

Diseño e Implementación de Modelo de Enseñanza de Ingeniería de Software con Base en Videojuegos Educativos

Gustavo A. Atuesta^{*}, Daniel Arenas^{**}

Fecha de recibido: : 20/09/2015 Fecha de aprobación: 28/10/2015

Resumen

Este trabajo presenta la creación de un modelo de educación el cual tiene como base principal la utilización de videojuegos como herramientas de aprendizaje dentro del contexto pedagógico universitario. Dicho modelo se implementa a modo de prueba dentro de una clase de Ingeniería de Software, explicando los conceptos generales dentro del tema de requerimientos de software, abarcando la definición, características, recolección y recopilación permitiendo de esta forma realizar una puesta en producción dentro de un entorno similar al de una clase real.

Para poder ejecutar dicha implementación del modelo, se realiza el desarrollo de un videojuego educativo el cual cuenta con cuatro clases, cada una de ellas creada con base en las nociones académicas de uso diario en la clase de ingeniería de software utilizadas por docentes de la materia, permitiendo de esta forma presentar un contenido similar a lo utilizado en clases reales. Como segunda medida, se implementa una estructura de teoría, seguida de una sección práctica interactiva, con el fin de permitir al estudiante poner a prueba el conocimiento adquirido de manera rápida.

Finalmente, el videojuego se pone a prueba con estudiantes de clases de programación e ingeniería de software, permitiendo de esta forma obtener un proceso de retroalimentación y de esta forma concluir la ejecución del modelo con la identificación de factores de mejora para una futura iteración.

Palabras clave: *E-learning, Juegos educativos, Clases virtuales, Educación virtual, Ingeniería de software, Videojuegos.*

^{*}Universidad Autónoma de Bucaramanga, Maestría en Gestión Aplicación y Desarrollo del Software. Avenida 42 No. 48 – 11, Bucaramanga - Colombia. E-mail: gustavoandres.ac@gmail.com.

^{**}Universidad Autónoma de Bucaramanga, Facultad de sistemas. Avenida 42 No. 48 – 11, Bucaramanga - Colombia. E-mail: darenass@unab.edu.co.

[‡] Se concede autorización para copiar gratuitamente parte o todo el material publicado en la *Revista Colombiana de Computación* siempre y cuando las copias no sean usadas para fines comerciales, y que se especifique que la copia se realiza con el consentimiento de la *Revista Colombiana de Computación*.

Abstract

This work presents the development of an educational model which is based mainly on the use of video games as learning tools within the educational context of the University. This model is implemented within a class of Software Engineering as a way to test it, explaining the general concepts of the software requirements topics like the definition, characteristics, collection and compilation thus allowing a live test inside a real class like environment.

In order to correctly implement this model, an educational game is created. This game has four classes, each one constructed in accordance with academic notions of daily use in software engineering class implemented by teachers of the subject, thus allowing it to display content in a similar fashion of what is used in actual classes. As a second step, a structure of theory, followed by an interactive practice section is implemented, in order to allow the student to test the acquired knowledge quickly.

Finally, the game is tested with students of programming and software engineering classes, in order to allow us to generate a feedback process and thus conclude the implementation of the model by identifying improvement factors for a future iteration.

Keywords: *Educative games, E-learning, Software engineering, Videogames, Virtual clases, Virtual education.*

1. Introducción

Actualmente se vive en una época que está corriendo a pasos agigantados hacia la integración total de la tecnología con las actividades cotidianas. Siendo ejemplo de esto la forma en que las principales ciudades del mundo cuentan con conexiones de internet activas las 24 horas del día, los 7 días de la semana permitiendo a sus habitantes enterados de toda clase de información de su interés en tiempo real sin importar el lugar en que ocurran.

Este campo de acción tecnológico ha comenzado a influenciar la forma en que las juventudes actuales aprenden, motivo por el cual, muchas instituciones han iniciado la búsqueda de nuevas formas de enseñar, teniendo como enfoque central el poder adaptarse a las nuevas necesidades de los estudiantes. Actividades como el cambio de libros físicos a versiones digitales o la creación de páginas web académicas con servicios en línea es un ejemplo de esto; sin embargo, es necesario asegurar la manera de ofrecer un valor agregado a estas actividades para poder sacar un verdadero beneficio a esta digitalización del conocimiento [1].

Reforzando esta idea, se encuentran diferentes estudios académicos aplicados dentro de aulas de clase, como lo es el uso de e-blocks

(herramientas informáticas diseñadas para propiciar el aprendizaje colaborativo) en salones de preescolar, el cual fue realizado por el Centro de Tecnología y Docencia de la Universidad de Concepción, Chile. En donde se evaluó el uso de herramientas digitales didácticas en aulas de preescolar. También se encuentra el análisis del manejo de dispositivos móviles en las aulas universitarias como alternativas pedagógicas complementarias en la educación superior [2]. Por su parte, la Facultad de Matemáticas de la Universidad Autónoma de Yucatán, México presenta otro estudio relacionado, en el cual se pudo observar el potencial del uso de tecnologías actuales como herramientas de educación y de manera similar, se evidenció el factor de resistencia al cambio tanto en estudiantes como docentes, haciendo de esto un factor importante a tener en cuenta en los desarrollos futuros [3].

En ambos casos se evidencia un aumento en la motivación de los estudiantes, especialmente cuando se involucran elementos dinámicos o multimedia, sin embargo, este nuevo proceso de enseñanza trae consigo incertidumbres, pues rompe el paradigma habitual de enseñanza tradicional, sin embargo, permite abrir la puerta al potencial uso de tecnologías como los videojuegos como alternativa académica conformada no solo por elementos multimedia sino por formar parte de un grupo de un nuevo mercado tecnológico en expansión.

Este artículo presenta a continuación un breve análisis de algunas herramientas existentes creadas para la enseñanza dentro del entorno de la ingeniería de sistemas, informática y afines, seguido de algunas recomendaciones a tener en cuenta al momento de planear la enseñanza de una clase dentro de esta área temática. Una vez finalizada esta sección de estado del arte, se presenta el modelo creado y la estructura general del videojuego diseñado como herramienta de validación del modelo, permitiendo finalizar con la presentación de los resultados obtenidos durante su puesta en producción con un pequeño grupo de estudiantes de clases de ingeniería de sistemas e informática y llegando así a la sección de conclusiones y aporte científico del mismo.

2. Videojuegos como Herramientas de Aprendizaje

A. Beneficios de los videojuegos en la educación

Prensky menciona la diferencia generacional entre las personas que ayudaron a crear el camino para la expansión de la tecnología llamados “inmigrantes digitales” y aquellos que nacieron durante el crecimiento tecnológico llamados “nativos digitales”, siendo estos últimos una generación habituada a la tecnología a un punto tal que dan por hecho su existencia [4]. Es por eso que los “Los estudiantes cambiaron la manera fundamental de aprender con el simple hecho de crecer rodeados con

esta tecnología”. Esta nueva generación busca un papel más activo al momento de aprender [5] y es aquí donde los videojuegos encuentran la manera de incorporarse en los salones de clase de una manera natural debido a su naturaleza interactiva.

Dentro este orden de ideas, se encuentra Coopman de la Universidad de Carroll, Wisconsin, Estados Unidos, quien menciona que “La gente aprende mejor cuando se sienten entretenidos, cuando pueden usar la creatividad para alcanzar metas complejas, cuando las clases incorporan tanto el pensamiento como las emociones, y cuando las consecuencias de las acciones pueden ser observadas.” [6], Reforzando una vez más la viabilidad del uso de videojuegos pues estos como su nombre lo implica están creados pensando en entretener, sin embargo, el tema de videojuegos educativos no representa un concepto nuevo hablando en términos generales, basta con realizar una búsqueda en internet para encontrar diferentes ejemplos en algunos temas dentro de la ingeniería de sistemas como se puede ver a continuación.

B. Trabajos existentes

Estos son algunos de los exponentes más significativos identificados como videojuegos existentes que permiten su incorporación dentro de un entorno universitario dentro de la ingeniería de sistemas e informática:

- Code Combat [7]: Cubre temáticas como sintaxis de programación, manejo de métodos, parámetros, uso de cadenas, ciclos y variables entre otros.
- Beta the game [8]: Este videojuego funciona más como un motor de videojuegos simplificado, con su código personalizado llamado “codePOP” a través del cual los usuarios podrán modificar los elementos presentados por defecto en el juego y crear su propia versión del mismo sin necesidad de compilar.
- Hack and slash the game [9]: Esta entrega permite a los jugadores reprogramar las propiedades de los objetos del juego, cambiar variables e incluso modificar el código del juego, siendo esta la única forma de finalizarlo.
- Dr Who code game [10]: La BBC de Londres se ha caracterizado por ofrecer contenido no solo de entretenimiento, sino también educativo y este es el caso del juego de desarrollo inspirado en la serie de culto Dr Who, el cual está enfocado a un público infantil. El objetivo de este videojuego es resolver una serie de misiones relacionadas con la programación, lo cual puede lograrse a través del uso de una línea de comandos para poder cambiar el entorno en que se juega.

