



le Ingeni
rsitaria Minuto c



Procesos y técnicas de ingeniería de Software para la Modelación de Videojuegos

Álvaro Iván Jiménez A.
Diana Patricia Rico L.
Carlos Mauricio Méndez S.
Stephany Ceron S.
Steven Palechor Betancourt

Recibido el 20 de octubre de 2010. Aprobado el 26 de noviembre de 2010

Resumen

El equipo de ingeniería del software perteneciente al semillero de investigación Ayni de la Fundación Universitaria Católica Lumen Gentium tiene como intención principal modelar el videojuego educativo "Prende Amor X Cali". Para lograr dicho cometido se decidió establecer un estado del arte en el cual se buscó estudiar las diferentes metodologías existentes en el tema ingeniería de software educativo para videojuegos, con el fin de determinar la manera más adecuada de modelar la base estructural del videojuego.

Palabras clave

Videojuego, modelamiento, ingeniería de software educativo.

Abstract

The software engineering equip belonging to the hotbed of investigation Ayni Fundación Universitaria Católica Lumen Gentium is primarily intended to model the educational videogame "Prende Amor X Cali". To accomplish this task it was decided to establish a state of the art in which we studied the different methodologies in the field of educational software engineering for videogames in order to determine the most appropriate way to model the structural basis of the game.

Keywords

Videogame, modeling, educational software engineering.

I. Introducción

El desarrollo de las tecnologías de entretenimiento en Colombia se puede considerar deficiente, en especial en lo relacionado al tema de los videojuegos educativos; este campo es uno de los más exigentes en el ámbito de la ingeniería del software. Algunas de las causas del estancamiento tecnológico en este campo han sido la ausencia de la actitud y rigor investigativo, el conocimiento técnico y falta de entidades e instituciones dispuestas a invertir recursos en dichos proyectos.

En la actualidad, la interacción constante con la tecnología, posibilita distintos y nuevos modos atractivos de incentivar su uso entre los niños. El empleo de un videojuego educativo se ha convertido en una herramienta que integra las tecnologías de información y comunicación con los procesos pedagógicos, “el uso de ambientes educativos debe propiciar la generación de motivación intrínseca en los usuarios, para lograr un efecto duradero en el proceso de enseñanza aprendizaje” (Gómez, Galvis & Mariño, 1991)

“El uso de las tecnologías de información en la educación pretende demostrar que es necesario su aprovechamiento en la formación de los niños y jóvenes de esta época. Y la escuela y los docentes deben prepararse cada vez mejor para esa tarea” (Lacueva, 2000)

El videojuego “Prende Amor X Cali” es un proyecto de investigación en curso del semillero de investigación AYNi de la Fundación Universitaria Católica Lumen Gentium - UNICATOLICA, el cual adelanta trabajo desde el año 2009. La intención principal del equipo de ingeniería de software del semillero fue modelar la estructura funcional del videojuego “Prende Amor X Cali”, a partir del estudio de una serie de metodologías y aplicación de una de ellas para que soportara el desarrollo y diseño del videojuego.

Por lo anterior, se presenta un conjunto de metodologías de procesos de desarrollo de software que en la actualidad están siendo empleados por la comunidad de desarrolladores de videojuegos, entre las que se cuentan la del investigador Alvaro Galvis Panqueva, la de Pere Marqués, de Cascada para Videojuegos y las metodologías de Desarrollo Ágil para Videojuegos como Scrum, Game Unified Process y eXtreme Game Development (XGD).

II. Justificación

Actualmente las personas que habitan la ciudad de Cali adolecen de un reconocimiento de su le-

gado histórico, sitios importantes, héroes, heroínas, mitos, leyendas, y referentes de comportamientos ciudadanos, entre otros. Una de las causas que mejor explica este aspecto, se encuentra en el fenómeno migratorio.

Estadísticas sobre la composición de la ciudad, según su procedencia, fue objeto del estudio realizado por el Cidse-Univalle / Orstom en el año de 1996 que muestra los siguientes datos: Nativos 28.52%, Resto Valle 16.94%, Antioquia - Viejo Caldas 15.27%, Cauca-Nariño y Putumayo 12.70%, Tolima, Huila y Caquetá 6.35%, Costa Pacífica 4.91%, Valle Pacífico 4.23%, Valle Sur 4.20%, Centro-Oriente 2.37%, Bogotá 1.71%, Chocó 1.20%, Costa Atlántica 0.88, Extranjero 0.74%.

La falta de sentido de pertenencia por la ciudad, se puede entender a través de este fenómeno que durante años ha hecho que Cali haya tenido un crecimiento poblacional multicultural y diverso. Los acontecimientos sociales, económicos, demográficos y políticos a la que ha estado sometida Cali en los últimos años han fracturado la convivencia y deteriorado la cultura ciudadana, la identidad caleña se ha diluido reflejando un débil sentido de pertenencia.

