

Videojuego 2D interactivo como herramienta de apoyo para la introducción a la Programación Orientada a Objetos

2D interactive video game as a support tool for the introduction to Object Oriented Programming

Praxedis Hernández Jacobo^a

Abstract:

The objective of this work is the development of a 2D videogame for the Windows desktop platform, which allows to reinforce the theoretical and practical knowledge of the student through the dosing of educational content in playful environments, as well as helping tool serve as a support and guidance for the teacher on the areas in which students may have deficiencies. The curricular approach of the system is oriented to the subject of Object Oriented Programming (OOP), of the semester of the Degree Program in Engineering in Computer Engineering, of the Superior School of Tizayuca.

Keywords:

Videogame, Oriented Object Programming, Mini-games

Resumen:

El presente trabajo tiene como propósito el desarrollo de un videojuego 2D para la plataforma de escritorio Windows, el cual permita reforzar los conocimientos teórico-prácticos del estudiante a través de la dosificación de contenido educativo en ambientes lúdicos, así mismo se espera que dicha herramienta sirva como un apoyo y guía para el docente sobre los rubros en los cuales pudiesen presentar deficiencias los estudiantes. El enfoque curricular del sistema está orientado a la asignatura de Programación Orientada a Objetos (POO) del cuarto semestre del Programa Educativo de Licenciatura en Ingeniería en Computación, de la Escuela Superior de Tizayuca.

Palabras Clave:

Videojuego, Programación Orientada a Objetos, Mini-juegos

^a Autor de Correspondencia, Universidad Autónoma del Estado de Hidalgo, Escuela Superior de Tizayuca, Email: pr308631@uaeh.edu.mx

Introducción

La búsqueda por mejorar el proceso de aprendizaje de los estudiantes durante los últimos 30 años ha motivado un proceso de investigación acerca de las bondades de la inclusión de herramientas digitales dentro del aula (Valderrama, 2011), esto con el fin de captar el interés de los alumnos. No obstante, dichos esfuerzos de articulación entre educación y tecnología se han visto limitados en la mayoría de los casos por el grado de aceptación que presentan los alumnos para con estos (Macías, 2015), es aquí donde nace el enfoque de incluir videojuegos con contenido educativo dentro de los espacios escolares como alternativa para mantener el interés del jugador, en este caso del estudiante, dentro del proceso de formación académica (López, 2015), pues proveen un espacio virtual libre de prejuicios y en la mayoría de los casos donde el progreso del participante puede verse recompensado con estímulos que detonan en expresiones y/o emociones de los receptores primarios de placer; es decir, el uso de dichas herramientas promueve un aprendizaje situado, experimental, plantea la generación de habilidades y destrezas para su resolución (Connolly, 2012), brindando así un medio de apoyo formativo para el docente así como también un forma de apoyo para el estudiante.

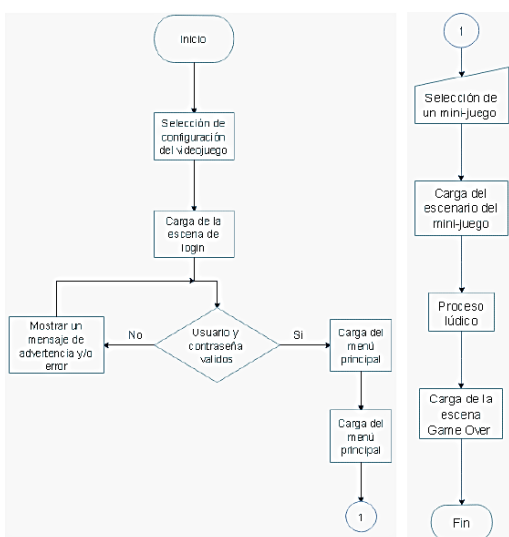


Figura 1. Diagrama general del sistema

Emplear la tecnología para apoyar el proceso de enseñanza contribuye a personalizar el proceso de aprendizaje del alumno, permitiendo a los educadores y educandos ser participantes activos en el proceso de enseñanza-aprendizaje de una forma mucho más personal e individual (Prensky, 2011), pues la tecnología proporciona herramientas para implementar nuevas formas de aprender más efectivas y reales.

