Videojuego educativo con enfoque en química

El reactor de Aether

*Educational videogame with a focus on chemestry*

*The Aether’s Reactor*

Uriel Alejandro Villegas Cuevas

Docente IPN - UPIIZ

Zacatecas, México

uvillegas@ipn.mx

Arantxa Fuensanta Arana Torres

Alumna IPN - UPIIZ

Zacatecas, México

aaranat1500@alumno.ipn.mx

Efraín Arredondo Morales

Docente IPN - UPIIZ

Zacatecas, México

earredondo@ipn.mx

Lucía Montserrat López Méndez

Alumna IPN - UPIIZ

Zacatecas, México

llopez1809@alumno.ipn.mx

Introducción

Este artículo presenta el proyecto de un videojuego educativo enfocado a apoyar el estudio de la tabla periódica para alumnos de los primeros niveles de la educación preparatoria. Se decidió crear un software multimedia interactivo con bases pedagógicas que, mediante el juego, contribuya al aprendizaje de las ciencias naturales, en especial la tabla periódica para la materia de química, durante el primer año de preparatoria. Se espera que sea una herramienta de enseñanza útil y de calidad para docentes y alumnos, y que genere en los adolescentes un mayor interés y curiosidad por aprender más acerca de tópicos científicos, proporcionándoles bases sólidas de conocimiento que les sirvan para explicar los procesos naturales en el mundo real y poder relacionar conceptos científicos con otras disciplinas.

Palabras Clave - Bachillerato, química, software educativo, videojuego, preparatoria, tabla periódica.

ABSTRACT

This article presents an educational video game focused on supporting the periodic table study for students in the first levels of high school. An interactive multimedia software with pedagogical foundations was created to contribute to the learning of natural sciences through the game, especially to the periodic table for the Chemistry subject during the first year of high school. It is expected to be a useful and quality teaching tool for teachers and students and to generate in adolescents greater interest and curiosity for learning more about scientific topics by giving them solid foundations of knowledge that will help them to explain the natural processes in the real world so they can relate scientific concepts with other disciplines.

Keywords - high school, Chemistry, educational software, video game, high school, periodic table.

Contexto

El área de investigación se centra en un videojuego educativo tipo juego de rol (Rol Playing Game o RPG por sus siglas en inglés) que por medio de diferentes actividades se van reconociendo y aprendiendo los elementos de la tabla periódica; este trabajo está coordinado por docentes de la Unidad Profesional Interdisciplinaria de Ingeniería Campus Zacatecas del Instituto Politécnico Nacional (UPIIZ-IPN).

El videojuego busca apoyar a los estudiantes de primer semestre de preparatoria a comprender y hacer más amena la experiencia de aprender las bases de la química, en este caso de la tabla periódica.

Introducción

En este artículo se presenta el proyecto de un videojuego RPG, (Rol Playing Game, por sus siglas en inglés) llamado: El Reactor de Aether, que se basa en encontrar los elementos de la tabla periódica mientras se combate contra monstruos y se resuelven acertijos planteados para ayudar y repasar temas de química, particularmente de primer semestre de preparatoria.

El videojuego fue trabajado con la metodología cascada, que permite un enfoque sistémico y secuencial, además de desarrollar el videojuego con tecnología Unity y software especializado en arte de píxeles; lo que permitió diseñar a los personajes y el entorno para poder lograr el objetivo principal del software que era apoyar a los alumnos de primer semestre de preparatoria en el conocimiento de la tabla periódica. La herramienta se programó para dispositivos móviles con sistema operativo Android.

1. OBJETIVOS

*A.- Objetivo general*

Diseñar y programar un software interactivo como herramienta de apoyo al aprendizaje de los principales tópicos relacionados a la tabla periódica.

*B.- Objetivos particulares*

* Diseñar, programar e integrar componentes generales y específicos del software.
* Diseñar y programar mecánicas de juego que se integren con la historia y los temas a tratar.
* Implementar esta herramienta en dispositivos móviles con Sistema Operativo Android.