Prende Amor x Cali, es un proyecto dirigido a niños y niñas entre los 7 y 9 años, que busca aportar una herramienta tecnológica que los acerque a su historia, especialmente, busca a través del uso de las TIC en el aula, el rescate de la tradición oral de la ciudad de Cali integrando de una manera didáctica y visual los mitos, leyendas y valores ciudadanos.

El presente artículo resume los resultados de una investigación, que propone, “hacer una reflexión sobre los nuevos problemas educativos generados por la omnipresencia de las tecnologías de la información y comunicación en la sociedad” (Area, 2001). Aborda, entre otras, cuestiones relativas a ¿qué formas debe adoptar la pedagogía ante las nuevas demandas y necesidades surgidas en la sociedad de la información? y ¿qué nuevas características deben adoptar los materiales didácticos en la era digital? (Area, 2001)

“Progresivamente, se han ido implantando en el ámbito educativo una serie de instrumentos y medios tecnológicos como son la televisión, el vídeo, el ordenador, la Internet o la videoconferencia entre otros, que, hasta hace poco tiempo, sólo pertenecían al uso y dominio de un grupo de expertos. Hoy en día, la utilización de estas herramientas tecnológicas como elementos de apoyo, adquiere gran importancia en el ámbito educativo, ya que favorece, mejora y

enriquece considerablemente el proceso de enseñanza-aprendizaje de los alumnos” (Silva, 2006). De ahí, que se haya planteado como eje problémico la pregunta de: ¿Qué procesos o técnicas de desarrollo de software para videojuegos se adaptan para el moldeamiento de un videojuego educativo que tiene como objetivo conocer los mitos y leyendas de la ciudad y promover el sentido de pertenencia por la ciudad de Cali en niños de 7 a 9 años?

III. Estado del Arte

“El ordenador es uno de los elementos de las nuevas tecnologías que presenta más funciones y posibilidades educativas. Las nuevas tecnologías están presentes en todos los ámbitos. Así como en el hogar se cuenta con el vídeo, la televisión, internet, etc., en el centro escolar se debe intentar que estas tecnologías estén presentes, pues serán una herramienta más, de gran utilidad en el proceso de enseñanza-aprendizaje y para la vida laboral en el futuro del alumno” (Silva, 2006).

El desarrollo y elaboración de software educativo es una de las actividades con mayor implementación en la actualidad ya que cumple un papel importante como medio de comunicación e información en la enseñanza y aprendizaje individual y grupal, al igual que permite cambiar el rol del docente al de un asesor, orientador y facilitador, e igualmente el rol del alumno se ve potenciado en aspectos como la autosuficiencia, responsabilidad, retroalimentación y el aprendizaje individual.

La informática educativa nace en los años 80, y desde 1990 aumentó su utilización gracias a diferentes acciones adelantadas por el gobierno nacional y el Servicio Nacional de Aprendizaje (SENA). Algunas universidades pioneras en la investigación y el desarrollo de la informática educativa han realizado algunas experiencias como son: Proyecto EIDOS (Educación informática e inteligencia artificial para el desarrollo de la inteligencia humana) por la Universidad Nacional, Proyecto Apolonio (Sistema tutorial inteligente para la solución de problemas en matemáticas) por la Universidad EAFIT. La Universidad Industrial de Santander (UIS) ha realizado proyectos en tres áreas básicas: anatomía, matemáticas básicas y avanzadas. El grupo de informática educativa Uniandes ha desarrollado algunos proyectos en informática educativa en las áreas de derecho, medicina y preescolar (Meléndez Acuña, 1995)

Es importante reconocer el trabajo que vienen adelantando diversas entidades que ofrecen al mercado productos y servicios de tecnología informática,

un gran ejemplo de esto es Parquesoft, entidad sin ánimo de lucro, de gran importancia y trayectoria en la industria del software, y que en sus inicios comenzó como una incubadora de proyectos, que ahora se han consolidado en empresas. Algunos de los proyectos realizados por Parquesoft y cuya finalidad es la intención educativa, son los siguientes:

- Blindux: Software desarrollado para personas con limitación visual.
- Burbuja: Software de evaluación e intervención pedagógica, para superar necesidades educativas especiales y potenciar capacidades de aprendizaje.
- Gérmenes y su familia: Material educativo desarrollado dentro de la línea de productos interactivos para público infantil, que combina la narrativa y la diversión para aprender y jugar.
- Guías didácticas Candy: Es una colección de multimedia, una por cada materia de cada uno de los cursos de primaria y bachillerato.
- Mateo y Juanita: La Odisea por Nuestra Historia V.1.0. Aventura interactiva que combina los dibujos animados, la información en texto y los juegos interactivos para acercar a los niños y niñas a la historia de Colombia de principios de siglo. En este volumen, Mateo y Juanita: La Odisea por nuestra historia, trata el período comprendido entre finales del siglo XIX hasta principios del XX (1870-1910); específicamente los siguientes temas: La Regeneración, La guerra de los Mil Días y La separación de Panamá.
- Submarino: Aventura pedagógica para niños entre 2 y 8 años que están en el proceso de adquisición del lenguaje oral y escrito para su aprendizaje.