Para el presente proyecto se planteó del desarrollo de cuatro mini-juegos (Buscaminas, Crucigrama, Memorama, Sopa de letras) enfocados en la generación de habilidades para el análisis de código fuente sobre los lenguajes de programación de C++, C# y Java, así como

reforzar los fundamentos propios de dicho paradigma de programación.

De manera general se puede describir el funcionamiento del sistema basándose en las etapas de este (Figura 1):

- I. Proceso de configuración para la ejecución inicial del juego (Resolución recomendada: 1024x768).
- II. Proceso de login del usuario para dar acceso al menú principal del Sistema (Menú de mini-juegos).
- III. Una vez dentro del menú principal, el programa ejecutará la carga de la escena en función del mini-juego seleccionado (buscaminas, crucigrama, memorama, sopa de letras).
- IV. Finalmente se cargará la escena de Game Over.

Mini-juego “Buscaminas”

Este mini-juego consiste en una temática similar a la del clásico juego de buscaminas, donde además de que la meta de encontrar el mayor número de bloques evitando las bombas se anexan el factor educativo. Esto es la agregación de reactivos de dos diferentes tipos (pensamiento reflexivo en código y teoría), los cuales permitirán al estudiante reforzar su dominio sobre alguno de los tres lenguajes base de los que trabaja el sistema, así como los fundamentos que involucra la POO.

Para ello se desarrolló un algoritmo de juego basado en una tabla matricial de juego de $n \times n$ dimensiones, donde a cada celda se le asigna un ítem, sea este un: bloque, bomba, vida extra, pregunta, fragmento de código; para establecer las posiciones y el número de reactivo a trabajar en tiempo de ejecución se hace uso de un algoritmo de números pseudoaleatorios, posteriormente se someten las coordenadas matriciales propuestas a su respectiva validación para garantizar que el sistema no sobrescriba datos en posiciones ya ocupadas, después se procede a la generación del escenario de juego a través de la respectiva instanciación de componentes de juego prefabricados (prefabs), se les asigna a estos los respectivos listeners y finalmente se tiene la carga de juego lista tal y como se aprecia en la siguiente figura 2.

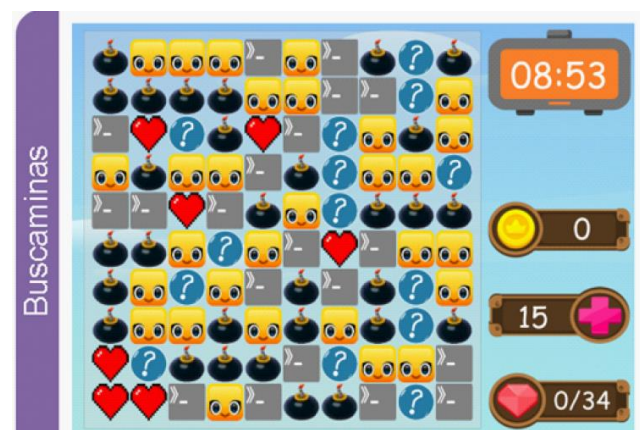


Figura 2. Captura de pantalla del mini-juego buscaminas

Mini-juego “Crucigrama”

La finalidad de este mini-juego es reforzar los conceptos clave del paradigma de POO a través de la generación de un crucigrama, donde la definición y/o idea general del punto o concepto a tratar sea la que se genere dentro del juego.

La lógica del juego se basa en la consulta a una base de datos en SQLite, sobre la cual se recuperan n cantidad de registros de forma pseudoaleatoria, una vez recuperados, la palabra clave se somete a un proceso de tratamiento de datos para eliminar caracteres no reconocidos por el sistema (ejem. Números, caracteres de acentuación, etc.), seguido de esto se realiza un proceso de ordenamiento de las palabras y se procede a la adecuación de estas en las posiciones horizontal y vertical, donde cada nueva iteración deberá de generar una posición diferente a la anterior buscando la igualdad de caracteres clave entre palabras a la par que respeta espacios de límite entre las mismas para evitar generar secuencias de orden poco claras; dando como resultado un crucigrama como el que se muestra en la figura 3.