Los anteriores juegos representan contenido enfocado exclusivamente en la enseñanza de programación, pero, se identificaron los siguientes trabajos enfocados en la ingeniería de software como elementos importantes a tener en cuenta como herramientas de aprendizaje en este campo temático:

- SESAM [11]: busca facilitar la adquisición de experiencia similar a la laboral en los estudiantes durante el proceso de aprendizaje universitario convirtiéndolo en una herramienta de soporte tanto para los estudiantes como los docentes dentro de la academia.
- SIMSE [12]: Videojuego simulador el cual presenta un escenario simulado de un desarrollo de software; aquí, el estudiante toma el lugar del director de un proyecto y deberá superar exitosamente los diferentes obstáculos del escenario.

Estos últimos dos videojuegos presentan alternativas interesantes en el tema de mayor interés de este trabajo, sin embargo, estos desarrollos se enfocan en presentar un elemento postclase, es decir, buscan permitir al estudiante reforzar y practicar los conocimientos adquiridos en la clase en un entorno que se asemeje de alguna manera al mundo real, pero no permiten al estudiante adquirir el conocimiento relacionado con la ingeniería de software antes o durante una clase. Con base en lo anterior, se da inicio a la ejecución del proyecto en cuestión para presentar un modelo de ejecución de clases que permita utilizar videojuegos como herramientas de apoyo en la enseñanza.

No obstante, a pesar de haber encontrado ejemplos de videojuegos educativos en el campo, estos no dicen lo suficiente por su cuenta respecto a los temas que se deben tener en cuenta al momento de desarrollar contenidos virtuales de aprendizaje, es por este motivo que la investigación debe comenzar a identificar y analizar aquellos elementos que ayuden con la creación de materiales pedagógicos digitales y la forma en que estos deben aplicarse.

C. Recomendaciones pedagógicas

Para la elaboración del modelo, se tomaron en cuenta algunos puntos propuestos para la ejecución de una correcta clase de programación, destacando así aquellas propuestas por Meyer [6] quien menciona:

- Evitar caer en la trampa de explicar temas basados en modas y no en necesidades del mercado actual.
- Planear clases de tal manera que todos los estudiantes, sin importar su nivel de conocimiento puedan ir a un ritmo similar.

- Asegurar que aquellos estudiantes con amplio conocimiento tengan alternativas que motiven su interés en la ejecución de las mismas y no sientan que no están aprendiendo nada nuevo.
- Seguir estándares regionales e internacionales para garantizar que los estudiantes puedan aplicar de manera inmediata sus conocimientos en el mercado.

Acompañando a estas recomendaciones, se encuentran aquellas mencionados por Isong Bassey [13] quien recomienda:

- Las aulas de clase deben estar bien equipadas para la ejecución de los programas académicos de desarrollo de software.
- Es necesario que todas las clases de desarrollo de software se realicen en aulas de cómputo y tengan algún elemento práctico.
- Los estudiantes deben recibir la documentación de los temas académicos antes de la ejecución de las clases.
- El docente debe mantener contacto constante con los estudiantes a modo de poder brindar la retroalimentación necesaria para mejorar el desempeño académico de los mismos.

Teniendo en cuenta estas recomendaciones y las diferentes teorías y ejemplos prácticos identificados se procede a presentar el modelo de aprendizaje diseñado.

3. Creación de un Modelo de Aprendizaje Utilizando Videojuegos

A. Modelo de aprendizaje propuesto

Partiendo del conocimiento adquirido al analizar los videojuegos educativos y las diferentes recomendaciones pedagógicas, se procede con la creación del primer entregable de la investigación, el cual consiste en el diseño de un modelo de aprendizaje que permita la integración de las clases catedráticas tradicionales y el proceso de desarrollo y utilización de videojuegos educativos como componente núcleo para la ejecución de la clase.

A través de este modelo se pretende dar inicio a un proceso de planeación, implementación y ejecución de las clases de ingeniería de software que permita la utilización de un videojuego educativo creado con base en el plan académico de la Universidad, permitiendo de esta forma que los estudiantes tengan de esta forma una nueva herramienta

didáctica que puedan utilizar tanto dentro como fuera de la clase como instrumento de aprendizaje:

Este modelo consta de:

Etapa de planeación: Esta etapa funciona como el primer escalón dentro del modelo, aquí se construirá el plan académico a través del cual se creará el videojuego como herramienta de soporte a las clases cátedra tradicionales, esta etapa se divide en dos subetapas:

- Planeación de la clase: el docente se encarga de generar el plan de ejecución de la clase a través de una sección teórica y una práctica.
- Creación de videojuego: El equipo de desarrollo trabaja con el docente para desarrollar un videojuego que permita dar apoyo al estudiante antes y durante la clase.

Etapa de ejecución: La segunda etapa corresponde a la puesta en producción de los productos obtenidos durante la planeación. Aquí, los estudiantes ingresan al modelo como participantes en dos subetapas:

- Pre-clase: El estudiante recibe con anterioridad el material necesario para la ejecución de la clase con la intención de prepararlo para la misma.
- Clase: el docente procederá según lo establecido durante el periodo de planeación y dará la explicación y retroalimentación de los temas tratados en el videojuego.

Etapa de ejecución: Con la clase finalizada, es necesario realizar un proceso de retroalimentación a través de la cual se podrá mejorar y actualizar el trabajo realizado hasta el momento. Esta última sección busca dos aspectos principales:

- Postclase: tiene como objetivo mejorar la interacción entre el docente y los estudiantes, para esto se propone en conjunto con las actividades tradicionales como lo son las tareas y los talleres, una sesión de preguntas y respuestas a través de la cual el docente podrá ofrecer una mejor guía de aprendizaje a los estudiantes.
- Creación de videojuego: Aquí se busca analizar los resultados obtenidos durante la ejecución de todas las etapas anteriores permitiendo así mejorar las clases futuras y corregir los elementos desarrollados anteriormente.

Una vez finalizado el modelo, se propone la creación de un videojuego educativo que permita explicar temas de la clase de ingeniería de

software como herramienta de evaluación del mismo. Por este motivo, una vez finalizado dicho juego, se realizarán pruebas con estudiantes de ingeniería de sistemas e informática, permitiendo así la ejecución de un ciclo completo del modelo, no obstante, para la elaboración de esta puesta en producción se presentan algunas limitaciones.

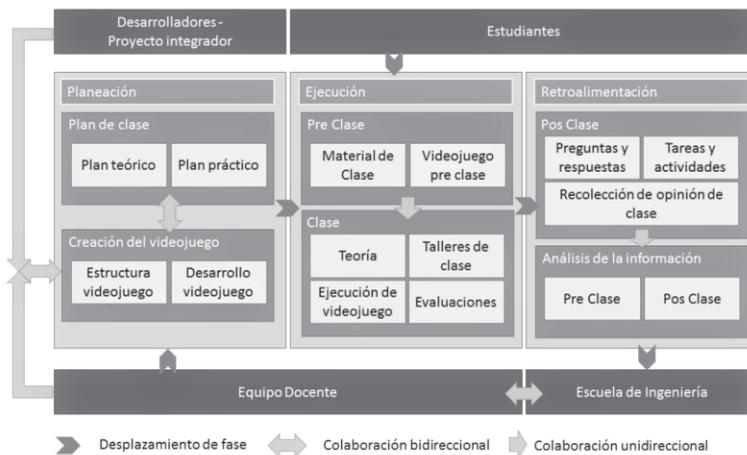


Fig. 1. Modelo propuesto para la enseñanza de ingeniería de software a través del uso de videojuegos.

En primer lugar, el videojuego se estará implementando de manera extracurricular, motivo por el cual no se podrá realizar una planeación completa de una clase con base en el programa académico definido por la Universidad y los docentes, impidiendo continuar completamente con el funcionamiento propuesto durante la etapa de planeación. Esto representa una diferencia importante que no debe pasar desapercibida respecto a la invitación realizada de convertir la implementación de esta actividad como un ejercicio pensado a largo plazo que involucre diferentes actores dentro de la institución académica.

La segunda limitación identificada hace referencia al proceso de entrega de contenido previo durante la etapa de ejecución. Debido a que esta actividad no se realiza dentro de una clase Universitaria, no es posible entregar el videojuego que los estudiantes utilizarán como material de estudio momentos previos al proceso de ejecución del mismo.

De la misma manera, la actividad de retroalimentación de los conceptos de ingeniería a explicar con el videojuego no contará con un correcto proceso de retroalimentación correspondiente pues la prueba no fue diseñada en conjunto con el plan de clase de los docentes, dificultando su empalme con el contenido temático existente en el programa académico de los estudiantes.

Con base en las limitaciones identificadas de la prueba del modelo propuesto, se da inicio a la etapa de planeación. Utilizando como base conceptual el tema de generalidades de ingeniería de software pues de esta forma el prototipo de videojuego a desarrollar podrá exponerse como un elemento introductorio a la materia y desde un punto de vista técnico, facilitará la entrega del producto completo dentro del tiempo previsto.

