios-juego-educacion-universitaria.html>