Es importante recalcar que la generación y la disposición de las palabras a identificar dentro del juego son generadas de forma dinámica, es decir, dependiendo de la cantidad de elementos o tamaño de la pantalla, es la cantidad de conceptos o palabras correspondientes al estudio de la POO que generarán dentro del minijuego.

Lo anterior es el resultado de haber contado con un grupo piloto durante el desarrollo de los minijuegos, del cual se obtuvieron comentarios y observaciones que enriquecieron el trabajo generado en el presente desarrollo de los minijuegos.

Mini-juego “Memorama”

El juego consiste en encontrar pares de cartas que contengan la misma figura tal como se muestra en la figura 4.

La operación de este mini-juego consiste en la validación del posicionamiento de las cartas dentro del escenario. La validación del minijuego se realiza a través de un algoritmo de posicionamiento matricial.



Figura 3. Captura de pantalla del mini-juego crucigrama

La disposición de las cartas se hace de manera aleatoria, por lo que cada vez que se cargue el mini-juego la distribución de las cartas siempre será diferente.



Figura 4. Captura de pantalla del mini-juego memorama

Mini-juego “Sopa de letras”

Este mini-juego está basado en la validación del posicionamiento de las palabras dentro de un escenario en el cual se validan la dirección, la comprobación de carácter por carácter dentro de un espacio matricial de tal manera que el juego resulta ser sencillo en su operación tal como se muestra en la figura 5.

Cabe mencionar que al igual que los demás mini-juegos, la disposición de las palabras se realiza de manera dinámica, por lo que por cada inicio del mini-juego se encontrará una distribución distinta de las palabras dentro de la sopa de letras.

La consulta de las palabras a mostrar en el mini-juego se hace a través del acceso a una base de datos que puede ser alimentada de manera manual.

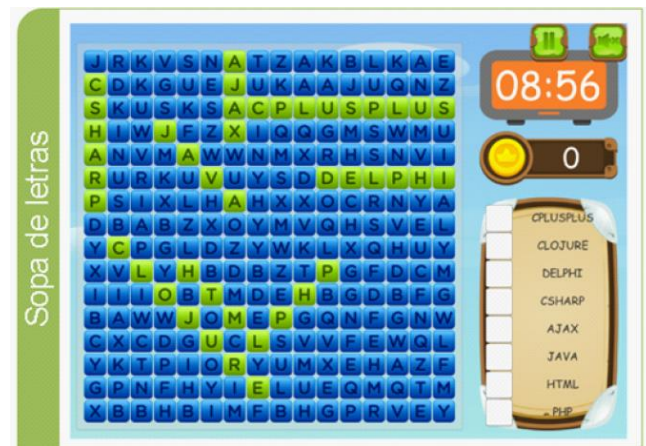


Figura 5. Captura de pantalla del mini-juego Sopa de Letras

Conclusiones

El proceso de pilotaje tomo alrededor de un mes, distribuido en seis sesiones de aproximadamente una hora de duración, donde de manera general se llevó a cabo la evaluación de los cuatro mini-juegos en su conjunto.

En la figura 6 se muestran las observaciones y/o recomendaciones hechos por el grupo piloto con el objetivo de obtener datos cualitativos que permitieran retroalimentar el trabajo realizado.

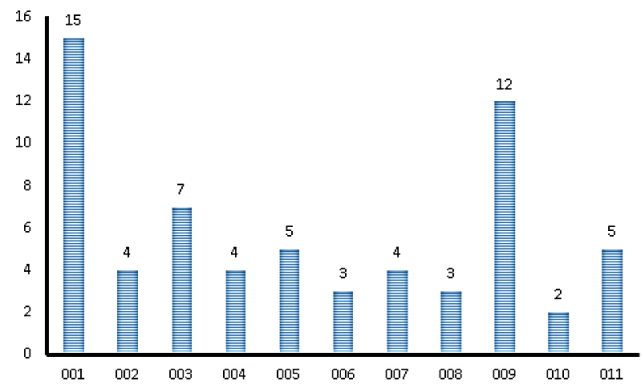
Por otro lado, los errores obtenidos con sus códigos y la frecuencia con la que ocurrieron durante la última fase de pilotaje se muestran en la figura 7, los cuales fueron atendidos para obtener la última versión de los mini-juegos.