4. Identificación de las Características del Videojuego a Desarrollar

A. Clases de videojuegos

El modelo desarrollado se encarga de definir los lineamientos a seguir para el desarrollo de un videojuego educativo, sin embargo, existen diferentes formas de desarrollar videojuegos según la experiencia que se quiera ofrecer. Para el caso puntual de un videojuego educativo se identifican las siguientes características que debe tener según la estructura definida por el modelo educativo y las diferentes recomendaciones encontradas para la elaboración de contenidos pedagógicos, estas son:

- El videojuego debe ser lo suficientemente sencillo para garantizar que cualquier estudiante pueda utilizarlo sin importar si cuenta o no con experiencia como video-jugador.
- El videojuego debe poder ejecutarse en la mayor cantidad de equipos de cómputo posibles, esto significa que no debe exigir demasiados recursos de máquina.
- El videojuego debe poder presentar a los estudiantes gran cantidad de teoría seguida de ejemplos prácticos.

Con base en lo anterior, surge la pregunta ¿Qué clase de videojuego se deberá desarrollar? Lo cual conlleva a una nueva etapa de investigación teórica, pues existen diversos géneros dentro de este medio, de los cuales se destacan los siguientes como los más utilizados:

- **Acción:** Estos juegos conocidos por un mayor enfoque en situaciones de combate como elemento primario de jugabilidad, por lo que requieren que el jugador tenga buenos reflejos y coordinación con el mando de control y aquello que se visualiza en la pantalla para poder superar los obstáculos.

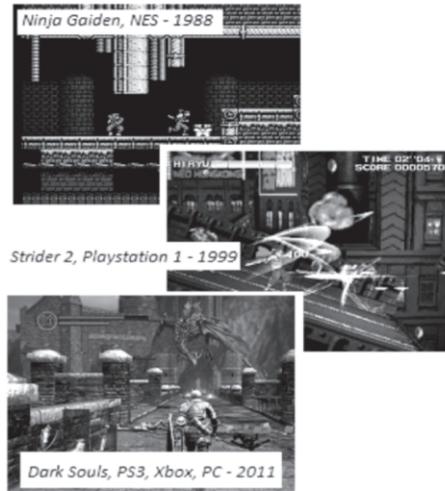


Fig. 2. Videojuegos de acción – Autoría propia.

- Plataformas: Estos juegos se enfocan en el desplazamiento del jugador a través de un mundo compuesto por diversas plataformas, usualmente a través de saltos pero sin limitarse a estos. Usualmente, estos juegos requieren una gran coordinación ojo – mano para poder superar obstáculos complejos.

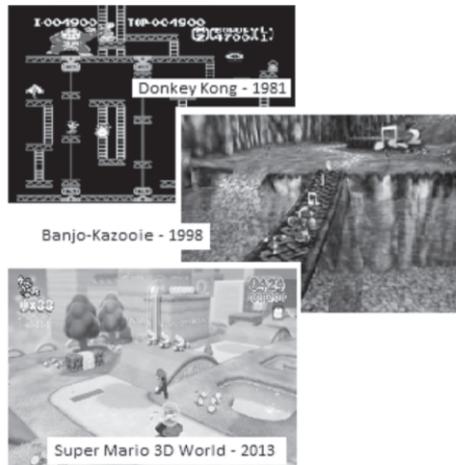


Fig. 3. Videojuegos de plataformas – autoría propia.

- Aventuras: Esta clase de juegos se enfocan en una jugabilidad más relajada sin necesidad de requerir gran cantidad de reflejos y coordinación para avanzar. Este tipo de juegos en su mayoría ofrecen

una historia como foco central y una serie de desafíos y acertijos como elementos para avanzar.



Fig. 4. Videojuegos de aventura – autoría propia.

- Disparos: Este género se caracteriza por usar proyectiles como elemento principal de jugabilidad los cuales deben dispararse contra objetivos para poder avanzar. Los juegos más populares utilizan cámaras en primera persona, simulando así estar en frente de la acción.



Fig. 5. Videojuegos de disparos – autoría propia.

- Simuladores: Esta clase de videojuegos busca ofrecerles a los jugadores la oportunidad de simular aspectos cotidianos como la administración de un parque, construcción de ciudades, etc. Ya sea en un entorno apegado a la realidad o completamente fantástico.



Fig. 6. Videojuegos de simulación – autoría propia.

- Estrategia en tiempo real (RTS – Real Time Strategy): Estos videojuegos buscan poner al video-jugador en situaciones que requieren administración cuidadosa de los recursos asignados y a pensar críticamente para anticiparse a los resultados de sus acciones para poder alcanzar las metas planteadas.



Fig. 7. Videojuegos de estrategia en tiempo real – autoría propia.

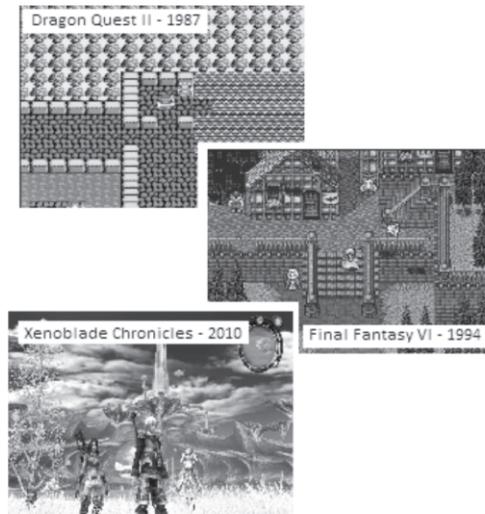


Fig. 8. Videojuegos de rol – autoría propia.

- Juegos de Rol (RPG – Role Playing Games): Buscan que el jugador tome el rol de uno o varios personajes para poder alcanzar las metas. Estos juegos se han caracterizado por el uso de una historia detallada y la personalización o incremento de características del personaje según eventos o situaciones que ocurren durante el transcurso del juego.

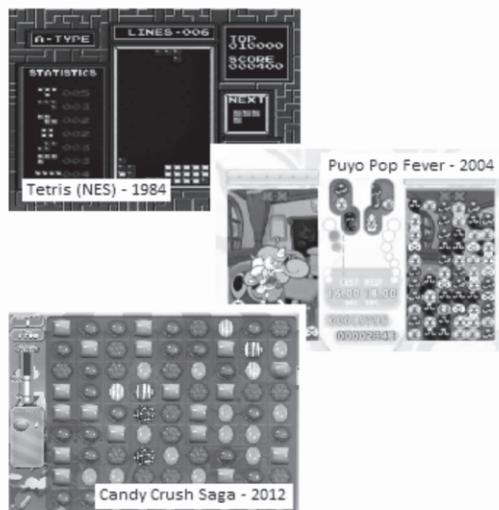


Fig. 9. Videojuegos de acertijos – autoría propia.

- Acertijos o retos: Estos videojuegos buscan que los jugadores resuelvan una serie de situaciones problemas para poder avanzar.

Aparte de estos, existen muchos géneros adicionales sin embargo, muchos de estos pueden ser considerados como mezclas de estas categorías generales. Por ejemplo, la gran mayoría de videojuegos de terror son considerados mezclas de acción y aventura, motivo por el cual se omite la mención de los mismos.

De igual manera, al finalizar la revisión de los diferentes géneros de videojuegos, se identifica que, las características de cada uno de estos videojuegos pueden tener un enorme grado de complejidad según sea la meta propuesta; es por esto que los equipos de desarrollo dentro del medio dan inicio a sus proyectos con la creación o adquisición de un medio que facilite el proceso de codificación y colaboración dentro del proyecto. Estas herramientas son conocidas como motores gráficos pues representan el núcleo con el cual se desarrolla el videojuego.

B. Seleccionando un motor gráfico

El motor gráfico es un elemento de software encargado de controlar, gestionar y actualizar los elementos multimedia (gráficos, sonido, conexiones, etc.) en tiempo real. La selección de un correcto motor gráfico permite disminuir los tiempos de desarrollo, facilitar la transferencia del conocimiento desde un punto de vista técnico y facilitar la implementación de elementos multimedia de los juegos como los gráficos y los efectos de sonido.

Es importante identificar correctamente las herramientas y el lenguaje de desarrollo a utilizar pues de esto depende en gran medida la proyección del cronograma de desarrollo para la creación del videojuego. Esta posibilidad de simplificar el proceso de creación del software, ayuda a los autores del videojuego de manera significativa, permitiéndoles finalizar sus productos en un menor tiempo, manteniendo un estándar fácil de seguir por todos los integrantes del equipo, haciendo más fácil el proceso de soporte y documentación técnica del motor o la comunidad y la fácil integración a diferentes plataformas como lo son sistemas operativos de equipos de cómputo, consolas de videojuegos o incluso dispositivos móviles como celulares.

Por este motivo, tomando como característica prioritaria el poco tiempo de ejecución del proyecto e identificando como medida adecuada para reducir la complejidad del proceso de creación de software y alcanzar la fecha de entrega esperada dentro de un periodo de seis meses, se decide tomar buscar un motor gráfico existente.