1. Metodología

La metodología en cascada fue de las primeras en orientarse al desarrollo de software. En esta metodología, se sigue un orden riguroso que impide el avance la siguiente fase hasta que la anterior haya sido completada. Los resultados no se hacen evidentes hasta el final del ciclo de la metodología. A diferencia de las metodologías ágiles, en la mayoría de las etapas de la metodología en cascada, el cliente no está presente [1].

Según Sommervile, en la metodología cascada se deben terminar cada una de las fases antes de empezar una nueva, lo que hace que el trabajo sea sistémico y secuencial, justo como se muestra en la figura 1.

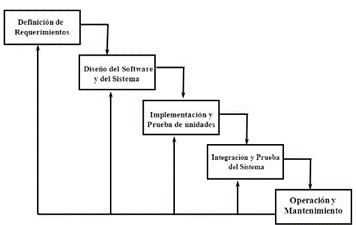


Figura 1. La modelo cascada usado en el proyecto de videojuego de química “El reactor de Aether”

Además, la metodología cascada permite regresar a fases anteriores si hubiera lo necesidad de hacer cambios o modificaciones al proyecto; la metodología cascada cuenta con cinco fases que permiten definir el rumbo del proyecto [1] [2], como se puede apreciar en la tabla 1.

Tabla I. Fases de la metodología cascada

|  |  |
| --- | --- |
| **Fases** | **Descripción** |
| Análisis y definición de requerimientos | El equipo de desarrolladores junto con el cliente se definen los servicios, restricciones, metas y limitantes del proyecto. |
| Diseño | Se hacen los diseños con base a los requerimientos, ya sea de arquitectura de datos, interfaz, entre otras |
| Implementación | Se programan con base a los diseños planteados, se revisa constantemente estos diseños para ver si se cumple con los solicitado. |
| Pruebas | Se prueba el sistema completo para asegurar que se cumplen con todos los requerimientos |
| Mantenimiento | Se laboran manuales con base al funcionamiento de del proyecto, se buscan posibles errores que no se hallan contemplado. |

1. Análisis y definición de requerimientos

En la fase de análisis y la definición de los requerimientos, se analizaron dos metodologías principalmente, la metodología SUM y la de cascada; sin embargo, después de analizar y hacer una prueba con SUM, se tomó la decisión de usar Cascada, principalmente por las ventajas antes mencionadas, además, el equipo de trabajo tenía más experiencia trabajando con esta metodología [2].

1. Requerimientos

Una vez definida la metodología se desarrolló un diagrama conceptual de los elementos que integran el videojuego, es decir, los escenarios, personajes, enemigos y todas las herramientas que permiten que sea interactivo con el usuario [2], como se puede ver en la figura 2.

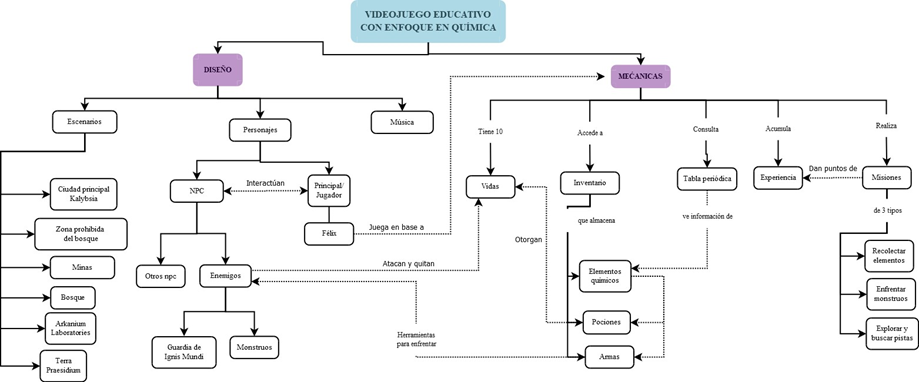


Figura 2 Diagrama conceptual del proyecto “El reactor de Aether”

Después de platicar con el cliente se definieron un total de 14 requerimientos funcionales y 9 no funcionales, como se puede observar la tabla 2.