El impacto generado en el grupo piloto, deja en evidencia la necesidad de una articulación íntegra y detallada enfocada con los estudiantes, esto principalmente para solventar las deficiencias vistas durante la fase de pilotaje del sistema; con base a este último punto existe una necesidad de reafirmar los fundamentos del paradigma de POO, para que así los estudiantes tengan la suficiente habilidad y destreza para la generación de programas en cualquiera de los tres lenguajes propuestos en el juego, en conjunto con el papel motivacional de los alumnos para que así se pueda obtener un impacto positivo en su rendimiento académico.

Observaciones/Recomendaciones	
1	Mejorar la UI
2	Mejorar la identificación de las pistas en el Crucigrama
3	Mejorar el temporizador
4	Manejar número de intentos máximo para la Sopa de letras
5	Limitar el tiempo entre clics consecutivos (Sopa de letras)
6	Mostrar la totalidad de una palabra al encontrar algún carácter de esta
7	Reducir la cantidad de tiempo para cada mini-juego

Figura 6. Observaciones obtenidas de la operación de los mini-juegos por el grupo piloto.

Es por ello que la generación de las versiones obtenidas de cada mini-juego, se sustentó con la opinión vertida de cada uno de los estudiantes que conformaron al grupo piloto. De tal manera que representara la experiencia vivida por el usuario, el elemento esencial para la conformación de cada mini-juego, basados siempre en la opinión real de grupo piloto de prueba.



Código de error	Descripción
001	Errores en el puntaje
002	Problemas durante la carga inicial del videojuego
003	Errores en la generación de eventos al presionar botones
004	Errores al iniciar una nueva partida de cualquier mini-juego
005	Errores del temporizador
006	Error en la funcionalidad para resolver algún mini-juego
007	Errores en la carga de las palabras (Sopa de letras)
008	Errores en el ingreso de datos (Crucigrama)
009	Errores en la carga de la imagen de las cartas (Memorama)
010	Errores en la visualización de los campos de texto (Crucigrama)
011	Errores en la visualización del escenario de juego

Figura 7. Frecuencias de errores y códigos de error obtenidos de los mini-juegos.

Finalmente, a manera de conclusión es necesario que los alumnos adquieran una motivación propia para aprender el paradigma de POO en cualquiera de sus diferentes lenguajes, aunado a la constante practica de sus habilidades a través de la generación y análisis de programas que contribuyan a obtener un impacto positivo en su rendimiento académico.

Referencias

- Anna Anthropy, N. C. (2014). *A Game Design Vocabulary*. New York: Pearson Education, Inc..
- Asensio, J. (2014). Progame: aprendiendo a programar usando videojuegos como metáfora para visualización de código. *ReVisión*, VII. Retrieved Febrero 14, 2017, from [http://www.aenui.net/ojs/index.php?journal=revisión&page=article&op=viewArticle&path%5B%5D=Balerdi, F. E. \(2014, Octubre 05\). Videojuegos y educación. Retrieved Febrero 13, 2017, from Universidad del País Vasco: https://campus.usal.es/~teoriaeducacion/rev_numero_02/n2_art_etxeberria.htm](http://www.aenui.net/ojs/index.php?journal=revisión&page=article&op=viewArticle&path%5B%5D=Balerdi, F. E. (2014, Octubre 05). Videojuegos y educación. Retrieved Febrero 13, 2017, from Universidad del País Vasco: https://campus.usal.es/~teoriaeducacion/rev_numero_02/n2_art_etxeberria.htm)
- Barinaga, B. L. (2010). *Juego: Historia, Teoría y Práctica del Diseño Conceptual de Videojuegos*. Madrid: Alesia Games & Studies.
- BBVA Innovation Center. (2012, Junio). *Gami...ación El negocio de la diversión*. BBVA Innovation Edge(3). Retrieved Febrero 26, 2016, from https://www.centrodeinnovacionbbva.com/sites/default/.../gami...acion_spanish.pdf
- Byl, P. d. (2012). *Holistic Game Development with Unity*. New York: Elsevier, Inc.
- Connolly, T. (2012). *A systematic literature review of empirical evidence on computer games and serious games*. Pergamon.
- David R. Michael, S. C. (2006). *Serious Games: Games that Educate, Train and Inform*. Boston, Massachusetts: Thomson Course Technology.