A continuación se presenta un listado de las herramientas identificadas:

- REN'PY [14]: Motor gratuito para el desarrollo de videojuegos que permite la creación de aventuras gráficas o novelas virtuales, estas consisten principalmente en el despliegue de imágenes, texto y se enfocan en la narrativa, también permite la creación de simuladores de vida o historias con rutas y variaciones lo que genera historias dinámicas.
- UNITY ENGINE [15]: Unity es un poderoso y flexible motor gráfico que permite la creación de en 3D y 2D, muy flexible y enfocado en multiplataforma, lo que permite importar el proyecto fácilmente a diversos sistemas, permite la creación de videojuegos en una amplia gama de géneros, desde juegos de acción, plataformas, disparos, lucha, juegos de Rol o RPG, carreras entre muchos otros.
- RPG MAKER VX Ace [16]: motor gráfico para video juegos en dos dimensiones (2D), su enfoque principal es la creación de juegos de Rol o RPG (Rol Playing Games), no obstante este también puede emplearse para la creación de aventuras gráficas o juegos de acción simples aunque principalmente maneja combates por turnos, se promociona bajo la idea de ser un motor fácil de usar mas no obstante lo suficientemente poderoso como para desarrollar.
- UNREAL ENGINE [17]: Muy semejante al Unity Engine, Unreal es un poderoso motor gráfico que permite generar videojuegos de diversos géneros tanto en ambientes 3D como 2D, un poco menos flexible que Unity encuentro a su compatibilidad con diversos sistemas, no obstante Unreal es compatible con todos los sistemas más populares.
- CRYENGINE [18]: Un poderoso motor gráfico que da soporte a todos los sistemas de última generación, desde computadores, equipos móviles y hasta consolas de videojuegos, se promociona como un motor que permite lograr gráficas impresionantes, físicas muy reales, audio de alta calidad, inteligencia artificial fácil de programar y compatibilidad directa con Direct X 11.
- GAME MAKER STUDIO [19]: Su enfoque es principalmente en videojuegos en 2D, no obstante permiten el uso de algunas gráficas en 3D, se promociona especialmente bajo la idea de ser un motor fácil de usar que es compatible con una gran cantidad de sistemas y cuenta con herramientas multiplataforma, permite la creación de varios géneros entre los que se encuentran aventuras gráficas, RPG, acción, plataformas entre otros.

- BLENDER GAME STUDIO [20]: El motor gráfico funciona muy de la mano con la herramienta gráfica Blender, permite la creación de videojuegos de diversos en ambientes 3D, no se recomienda para manejo de videojuegos en ambiente 2D; no obstante se puede emplear para generar gráficas en 2.5D o gráficas en 3D que se comportan como si se encontraran limitadas a un ambiente de dos dimensiones.

5. UNAB Vr: Un Videojuego para la Enseñanza de la Ingeniería de Software

A. Definiendo el videojuego a desarrollar

Finalizada la etapa de indagación tecnológica anterior, se procede con la identificación de las características del videojuego a desarrollar y la selección del motor gráfico a utilizar, con miras a la finalización del segundo producto de esta investigación. Para poder identificar correctamente la clase de videojuego a desarrollar, se procede a identificar en primera medida la estructura general del videojuego que se desea desarrollar, el cual, para este caso puntual, busca presentar a los estudiantes el contenido introductorio a la materia.

Luego, se da inicio a la identificación del conjunto de contenidos que se deberán explicar a los estudiantes, siendo estos principalmente teóricos debido a la naturaleza misma de la materia que se desea explicar. No obstante, indicar la teoría de la ingeniería de software como un todo resultaría en un proceso demasiado extenso para poder incluir en un prototipo inicial, adicionalmente, se debería contar con mucho más tiempo que aquel estipulado para la elaboración del proyecto. Por este motivo, se decidió limitarlo únicamente a la primera parte del proceso de ingeniería de software, es decir, a la definición de requerimientos y al conocimiento básico del tema para poder realizar el levantamiento de los mismos.

Este contenido se dividió en 4 secciones principales:

1. Definición e importancia de los requerimientos.
2. Características de los requerimientos de software.
3. Métodos de levantamiento de requerimientos.
4. Elaboración general de un documento.

Ahora, esta clase de temas resultan ser bastante teóricos, luego se procede con la revisión de las características en detalle de los diferentes motores gráficos identificados para poder identificar cual se puede

acoplar mejor a los requerimientos existentes. Esta consulta, identifica que los motores más robustos como CryEngine, Unreal Engine y Unity resultan ser bastante versátiles, sin embargo, la curva de aprendizaje es elevada y se presenta el riesgo de no poder entregar el juego en la fecha estimada. Por este motivo se decide simplificar la elección a los motores más fáciles de aprender a utilizar, siendo estos Game Maker y RPG Maker VX ACE.

Queda pendiente entonces la identificación de las características propias del motor gráfico que permitan un mejor acoplamiento a las necesidades del proyecto. Sin embargo, para esto, se decide tomar como factor decisivo la identificación del género de videojuego a utilizar.

Partiendo de la idea definida para el contenido a desarrollar en donde se planea separar cada sección de contenido como un componente temático independiente, y de la estructura del modelo propuesto, se decide crear una estructura de clase de la siguiente manera:

- Inicio con una sección teórica.
- Ejecución de un ejercicio que ayude a reforzar la idea o conceptos explicados.
- Complemento o refuerzo teórico
- Actividad o taller final.

Con esto, se identifica que esta estructura de clase virtual resulta mucho más fácil de implementar a través de un juego de rol (RPG) pues este género se puede beneficiar más de la narrativa y estrategia planteada, de igual manera, se identifica una curva de aprendizaje bastante inferior a las demás, asegurando la entrega del proyecto dentro del periodo de tiempo deseado. Luego, se decide seleccionar el motor gráfico RPG Maker VX ACE al considerarse la herramienta más práctica a utilizar para el proceso de desarrollo pues se acopla de manera fácil a la estructura de videojuego deseada.

B. Estructura de la clase virtual

Una vez seleccionado el motor gráfico, se procede a la planeación de cada una de las clases, los niveles del juego y los personajes. Estos dos últimos elementos se solucionaron fácilmente gracias a las diferentes herramientas provistas por el motor gráfico seleccionado, el cual cuenta con un editor de mapas y una serie de elementos visuales que pueden utilizarse sin inconveniente, acortando así lo que sería el proceso de elaboración de los elementos desde cero y acortando los tiempos de entrega.

El paso siguiente es el de la planeación de las clases y a la elaboración del guion que servirá como guía para el correcto desarrollo de los mismos. Para esto se realizó una búsqueda en diferentes libros y materiales utilizados actualmente por la Universidad Autónoma de Bucaramanga para la planeación de las clases dentro de su catálogo real. No obstante, se debe realizar algunos ajustes a la manera de presentar dicha información pues como se mencionó anteriormente, aquello que funciona correctamente en el mundo físico, puede presentar inconvenientes dentro del mundo virtual o simplemente no funcionar como se espera.

Por este motivo, se diseñó un personaje con el rol de docente de ingeniería para la clase, a través del cual se esté entregando el conocimiento y sirva como punto de partida para ir a los talleres de clase.

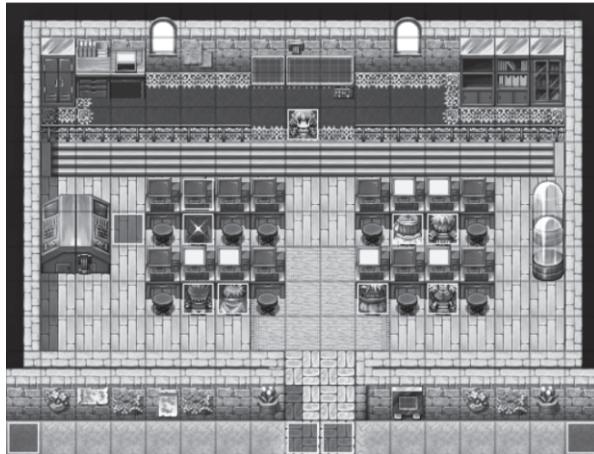


Fig. 10. Imagen del salón de clases creado dentro del videojuego utilizando el motor gráfico RPG Maker VX Ace.

El docente del salón virtual es el encargado de indicar cuando se dará inicio a cada uno de los talleres, los cuales se ejecutarán en un mapa de juego totalmente diferente a la zona estándar. Cada taller fue diseñado de forma tal que diera soporte a un concepto de clase de la siguiente forma:

- Taller 1: Tiene como objetivo presentar una analogía de qué ocurre cuando se encuentra a un cliente que no está seguro de lo que desea. Para su funcionamiento se creó una tienda, dentro de la cual llegarán dos clientes seleccionados aleatoriamente a partir de una serie de casos creados con anterioridad.

A través de este taller, se espera resaltar la importancia de identificar los requerimientos del cliente antes de proceder con el desarrollo.

- Taller 2: El segundo taller fue concebido como una evaluación de selección múltiple con dos respuestas de la cual solo una es verdadera con la intención de validar el conocimiento de los estudiantes utilizando una estructura de examen tradicional de los salones de clase. Este ejercicio se desarrolla en un mapa conformado por diferentes cuartos de manera similar al estilo de un laberinto, en donde cada respuesta correcta acercará al jugador a la salida del mapa, mientras que las respuestas erradas lo alejaran de esta.