Tabla II. Requerimientos funcionales y no funcionales

|  |  |
| --- | --- |
| **Requerimientos** | **Estado** |
| RF\_01: Implementación de temas de química | Satisfactorio |
| RF\_02: Nivel de experiencia. | Satisfactorio |
| RF\_03: Dispositivo móvil | Satisfactorio |
| RF\_04: Controles | Satisfactorio |
| Rf\_05: Personalización | Satisfactorio |
| RF\_06: Punto de guardado | Satisfactorio |
| RF\_07: Historia del héroe | Satisfactorio |
| RF\_08: Historia del juego. | Satisfactorio |
| RF\_09: Misiones - Bosque | Satisfactorio |
| RF\_10: Misiones - Kalybsia. | Satisfactorio |
| RF\_11: Misiones - mina | Satisfactorio |
| RF\_12: Misiones - zona prohibida del bosque. | Satisfactorio |
| RF\_13: Misiones - Laboratorios Arkanium. | Satisfactorio |
| RF\_14: Misión final | Satisfactorio |
| RNF\_01: Tiempo de carga | Satisfactorio |
| RNF\_02: Peso del juego | Satisfactorio |
| RNF\_03: Mantenimiento | Satisfactorio |
| RNF\_04: Temas legales | Satisfactorio |
| RNF\_05: Escalabilidad | Satisfactorio |
| RNF\_06: Jugabilidad | Satisfactorio |
| RNF\_07: Confiablidad | Satisfactorio |
| RNF\_08: Frecuencia | Satisfactorio |
| RNF\_09: Ambientación sonora | Satisfactorio |

1. DISEÑO

En esta fase, se hizo el diseño de los diferentes diagramas que constituyen el proyecto. Estos diagramas servirán como guía para las siguientes fases a la hora de implementarlos.

* 1. Diseño de arquitectura

La arquitectura del “Videojuego educativo con enfoque en química” se decidió el uso de una arquitectura en capas [4]; cuyo funcionamiento se basa en la separación de distintas funcionalidades del sistema en capas, donde cada una de estas se encarga de tareas en específico [5] [6].

La aplicación de esta arquitectura para el proyecto correspondiente a “Trabajo terminal” constará de 3 capas: Capa principal del juego, Capa de personajes, Capa de interacción con el mundo.

La arquitectura del sistema se puede ver en la figura 3

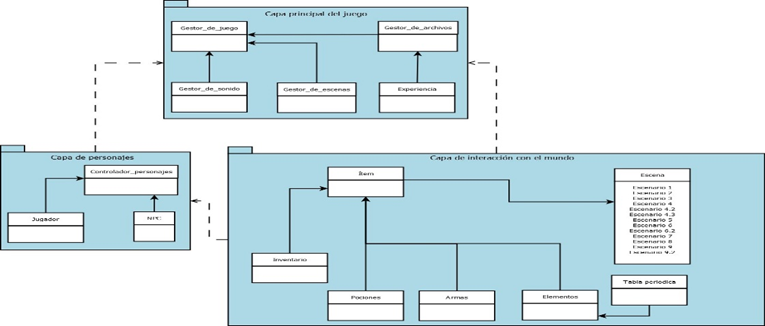


Figura 3 Arquitectura del sistema

* 1. Diagrama de clases

Cada una de clases corresponde a los scripts generados para el proyecto, para una mejor organización se decidió separar los archivos y clases de acuerdo con la función que realizan por lo que hay clases enteramente dedicadas a controlar aspectos solo como la personalización, misiones, movimiento de los enemigos, características del jugador como nivel de vida, experiencia, movimiento, ataques, entre otras [7].

A su vez, una gran cantidad de estas clases están conectadas a “DatosPartida” ya que esta sirve para guardar todos los datos que el usuario haya generado durante la sesión y que al salir de la partida los datos no se pierdan y se pueda continuar con el juego normalmente [2]. Debido al tamaño del diagrama, además de incluirlo en la siguiente página, también se ha cargado en una carpeta de OneDrive, de modo de que cada elemento sea fácilmente distinguible.