- De Carvalho, C. V. (2015). *Serious Games, Interaction, and Simulation: 5th International Conference*. Novedrate, Italy: Springer.
- Edel Navarro, R. (2003). El rendimiento académico: concepto, investigación y desarrollo. REICE. Revista Iberoamericana sobre Calidad, Eficacia y Cambio en Educación. Retrieved Marzo 24, 2017, from <http://www.redalyc.org/articulo.oa?id=55110208>
- Fernandez, D. V. (2011). *Desarrollo de Videojuegos: Arquitectura del Motor de Videojuegos (Vol. I). Cursos en Español*.
- Fullerton, T. (2014). *Game design workshop: a playcentric approach to creating innovative games (Tercera ed.)*. New York: CRC press.
- González Seguí, H. O. (2005). Veinticinco años de videojuegos en México. Las mercancías tecnoculturales y la globalización económica. *Comunicación y Sociedad*, 38.
- López Raventós, C., & Belli, S. (2008). Breve historia de los videojuegos. *Athenea Digital. Revista de Pensamiento e Investigación Social*, 159-179. Retrieved from <http://www.redalyc.org/articulo.oa?id=53701409>
- López, I. (2015, Agosto 3). *Juegos Serios En El Aprendizaje*. Retrieved from <http://ventanales.casagrande.edu.ec/biblioteca/juegos-serios-en-el-aprendizaje/>
- Macías, V. (2015, Octubre 13). *UNAM y la industria del videojuego en México*. Retrieved from <http://www.fundacionunam.org.mx/vanguardia-unam/unam-y-la-industria-del-videojuego-en-mexico/>
- Martín, C. M. (2012, Diciembre 4). Dialnet. Retrieved from *Actas IV Congreso Internacional Latina de Comunicación Social*: <https://dialnet.unirioja.es/servlet/libro?codigo=514729>
- Menárguez, A. T. (2016, Julio 18). El cerebro necesita emocionarse para aprender. Retrieved Noviembre 06, 2016, from *El País* http://economia.elpais.com/economia/2016/07/17/actualidad/1468776267_359871.html
- Moraldó, H. H. (2009). Desafíos y tendencias en el diseño de videojuegos. *Revista de Comunicación de la Universidad de Sevilla*, 15.
- Prensky, M. (2011). Enseñar a nativos digitales. In M. Prensky, *Enseñar a nativos digitales* (pp. 1-47). Barcelona: Grupo SM.
- Sara de Freitas, P. M. (2011). *Digital Games and Learning*. Reino Unido: Blooms-bury Publishing.
- Schel, J. (2015). *The Art of Game Desing A Book of Lenses*. Pittsburgh, Penn-sylvania, USA : CRC Press.
- Scolari, C. A. (2013). *Homo videoludens 2.0. De Pacman al la gamication*. Barcelona: Laboratori de Mitjans Interactius.
- Severin, E. (2016). *Tecnologías digitales al servicio de la calidad educativa*. Chile: Oficina Regional de Educación para América Latina y el Caribe.
- Sylvester, T. (2013). *Designing Games A Guide to Engineering Experiences*. United States of America.: O'Reilly Media, Inc.
- Unity Technologies. (2017). *Requisitos del sistema para la Unity versión*. Retrieved Marzo 29, 2017, from Unity: <https://unity3d.com/es/unity/system-requirements>
- Universidad Marista de Mérida. (2017). *Proceso de Enseñanza Aprendizaje*. Retrieved Marzo 28, 2017, from Universidad Marista de Mérida: <http://www.marista.edu.mx/p/6/proceso-de-ensenanza-aprendizaje>
- Valderrama-Ramos, J. A. (2011, Agosto). *Videojuegos y educación: explorando aprendizajes entre adolescentes*. Retrieved from *Repositorio Institucional del ITESO*: <http://rei.iteso.mx/handle/11117/2403>