El jugador tiene un tiempo limitado para poder finalizar el laberinto, en caso de no alcanzar la salida antes de que el contador de tiempo llegue a cero, se deberá repetir el taller.

- Taller 3: Este taller tiene como objetivo realizar una entrevista de levantamiento de requerimientos en una empresa. Para poder implementar el taller fue necesario definir un guion que se asimilara a un caso real. Durante esta actividad el usuario deberá entablar diálogos con los empleados que cuentan con la información necesaria para identificar los requerimientos, cada usuario tendrá una serie de diálogos y el usuario deberá seleccionar que parte del dialogo corresponde realmente a un requerimiento o a información que debe ser omitida.

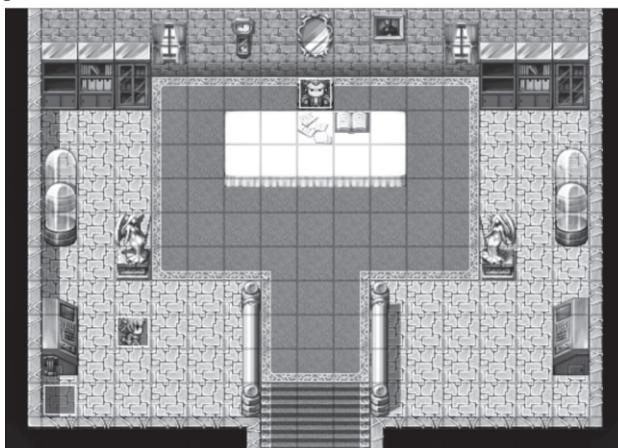


Fig. 11. Imagen del taller 3 creado con el motor gráfico RPG MakerVX Ace.

- Taller 4: El cuarto taller retoma la idea de la validación del conocimiento a través de preguntas con posibles respuestas. Sin embargo, para representar un cambio de ambientación, se realizará un cambio en el entorno al presentar un grupo de enemigos que solo serán derrotados con la respuesta correcta.
- Taller 5: Finalmente, el último taller busca el ordenamiento de la estructura de un documento de requerimientos. Para esto, el jugador deberá encontrar e identificar una serie de personajes, cada uno con una sección del documento.

C. Diseñando los elementos del videojuego

Gracias a las herramientas internas del motor gráfico, no fue necesario generar demasiados elementos multimedia desde cero, pues la herramienta cuenta con su propia estructura de personalización para la creación de componentes a utilizar dentro del videojuego. Luego, para el desarrollo del prototipo, se diseñaron los siguientes personajes, cada uno de ellos cuenta con una función específica dentro del videojuego con la finalidad de dar mayor variedad durante la ejecución del juego y al mismo tiempo mantener un estilo similar al género de juego de rol seleccionado.

- Docente: Encargada de presentar toda la teoría de las clases y de llevar al jugador a los talleres.
- Vendedor: Personaje utilizado únicamente dentro del ejemplo de introducción.
- Cliente: Personaje utilizado únicamente dentro del ejemplo de introducción.
- Dueño de tienda: Personaje encargado de explicar el funcionamiento del taller 1.
- Jefe: Personaje utilizado para explicar las bases de una entrevista de requerimientos en el taller 3.
- Ingeniero: Personaje utilizado para explicar las bases de una entrevista de requerimientos en el taller 3.
- Gestor documental: Personaje utilizado para el taller 3 como entrevistado.
- Publicista: Personaje utilizado para el taller 3 como entrevistado.
- Relaciones públicas: Personaje utilizado para el taller 3 como entrevistado.
- Domador de monstruos: Personaje utilizado para dar inicio al taller 4

Finalmente, se generó un guion que define los diálogos de cada uno de los personajes durante el transcurso de la clase. Permitiendo así, agregar un componente narrativo que corresponde a uno de los elementos característicos del género de videojuego utilizado y al mismo tiempo, facilita la transferencia de la información con los jugadores.

D. Características del desarrollo del videojuego utilizando el motor gráfico RPG Maker VXACE

La programación de RPG Maker VXACE utiliza una interfaz gráfica de usuario amigable con una funcionalidad de seleccionar e implementar según las herramientas ofrecidas por el motor. Por ejemplo, para la creación de un nuevo juego, el programador solo necesita ingresar a la

aplicación, crear un nuevo proyecto y seleccionar la carpeta en donde se guardará el producto final. Una vez realizada esta actividad el usuario contará con la amplia gama de herramientas para comenzar a crear su videojuego.

A través del asistente interno de la herramienta se procede a generar una serie de mapas o escenarios, en donde se estarán realizando las diferentes actividades programadas para cada una de las clases. El comportamiento de cada componente pedagógico se programa a través de eventos, los cuales, hacen referencia a elementos programados para presentar acciones específicas según las condiciones requeridas para cada componente.

La comunidad wiki de RPG Maker en español, realiza una breve traducción de la documentación oficial en inglés definiéndolos como “Eventos comunes o funciones que contienen los códigos generales que están a disposición del juego en todo momento, independientemente del mapa en que se esté ubicando” (Colaborador, 2015). A través de estos eventos, es posible programar funcionalidades dentro de un juego utilizando el editor del motor gráfico sin necesidad de escribir una línea de código.

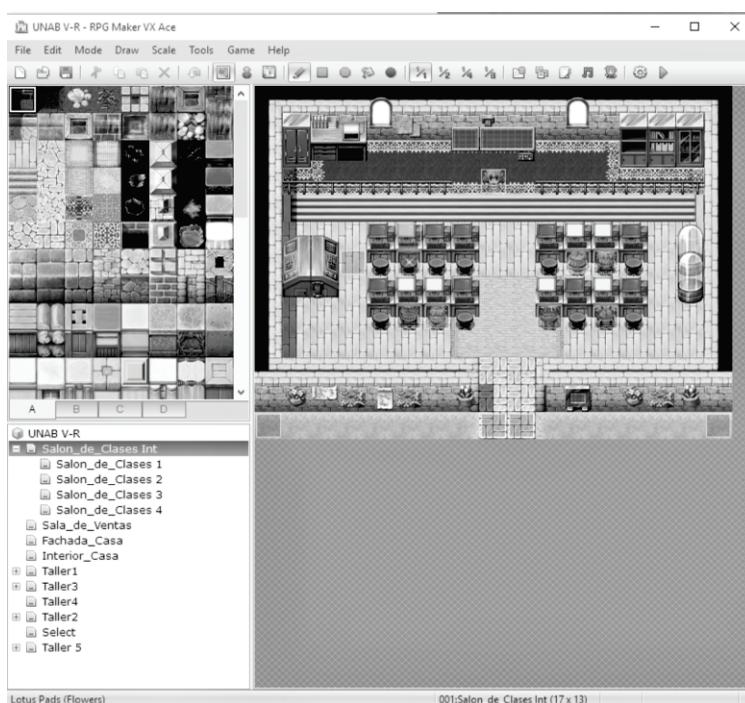


Fig. 12. Muestra en interfaz de edición de RPG Maker VX – Autoría propia.

Para poder agregar un evento en el mapa, el desarrollador deberá seleccionar un lugar en el mapa y pulsar la opción de agregar un evento nuevo. Una vez creado el evento, el usuario procederá a agregar las propiedades del evento como nombre, gráfico a utilizar, condiciones de inicio, condiciones de finalización, variables a utilizar, etc. En cuanto al comportamiento del mismo, se utiliza el editor de la aplicación para poder ajustar las variables directamente, por ejemplo, para mostrar un texto en pantalla, se seleccionará el comportamiento directamente desde el asistente y se insertará solicitando al usuario por los caracteres a mostrar. Este mismo asistente se utilizará para poder agregar todos los comportamientos soportados por el motor según los parámetros de los juegos de rol de los que este toma inspiración.

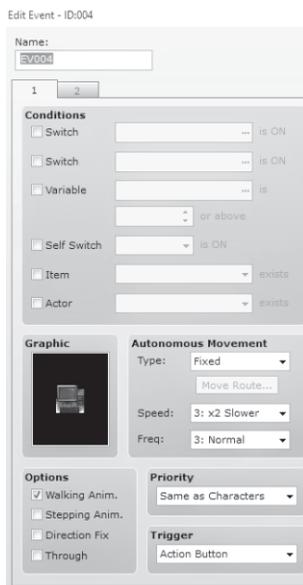


Fig. 13. Evento de encender PC utilizado en UNAB VR 1 - Autoría propia.



Fig. 14. Evento de encender PC utilizado en UNAB VR 2 – Autoría propia.



Fig. 15. Ejemplo de selección de comportamientos – Autoría propia.

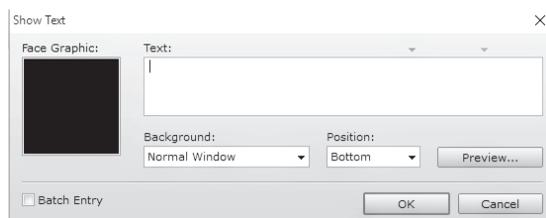


Fig. 16. Ejemplo de solicitud de parámetros para instalar texto – Autoría propia.