En el siguiente enlace se encuentra disponible el diagrama de clases para su mejor visualización: [Link de Diagrama de Clases](https://correoipn-my.sharepoint.com/:i:/g/personal/aaranat1500_alumno_ipn_mx/EVKj00OOk5RKnSETeNMRDOoBbPMPL_Y_7N04NFdUxWvBmw?e=dS4Vl6)

* 1. Diagrama de actividades

Los diagramas de actividades se utilizan para ver de manera más clara el proceso que se sigue al realizar una actividad [8] [9]. En la siguiente figura se muestra un de los diagramas que tiene que ver con la acción de equipar armas del personaje:

1. El avatar del usuario debe estar en la pantalla de inventario.
2. Se da click en el escudo o el arma a equipar.
3. El sistema regresa información relacionada con item del arma o escudo y lo muestra al usuario.
4. El usuario da click para usar el arma o escudo.
5. El sistema reproduce el sonido de “equipar objeto”.
6. Mueve el objeto a su casilla correspondiente.
7. Guarda el objeto en datos del jugador.

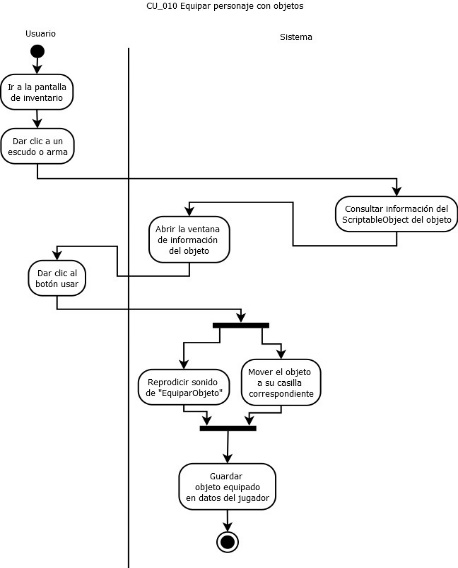


Figura 4 Diagrama de actividad “equipar arma”

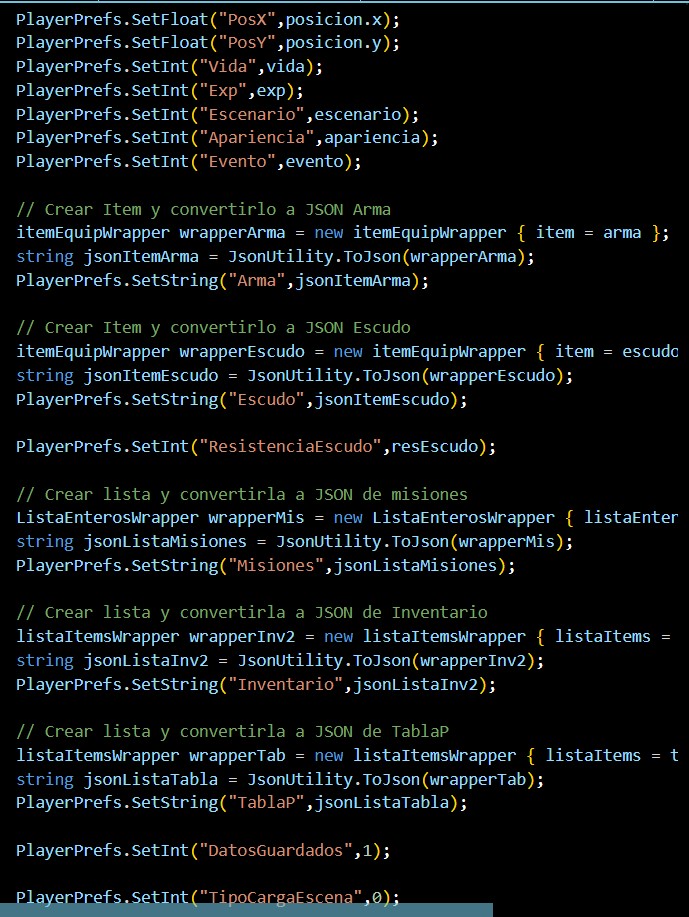
*D*. *Diagrama de casos de uso*

Figura 6 Código para guardar datos utilizando PlayerPrefs

Los diagramas de caso de uso se utilizan para poder visualizar la interacción del usuario con el sistema, a continuación, se muestra el diagrama de caso de uso del sistema, donde se pueden observar todas las acciones que podrá realizar el jugador dentro de éste [10] [11]. Esto se puede observar en la figura 5