E. Evaluación del videojuego

Una vez finalizado el prototipo de videojuego, se realizó una actividad con una población muestra de 17 estudiantes de clases de ingeniería de software y programación, los cuales se encontraban cursando quinto semestre de Ingeniería de sistemas al momento de realizar la prueba en donde se les permitió utilizar el videojuego desarrollado por un momento previo a una clase y con base en la experiencia, A continuación se presenta una tabla con las preguntas realizadas a los estudiantes.

1. ¿Su nivel actual de conocimiento en Ingeniería de Software es?:		
Excelente	Bueno	Malo
6,3%	75,0%	18,8%
2. Teniendo como base el simulador UNAB VR, ¿Considera usted que esta herramienta facilita el aprendizaje de ingeniería de software?		
Sí	No	No está seguro
84,5%	0,0%	15,5%
3. Teniendo como base el simulador UNAB VR, ¿Aprendió algo nuevo de ingeniería de software durante el tiempo dedicado a usar este simulador?		
Sí	No	
100,0%	0,0%	0,0%
4. ¿Considera usted que simuladores como UNAB VR pueden mejorar su motivación en el aprendizaje de la ingeniería de Software?		
Sí	No	No está seguro
87,5%	6,3%	6,3%
5. ¿Cómo definiría usted la facilidad de uso de UNAB VR?		
Fácil	Ni fácil ni difícil	Difícil
62,5%	37,5%	0,0%
6. Desde un punto de vista no académico, ¿considera usted que UNAB VR es entretenido?		
Sí	No	
81,2%	18,8%	
7. Teniendo en cuenta la interfaz gráfica de usuario de SIMSE, ¿Considera que la GUI es fácil de entender y utilizar?		
Sí	No	No está seguro
100,0%	0,0%	0,0%

Tabla 1. Encuesta elaborada a los estudiantes de clases de ingeniería de sistemas.

Esta encuesta también contaba con una serie de preguntas abiertas, a través de las cuales los participantes podrían dar información más detallada acerca de su opinión personal respecto a su experiencia con el videojuego:

8. ¿Qué le mejoraría a la aplicación?
9. ¿Que opina acerca del uso de videojuegos como herramientas para el aprendizaje de Ingeniería de Software?
10. ¿Estaría dispuesto a usar videojuegos que explicaran conceptos de ingeniería de software fuera de los horarios de clase?

Tabla 2. Segunda parte de la encuesta implementada en los estudiantes de ingeniería de sistemas.

Iniciando a la revisión de los resultados, se encuentra como primer dato de interés es el nivel de conocimiento de los estudiantes pues tal como se anticipaba, la gran mayoría de los participantes considera contar con un conocimiento bueno respecto a los temas de ingeniería de software. Esta información resulta bastante oportuna pues las preguntas de validación de conocimiento cuentan de esta forma con impacto mucho mayor dado a que ninguno de los estudiantes había recibido información alguna respecto al tema tocado a través del videojuego según indicó el docente de la clase.

Ahora, gracias a la segunda pregunta, se logra identificar que el 87,5% de los estudiantes considera esta clase de iniciativas de aprendizaje como algo positivo que realmente facilita el aprendizaje. El 12,5% restante no está seguro lo cual no representa su aceptación, pero tampoco demuestra rechazo ante el concepto propuesto. Lo anterior sirve para corroborar la premisa de Prensky [4], quien menciona que las nuevas generaciones tienden a interesarse más ante este tipo de herramientas.

La tercera pregunta presenta otro dato interesante, pues un 100% de los participantes asegura haber aprendido algo de ingeniería de software. Lo cual se corrobora de manera sencilla con los resultados de la sección de validación de conocimiento y la premisa de aprendizaje de temas de ingeniería de software a través de videojuegos. De manera similar, la pregunta número cuatro muestra que la gran mayoría de los estudiantes considera el uso de videojuegos como una forma de incrementar su motivación con un 80% a favor contra un 10% en contra y un 10% que no está seguro.

Respecto a la ejecución de las preguntas de validación de conocimiento, se identificó que:

- Un 60% de los participantes llegó a la sección final del videojuego en el tiempo estipulado.
- Todos los estudiantes pudieron responder la pregunta relacionada con la clase que alcanzaron a visualizar.

En cuanto a la funcionalidad del videojuego, un 62,5% de los participantes responde que es fácil de utilizar, el 37,5% menciona que no

es ni fácil ni difícil lo cual, al transponerlo con la respuesta a la pregunta 7 en la cual el 100% de los estudiantes responde que la interfaz gráfica del juego es fácil de entender, permite inferir que el prototipo desarrollado representa una herramienta pedagógica de fácil manejo que según lo estipulado en la respuesta a la pregunta 6, resulta divertido para la mayoría de los estudiantes con un 81,3% de aprobación contra un 18,8% de rechazo, presentando un grado favorable de aceptación de los estudiantes.

Finalmente, en las preguntas abiertas se identificaron las siguientes respuestas:

• **¿Qué le mejoraría a la aplicación?**

- Hacer más claros los tutoriales.
- Migrar el juego a dispositivos móviles.
- Migrar el juego a la web
- Mejorar los gráficos.
- Mejorar las recompensas.

• **¿Que opina acerca del uso de videojuegos como herramientas para el aprendizaje de Ingeniería de Software?**

- En promedio fue considerado una buena idea.
- Se debe utilizar solo como herramienta de ayuda y no como fuente principal de conocimiento.

• **¿Estaría dispuesto a usar videojuegos que explicaran conceptos de ingeniería de software fuera de los horarios de clase?**

- En general los estudiantes están a favor de esta clase de iniciativas siempre y cuando sean solo usadas como materiales de apoyo.

Estos últimos resultados permiten ver con buenos ojos la implementación de esta clase de elementos pues demuestran en términos generales la aceptación e interés de estudiantes actuales ante la propuesta académica planteada; sin embargo, se aprecian varias críticas que pueden ser utilizadas para mejorar la aplicación, siendo puntos importantes la mejora de los elementos de jugabilidad de tal manera que el funcionamiento del videojuego ofrezca una mayor variedad de modos de juego y de recompensas por finalizar los talleres u objetivos diseñados y la posibilidad de expandir el videojuego a plataformas de mayor facilidad de acceso como los dispositivos móviles o la red.

6. Comparación de Unab Vr con el Videjuego Simse

Para poder identificar los cambios en la percepción de los estudiantes respecto a SIMSE como aplicación enfocada en la puesta en práctica de conocimientos ya adquiridos a modo de simulador y UNAB V-R como un juego diseñado para apoyar al estudiante con la adquisición de nuevos conocimientos, se realiza una comparación entre las encuestas realizadas de tal manera que sea posible identificar condiciones de mejora de los aplicativos respecto a otras tecnologías existentes.

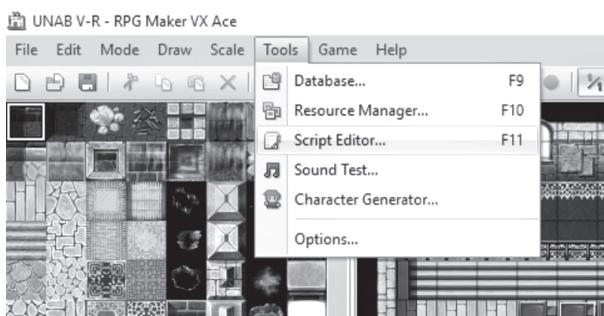


Fig. 17. Imagen de funcionamiento del motor gráfico RPG Maker VX Ace.

2. Teniendo como base el videojuego, ¿Considera usted que esta herramienta facilita el aprendizaje de ingeniería de software?					
Sí		No		No está seguro	
SIMSE	UNAB VR	SIMSE	UNAB VR	SIMSE	UNAB VR
81,8%	84,5%	0%	0,0%	18,2%	15,5%
3. Teniendo como base el videojuego, ¿Aprendió algo nuevo de ingeniería de software durante el tiempo dedicado a usar este simulador?					
Sí		No			
SIMSE	UNAB VR	SIMSE	UNAB VR		
90,9%	100,0%	0,0%	9,1%		
4. ¿Considera usted que simuladores como el videojuego pueden mejorar su motivación en el aprendizaje de la ingeniería de Software?					
Sí		No		No está seguro	
SIMSE	UNAB VR	SIMSE	UNAB VR	SIMSE	UNAB VR
81,8%	87,5%	0%	6,3%	18,2%	6,3%
5. ¿Cómo definiría usted la facilidad de uso del videojuego					
Fácil		Ni fácil ni difícil		Difícil	
SIMSE	UNAB VR	SIMSE	UNAB VR	SIMSE	UNAB VR
18,2%	62,5%	72,7%	37,5%	9,1%	37,5%
6. Desde un punto de vista no académico, ¿considera usted que el videojuego es entretenido?					
Sí		No			
SIMSE	UNAB VR	SIMSE	UNAB VR		
72,7%	81,2%	27,3%	18,8%		
7. Teniendo en cuenta la interfaz gráfica de usuario del videojuego, ¿Considera que la GUI es fácil de entender y utilizar?					
Sí		No		No está seguro	
SIMSE	UNAB VR	SIMSE	UNAB VR	SIMSE	UNAB VR
36,4%	100,0%	45,5%	0,0%	18,2%	0,0%

Tabla 3. Resultados de la comparación de UNAB VR y SIMSE.

A través de esta comparación se identifica que en el caso de ambas aplicaciones, los estudiantes consideran que efectivamente el uso de estas herramientas les puede facilitar el aprendizaje de temas de Ing. de software, corroborando el interés de los estudiantes respecto de esta clase de tecnología.

En la tercera pregunta se logra identificar una ligera variación a favor de UNAB VR, pues si bien el porcentaje de usuarios que mencionó haber aprendido algo con la aplicación fue alta en SIMSE, el prototipo desarrollado en este proyecto logró obtener un resultado de 100% lo cual significa que todos los participantes adquirieron algún conocimiento nuevo.

Respecto a la cuarta pregunta, se identifica un pensamiento equivalente nuevamente respecto a la manera en que esta clase de herramientas puede ayudar a incrementar la motivación de los estudiantes al momento de aprender temas de Ing. de software.

Respecto al funcionamiento y facilidad de uso, nuevamente se observa un puntaje mucho más favorable para UNAB VR pues la gran mayoría de usuarios lo encontraron mucho más fácil de usar a diferencia de SIMSE el cual fue categorizado por la mayoría como algo ni fácil ni difícil.

Finalmente, se encuentra un resultado similar en cuanto al facto de entretenimiento; sin embargo, en cuanto a su interfaz gráfica de usuario, UNAB VR logra obtener nuevamente un 100% haciéndolo mucho más fácil de utilizar.

Ante esta comparación, resulta posible identificar que los estudiantes tienen una percepción bastante similar en cuanto a la cantidad de conocimiento académico que pueden adquirir a través de estos juegos, sin embargo, el prototipo propuesto logra destacar en la facilidad de uso y de manejo, lo cual ayuda a convertirlo en una herramienta mucho más fácil de implementar.

7. Mejoras de la Aplicación: Sistema de Metas Implementado

Continuando con la ejecución del modelo propuesto, se decide realizar la implementación de uno de los aspectos de mejora que se pudiera acoplar fácilmente dentro del cronograma de entrega. Luego, teniendo como base el interés de los estudiantes respecto a contar con alguna meta adicional que los ayudará a incrementar más su interés en la conclusión del videojuego se decide implementar un sistema básico de recompensas o logros dentro del juego.

El sistema de logros es un elemento utilizado en los videojuegos actuales como método de motivar al usuario a completar ciertos objetivos dentro del mismo. Su funcionamiento consiste en analizar el progreso del video jugador y en el momento en que este realiza una acción o completa algún objetivo específico, el juego mostrará un mensaje notificándole el alcance del logro y otorgándole ya sea algún elemento secreto o un puntaje.

Para poder implementar esta funcionalidad al prototipo, se realizó una búsqueda en la comunidad de RPG Maker VX ACE para poder obtener ideas que facilitaran su implementación. Dicha búsqueda resulto ser bastante útil pues un usuario dentro de la comunidad había publicado un script personalizado de uso libre para juegos no comerciales con la posibilidad de comprar una licencia que permita su implementación en usos comerciales [21].

Este script cuenta con tres partes para su implementación, la primera hace referencia al elemento principal o “Core Script” el cual contiene las instrucciones generales que le indican al juego como interactuar con los elementos correspondientes. El segundo elemento es el “Menu organizer” el cual permite modificar el menú del juego para poder realizar el seguimiento a los diferentes logros. Finalmente, se encuentra el “Achievements script” el cual contiene los datos necesarios para la configuración de los logros.

Para implementar estos elementos al juego se debe ingresar al menú “tools” de RPG Maker VX ACE y seleccionar la opción “Script editor”.

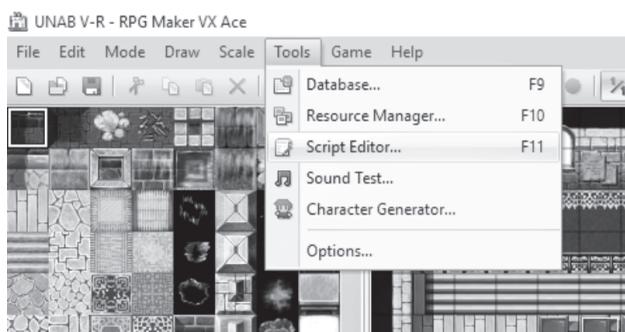


Fig. 18. Editor de Scripts del RPG Maker – Autoría propia.

Eso abrirá una ventana con todos los “scripts” del juego. En esta nueva ventana se deberá buscar la sección “Materials” en donde se copiarán y pegarán las líneas de código personalizadas. Una vez insertados los tres elementos, el motor gráfico se encargará de agregarlos a su funcionamiento interno automáticamente.



Fig. 19. Editor de Scripts del RPG Maker funcionando– Autoría propia.

Siguiendo las instrucciones suministradas por el “script”, se realizan los ajustes necesarios para poder agregar la funcionalidad al videojuego. Se decidió que se completaría un logro cada vez que un videojugador finalizara un taller de clase, otorgándole de esta manera no solo el mensaje de logro, sino un puntaje que podrá observarse desde el menú del juego.

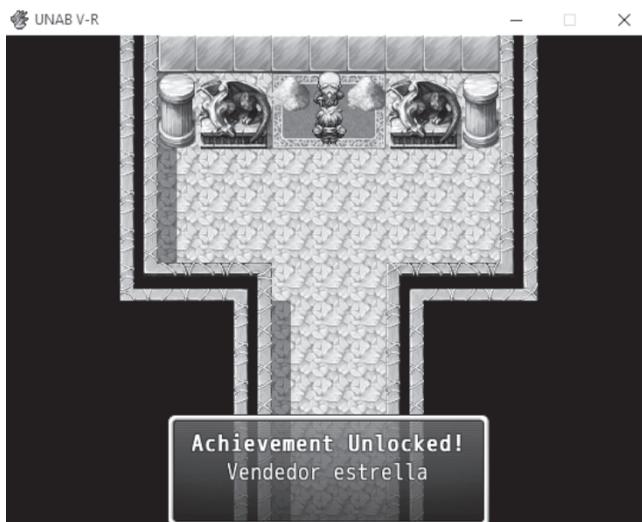


Fig. 20. Imagen del Sistema de metas implementado.

8. Conclusiones

- Fue posible diseñar un modelo funcional que incorporara el manejo de videojuegos como herramienta pedagógica para la enseñanza de ingeniería de software y que permitiera su incorporación de manera sencilla dentro de la estructura de clases tradicional.
- Se diseñó satisfactoriamente una estructura de enseñanza acorde al modelo planteado y las recomendaciones pedagógicas identificadas en el estado del arte, permitiendo así el diseño de una herramienta educativa fácilmente adaptable a cualquier clase teórica tradicional.
- Se desarrolló un videojuego acorde a las reglas planteadas por el modelo y el plan de enseñanza propuestas, el cual permitió la explicación de temas generales de la ingeniería de software, demostrando así la viabilidad de esta clase de herramientas pedagógicas en el entorno universitario.
- A través de la evaluación realizada se logró identificar un alto nivel de aceptación en los estudiantes universitarios de primeros niveles de Ingeniería informática y de sistemas corroborando la viabilidad de los videojuegos como herramientas académicas capaces de aumentar la motivación de los estudiantes a aprender temas con alto contenido teórico.
- El modelo pedagógico propuesto tiene un enorme potencial como herramienta integradora de diferentes facultades y áreas académicas dentro de una Universidad, pues este permite el trabajo colaborativo entre la facultad de ingeniería informática y sistemas y los demás departamentos de la institución, ya sea para formar equipos de desarrollo de componentes para el videojuego o como mediador para la creación del desarrollo de nuevos videojuegos educativos que atiendan las necesidades de las diferentes carreras profesionales existentes en el catálogo académico.
- Se identifica que el proceso de creación de videojuegos representa un trabajo de ingeniería bastante complejo el cual abarca no solo conceptos de programación y gestión de software, también es necesario contar con elementos multimedia cuya creación e integración representa un tiempo considerable que de no planearse adecuadamente puede presentar grandes retrasos en la entrega del producto.
- La selección de un motor gráfico comercial ya sea de código abierto o propietario ayuda considerablemente con la creación del producto

final pues elimina por completo la creación de las bases técnicas necesarias para la construcción de un videojuego y permite enfocarse en el desarrollo e implementación del mismo.

- Es necesario identificar correctamente el estilo de videojuego a desarrollar antes de proceder con el desarrollo del mismo, esto facilitará el proceso de selección de motor gráfico y permitirá reconocer fácilmente las brechas que deberán ser superadas a través de código personalizado.