Figura 5 Diagrama de caso de uso

* 1. Manejo de archivos

Para el almacenamiento de datos importantes para el óptimo funcionamiento del videojuego se decidió hacer uso de los PlayerPrefs proporcionados por Unity, estos constan de un archivo con extensión “.xml” el cual puede crear, leer y editar. En este archivo se almacenarán datos como:

* Progreso del jugador durante la partida correspondiente: posición en la que estaba antes de salir del juego, misiones completadas, elementos y objetos recolectados, puntos de experiencia, vida, y elementos acomodados de forma correcta en la tabla periódica.
* El número de partida.
* Apariencia personalizada del jugador.

A continuación, se mostrará una parte del código destinada a guardar los datos de la partida con los PlayersPrefs. En esta se observa a los playerPrefs trabajando principalmente con tres tipos de datos, int, string y float. Lo cual es muy útil al momento de guardar datos sencillos, como fue el caso de las variables posición, vida, experiencia, escenario, apariencia y evento.

Sin embargo, el guardado se vuelve más complejo al querer guardar datos distintos de los 3 mencionados anteriormente.

Para estos casos, existen varios trucos para aun así poder utilizar los PlayerPrefs, como se observa, en el siguiente caso, donde para poder guardar variables de tipo Item o List<x>, primero se convertían en un JSON.

* 1. Prototipos

Un prototipo es un primer modelo que sirve como representación de un producto, este permite ver el diseño y la funcionalidad del producto [12]. A continuación, se muestra una pantalla de los prototipos, la cual corresponde al inventario, en esta se pueden observar los ítems que podrá recoger el jugador a lo largo del juego, así como la apariencia del personaje principal [13]. En esta vista también se encuentran los cuadros de equipamiento, donde estarán las armas que el jugador decida usar.



Figura 7 Prototipo de la ventana de inventario



Figura 8 Diseño de las posibles personificaciones del avatar del jugador

1. Implementación

En esta fase se realizaron diversas tareas relacionadas con el modelado y la implementación del proyecto. Al ejecutar el software, la primera escena en aparecer será el menú de inicio, que contiene cuatro botones: juego nuevo, ajustes, continuar juego y salir. Al posicionar el ratón sobre un botón o darle clic estos cambiarán de color para mostrar que han sido seleccionados (figura 9).

Si el usuario comienza un juego nuevo, la siguiente pantalla en aparecer será la de personalización en donde hay 3 opciones de estilo de cabello y 3 opciones de color de ojos, al seleccionar las opciones deseadas, la apariencia del avatar de la izquierda cambia de acuerdo con la elección del usuario (figura 10).



Figura 9 Menu de inicio



Figura 10 Pantalla de personificación

El videojuego se modeló para que tuviera el aspecto de un juego de aventuras y acertijos centrado en reforzar la comprensión de la tabla periódica, por lo que se diseñó una tabla periódica que se iría llenando conforme se encuentren los elementos químicos en el videojuego.



Figura 11 Tabla periódica del videojuego

1. Pruebas

Cuando se llegó a esta fase se realizaron las pruebas pertinentes para validar y revisar que el funcionamiento del videojuego fuera correcto, aunque durante el desarrollo del proyecto se acotaron varios elementos como fueron:

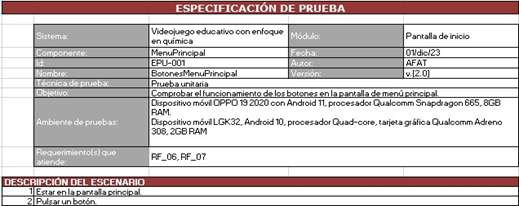
* + 1. Reducción de alcance: Se redujeron sustancialmente las funcionalidades, escenarios y módulos del proyecto, lo que resultó en una disminución significativa en el número total de pruebas que debían llevarse a cabo.
    2. Agrupación de escenarios como módulos: Durante el análisis del documento, se identificó que cada escenario podía considerarse como un módulo independiente. Esto llevó a la decisión de agrupar todas las pruebas posibles relacionadas con un escenario específico, abordando aspectos como el movimiento, la interacción con diferentes elementos del juego, combates, peleas, y otras funcionalidades específicas.