9. Aporte Científico

Este trabajo investigativo aporta al campo pedagógico universitario una nueva perspectiva de enseñanza en donde se demuestra la viabilidad del uso de videojuegos como herramientas de enseñanza teóricas en clases universitarias. Enfocado en el acompañamiento a los estudiantes en momentos previos y durante la ejecución de la clase, diferenciándose así de otras propuestas existentes que buscan la implementación de videojuegos como simuladores post-clase utilizados para validar el conocimiento aprendido durante la ejecución de un curso completo.

De igual manera, con este proyecto se presenta una alternativa a través de la cual los docentes podrán entregar contenidos de clase de manera previa al desarrollo de la misma, logrando así que el estudiante llegue con mayor conocimiento a la misma. Esta premisa se apoya en el uso de videojuegos, pues permite que los estudiantes despierten un mayor interés en los temas propuestos al encontrarse ante un entorno familiar diferente a una sesión catedrática tradicional y permitiendo a los docentes ejercer un papel protagónico en el control de los contenidos que se entregarán.

10. Trabajo Futuro

Teniendo en cuenta las recomendaciones presentadas por los estudiantes y elementos que se salen del enfoque definido para la ejecución de este trabajo, se presentan las siguientes recomendaciones como elementos del trabajo futuro a desarrollar:

- Se recomienda llevar este trabajo más allá de un prototipo funcional y establecer un plan de trabajo para una materia, permitiendo así, evaluar dentro de un contexto real el modelo propuesto.
- El desarrollo de videojuegos cuenta con un gran potencial para integrar personas de diferentes áreas de conocimiento hacia un objetivo común. Por este motivo, se plantea la creación de un

programa de núcleo integrador que permita la colaboración de la facultad de ingeniería informática y sistemas con otras carreras de la universidad para la elaboración de un videojuego enfocado en una situación problema de interés de la institución.

- Teniendo en cuenta las recomendaciones de los estudiantes, se debe considerar la migración del proyecto a un motor gráfico más robusto que permita una mejora de los elementos visuales del juego, enfocándolos en modelos poligonales que estén más acordes con los videojuegos populares en el mercado actual.
- Se debe contemplar la posibilidad de migrar el proyecto a una plataforma online, aumentando así la posibilidad de entrega de los videojuegos a cualquier persona sin importar el dispositivo utilizado. Permitiendo de esta forma, disminuir los requerimientos de hardware para esta implementación y facilitar la entrega de los contenidos.

Agradecimientos

Agradezco desde la parte académica al director de este trabajo pues sus consejos ayudaron enormemente a enfocar el desarrollo del mismo para obtener así los mejores resultados y de manera personal a mi familia y amigos pues su apoyo fue crucial para poder avanzar durante el periodo de tiempo en que se ejecutó este trabajo investigativo.

Referencias

- [1] Bornand, S. I. (9 de Diciembre de 2013). Avances, Experiencias y Futuro de los Textos Escolares Digitales. Yucatán, Mexico. Recuperado el 2015, de <http://www.tise.cl/volumen9/TISE2013/30-35.pdf>.
- [2] Gatica, N., Valdivia, J., Sanhueza, V., Rodríguez, P., Bagueño, C., & Ávila, A. (2013). desarrollo de habilidades cognitivas en pre-escolar a través del uso del e-blocks. Concepción, Chile. Recuperado el 2015.
- [3] Canche, M., Leal, N., Llanes, E., & González, C. (2013). Explorando alternativas de apoyo a la educación universitaria: sesión tutorial basada en servicios móviles. Yucatán, México. Recuperado el 2013.
- [4] Prensky, M. (2006). Don't Bother Me Mom — I'm Learning. Estados Unidos: Paragon House. Obtenido de <http://marcprensky.com/dont-bother-me-mom-im-learning/>.

- [5] Prensky, M. (2001). Digital Game-Based Learning. United States of America. Recuperado el 2015, de <http://www.amazon.com/Digital-Game-Based-Learning-Marc-Prensky/dp/1557788634>.
- [6] Meyer, B. (2003). The Outside-In method of teaching introductory programming. Lecture Notes in Computer Science. Zurich, Suiza. Obtenido de <http://se.ethz.ch/~meyer/publications/teaching/teaching-psi.pdf>
- [7] Code Combat - <https://codecombat.com/about>
- [8] Beta the game - <http://www.betathegame.com/>
- [9] Hack and Slash - <http://www.hacknslashthegame.com/>
- [10] DR Who code game - <http://www.engadget.com/2014/10/20/doctor-who-coding-game/>
- [11] Ludewing, J., Bassler, T., Deininger, M., Schneider, K., & Schwille, J. (1992). SESAM - Simulation Software Projects. Stuttgart, Alemania.
- [12] SIMSE – GUI - <http://www.ics.uci.edu/~emilyo/SimSE/>
- [13] Isong, B. (Septiembre de 2014). A Methodology for Teaching Computer Programming: first year students' perspective. Obtenido de http://www.academia.edu/9056660/A_Methodology_for_Teaching_Computer_Programming_first_year_students_perspective
- [14] Ren'Py. (2015). Ren'Py Why. Obtenido de Ren'Py: <http://renpy.org/>
- [15] Unity. (2015). unity 3D. Obtenido de unity 3D: <https://unity3d.com/es>
- [16] RPGMAKER. (2015). Obtenido de <http://www.rpgmakerweb.com/>.
- [17] EPIC Games. (2015). Unreal. Obtenido de Unreal: <https://www.unrealengine.com/>
- [18] Crytek. (2015). CryEngine. Obtenido de CryEngine: <http://cryengine.com/>
- [19] YOYO Games. (2015). Game Maker Studio. Obtenido de Yoyo Games: <http://www.yoyogames.com/studio>

- [20]Blender. (2015). Blender reference manual. Obtenido de Blender: https://www.blender.org/manual/game_engine/introduction.html
- [21]Casper Gaming, C. (18 de Julio de 2012). <http://www.rpgmakervxace.net/>. Obtenido de RPG Maker VX Ace: <http://www.rpgmakervxace.net/topic/5554-csca-achievements/>
- [22] Angustias, M., & Fernández, A. (2012). El aprendizaje semipresencial o virtual: nueva metodología de aprendizaje en Educación Superior. *Revista Latinoamericana de Ciencias Sociales, Niñez y Juventud*. Obtenido de <http://www.scielo.org.co/pdf/rllcs/v10n1/v10n1a09>
- [23] Blender. (2015). Blender reference manual. Obtenido de Blender: https://www.blender.org/manual/game_engine/introduction.html
- [24]Bourque, P., & Fairley, R. (2014). SWEBOK v 3.0. IEEE Computer society. Obtenido de <http://www.computer.org/web/swebok>
- [25]Canche, M., Leal, N., Llanes, E., & González, C. (2013). Explorando alternativas de apoyo a la educación universitaria: sesión tutorial basada en servicios móviles. Yucatán, México. Recuperado el 2013
- [26]Carlson, S. (15 de Agosto de 2003). Can Grand Theft Auto Inspire Professors? Educators Say the Virtual Worlds of Video Games Help Students Think More Broadly. Obtenido de <http://eric.ed.gov/?id=EJ676406>
- [27]Colaborador, R. (2015). Wiki RPG Maker Español. Obtenido de RPG Maker Español: <http://wiki.rpgmaker.es/ayuda/general/eventoscomunes>
- [28]Ibieta, A., Isaacs, M.-A., Hinostraza, J., Labbé, C., & Claro, M. (2013). Caracterización del uso de las TIC en jóvenes chilenos: Juegos y aprendizaje conviven en los hogares. Montevideo, Temuco, Chile. Recuperado el 2015
- [29] Jennings, J. (30 de Noviembre de 2014). Teachers re-evaluate value of video games. *The sidney morning herald*. Sidney, Australia. Obtenido de <http://www.smh.com.au/national/education/teachers-reevaluate-value-of-video-games-20141130-11jw0i.html>
- [30]Ludewing, J., Bassler, T., Deininger, M., Schneider, K., & Schwille, J. (1992). SESAM - Simulation Software Projects. Stuttgart, Alemania.

- [31] Pressman, R. (2010). *Ingeniería del software, un enfoque práctico* séptima edición. Mc Graw Hill.
- [32] Trépanier, N. (Mayo de 2014). *The Assassin's Perspective: Teaching History with Video Games*. American Historical Association. Estados Unidos. Obtenido de <http://www.historians.org/publications-and-directories/perspectives-on-history/may-2014/the-assassin%E2%80%99s-perspective>
- [33] Tsung-Yen, C., & Wei-Fan, C. (2009). *Effect of Computer-Based Video Games on Children: An Experimental Study*. Tainan, Taiwan, China. Recuperado el 2015, de http://www.ifets.info/journals/12_2/1.pdf
- [34] UNAB. (01 de junio de 2015). *Artículos UNAB: Ruta TIC*. Obtenido de UNAB Virtual: <http://www.unabvirtual.edu.co/index.php/noticias/item/1417-docentes-unab-conozcan-y-participen-en-la-nueva-ruta-tic>