Estas adaptaciones en el plan de pruebas no solo reflejan una respuesta eficiente a los cambios en el proyecto, sino que también buscan optimizar los recursos al enfocarse en pruebas relevantes y necesarias. La agrupación de escenarios como módulos y la eliminación de pruebas redundantes contribuyen a una estrategia de pruebas más efectiva y centrada en los objetivos del proyecto.

Tabla III Resumen de tipo de pruebas y cantidad

| Tipo de prueba | **Cantidad de pruebas** |
| --- | --- |
| Pruebas unitarias | 14 |
| Pruebas de integración | 3 |
| Pruebas de sistema | 4 |

A continuación, se muestra un ejemplo de especificación de prueba utilizada para probar los botones del menú inicial.



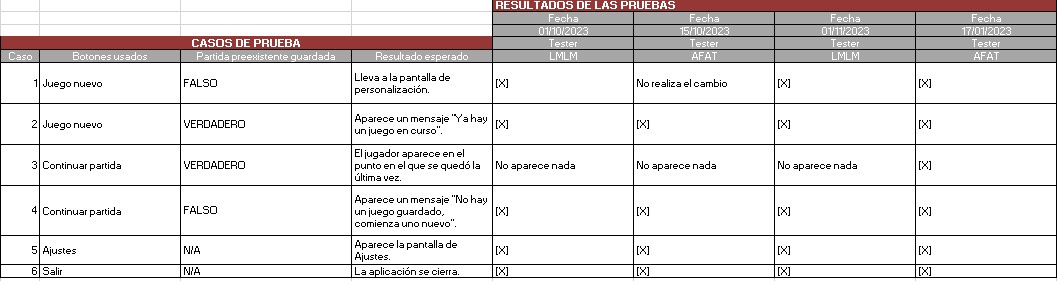
Figura 12 Ejemplo de prueba unitaria

Figura 13 Resultados de la prueba unitaria

1. Mantenimiento

Durante la fase de mantenimiento del proyecto “El Reactor de Aehter”, se desarrollaron tres manuales para brindar información y guía sobre el uso del software; el objetivo de estos manuales era asegurar el correcto funcionamiento y continuidad de las herramientas del videojuego lo que proporciona la información necesaria tanto para los usuarios finales como para los desarrolladores.

El Manual de Usuario fue diseñado para guiar a los usuarios en el uso adecuado de las herramientas, ofreciendo una explicación detallada sobre el funcionamiento de cada elemento en las diferentes escenas.

Por otro lado, el Manual Técnico proporcionó una visión interna del funcionamiento interno de la aplicación (videojuego orientado hacia la química). Se incluyeron detalles sobre los requisitos técnicos de la computadora necesaria para ejecutar el proyecto y una descripción detallada de cómo funcionan las diferentes partes y componentes de la aplicación.

Además, se creó el Manual de Mantenimiento, que tenía como objetivo facilitar las futuras actualizaciones y mejoras, con instrucciones detalladas para modificar el proyecto.

Durante el mantenimiento, el equipo a cargo corrigió errores identificados en la interfaz de usuario para mejorar la experiencia del usuario. Además, se realizaron actualizaciones de los modelos para reflejar los cambios en el entorno del campus universitario, para asegurar que la información se mantenga actualizada.

Los manuales desarrollados proporcionan la información necesaria para el uso adecuado de las herramientas, permitiendo su implementación en otros dispositivos y facilitando futuras actualizaciones y mejoras.

1. Conclusiones

Como conclusiones se puede decir que este proyecto demandó una dedicación superior en comparación con otros proyectos, representó una de las fuentes más enriquecedoras de conocimiento. El aprendizaje abarca varias etapas, desde la aplicación de metodologías y creación de diagramas UML hasta la gestión de recursos financieros y temporales en un proyecto, e incluso la adopción de tecnologías completamente nuevas.

El aspecto técnico del proyecto resulta desafiante debido a la diversidad de proyectos y el uso frecuente de tecnologías nuevas. Los desarrolladores deben cultivar habilidades autodidactas, aprender a resolver problemas incluso en situaciones adversas y reconocer sus limitaciones antes de asumir nuevos retos.

Afrontar el desafío de aprender una nueva tecnología, como en el caso de Unity, otorgó a ambos integrantes el control autónomo sobre su proceso de aprendizaje. La complejidad del proyecto, desde el diseño hasta la programación, resalta la importancia de revisar minuciosamente cada detalle, dado que cualquier modificación, por mínima que sea, puede impactar significativamente la funcionalidad del producto final.

Entre las tecnologías dominadas, se destacan el uso de software especializado para arte en píxeles, la programación en Unity y el manejo de GitHub. Este último se utilizó como repositorio principal, proporcionando valiosas lecciones sobre su creación, clonación, actualización y la gestión de ramas para almacenar distintas versiones del proyecto.

Las posibilidades de mejora para el proyecto son extensas, y se planea implementar funciones descartadas por limitaciones de tiempo, como la adición de sonidos ambientales, escenarios, funciones de creación y descomposición de ítems, entre otras.

Agradecimientos

Quisiéramos expresar nuestro más sincero agradecimiento al maestro Efraín Arredondo Morales, quien ha sido un mentor invaluable en el desarrollo de este proyecto científico.

También deseamos extender nuestro agradecimiento al Doctor Uriel Alejandro Villegas Cuevas, quien brindó su apoyo constante a lo largo del transcurso del desarrollo de este proyecto.

Por último, pero no menos importante, quisiéramos expresar nuestro agradecimiento a todos los compañeros de equipo que participaron en este proyecto.

Referências Bibliográfica

|  |  |
| --- | --- |
| [1] | I. SOMMERVILE, Ingeniería de Software, México: Pearson , 2011. |
| [2] | J. AGUIRRE-BARRERA y S. AGUIRRE-BARRERA, «Metodologías para el desarrollo de Proyectos,» *Administración de empresas,* vol. 1, nº 1, pp. 1-15, 2020. |
| [3] | J. ZUMBA-GAMBOA, «Evolución de las Metodologías y Modelos utilizados en el Desarrollo de Software,» *INNOVA Research Gamboa,* vol. 3, nº 10, pp. 20-33, 2018. |
| [4] | L. CASTRO, Arquitectura Software, Coruña: Mads Group, 2015. |
| [5] | F. GÓMEZ, V. LOLY, P. MORENO y R. TOMAS, «Propuesta de modelo en cinco capas para aplicaciones web,» *SABER,* vol. 26, nº 2, pp. 168-173, 2014. |
| [6] | H. CERVANTES-MACEDA, P. VELASCO-ELIZONDO y L. CASTRO-CAREAGA, Arquitectura de software, México: CENGAGE Learning, 2016. |
| [7] | C. VIDAL y S. RIVERO, «Propuesta y Aplicación de Diagramas de Clases UML JPI,» *Centro de Informacióin Tecnológica,* vol. 25, nº 5, 2014. |
| [8] | M. S. TABARES, J. D. PINEDA y A. F. BARERA, «Un patrón de interacción entre diagramas de actividades UML y sistemas Workflow,» *Revista EIA,* nº 10, pp. 105-120, 2008. |
| [9] | Z. CATALDI, F. LAGE, R. PESSACQ y R. GARCIA MARTÍNEZ, «Ingeniería de software educativo,» de *Congreso Internacional de Ingeniería Informática*, Buenos Aires, 1999. |
| [10] | M. VEGA, Casos de uso, 2010. |
| [11] | D. GUTIERREZ, Casos de uso Diagramas de Casos de Uso, Caracas: Iniversidad de los Andes, 2011. |
| [12] | F. ANGELES-ANGELES, «Prototipo,» *Con-Ciencia Boletín Científico de la Escuela Preparatoria No. 3,* vol. 13, pp. 33-34, 2020. |
| [13] | S. A. JUNCA-VALERO, «Desarrollo de un prototipo de videojuego educativo para educación básica, en base un estudio educativo de videojuegos,» Universidad Autónoma de Bucaramanga, Bucaramanga, 2013. |