La industria de desarrollo de software ha tenido un crecimiento del 48%, entre el año 2000 y 2004, según el estudio titulado El mercado del software en Colombia, realizado por la Oficina Económica y Comercial de la Embajada de España en Bogotá; aún no existen estudios recientes del crecimiento en los últimos años, pero se puede concluir que la producción de software está enfocada al desarrollo de soluciones comerciales, siendo muy poco el porcentaje de empresas que se dedican a la producción de software educativo.

IV. Ingeniería de Software para el Modelamiento de Videojuegos

La ingeniería de software es un área de las ciencias de la computación que ofrece los métodos y herramientas para el desarrollo y mantenimiento de un software de calidad. El término de ingeniería del software aparece utilizado de manera formal con oca-

sión de un congreso de la OTAN (Buxton, J. M.; Naur, P. & Randell, B. 1960), aunque existe evidencia de su uso informal en los años anteriores.

En esa ocasión se definió como “El empleo de técnicas y procedimientos típicos de la ingeniería en el desarrollo de productos de software”. Pero una de las primeras definiciones fue la propuesta por Fritz Bauer en la primera conferencia importante (Naur, P. & Randell, B., 1969) dedicada al tema: “El establecimiento y uso de principios de ingeniería robustos, orientados a obtener software económico que sea fiable y funcione de manera eficiente sobre máquinas reales”.

Galvis (2008), lo interpreta así: “Ingeniería de Software Educativo, es una referencia bastante completa y es una buena guía para el desarrollo del software. En esencia se conservan los grandes pasos o etapas de un proceso sistemático para desarrollo de materiales (análisis, diseño, desarrollo, prueba y ajuste, implementación). Sin embargo, en este caso se da particular énfasis a los siguientes aspectos: la solidez del análisis, como punto de partida; el dominio de teorías sustantivas sobre el aprendizaje y la comunicación humanas, como fundamento para el diseño de los ambientes educativos computarizados; la evaluación permanente y bajo criterios predefinidos, a lo largo de todas las etapas del proceso, como medio de perfeccionamiento continuo del material; la documentación adecuada y suficiente de lo que se realiza en cada etapa, como base para el mantenimiento que requerirá el material a lo largo de su vida útil”.

Desde el comienzo se han entendido a los videojuegos como “un juego”, pero un videojuego se entiende también como un medio de almacenamiento en el cual se graba, de manera digital y computarizada, un tipo de juego especial que se le ha bautizado con el mismo nombre que el medio en el que se guarda (Ruiz, Gómez, M. & Gómez, C. 2005)

Un videojuego educativo o software educativo es una experiencia de aprendizaje mediada por un software informático, que integra elementos de un juego (competencia, records, etc.), de interfaz de usuario (aspecto comunicacional), y elementos técnicos (aspecto computacional).

Los videojuegos nacen desde la misma concepción que un software tradicional, emplean la misma base teórica en su desarrollo, sin embargo algunas particularidades lo diferencian del resto de software existente, como se muestra en el Cuadro 1.

Video Juegos Educativos	Software Tradicional
<i>El enfoque es la diversión con aprendizaje.</i>	<i>En el enfoque es la funcionalidad.</i>
<i>Están diseñados para vivir una experiencia.</i>	<i>Están pensados para el desarrollo de procesos eficientes.</i>
<i>Facilidad en el uso y exigente en el nivel de estrategia, competencia y records.</i>	<i>Facilidad de uso.</i>
<i>Exigente en el diseño de interfaz.</i>	<i>Interfaz de usuario amigable.</i>

Cuadro 1. Comparación Videojuego Educativo y Software Tradicional. Fuente: Los Autores, 2010.

La Metodología de Alvaro Galvis Panqueva

La metodología de Ingeniería de Software Educativo propuesta por Galvis, publicada en 1991, ofrece mecanismos de análisis, diseño educativo y comunicacional, prueba piloto y de campo bastante sólidos, toda vez que se fundamentan en principios educativos, comunicacionales y de tecnología educativa de validez comprobada. Sin embargo, desde la perspectiva computacional no ha evolucionado, con lo que cabe enriquecerla tomando en cuenta los avances tecnológicos en el diseño y desarrollo computacional que se han logrado en los últimos años. Estos avances permiten incluir dentro de los productos de software nuevos recursos que enriquecen el potencial de acción de los mismos y que cabe usar desde el momento de formular su diseño (Gómez; Galvis & Mariño, 1991). La Figura 1 muestra de una manera esquemática la metodología propuesta por Alvaro Galvis.

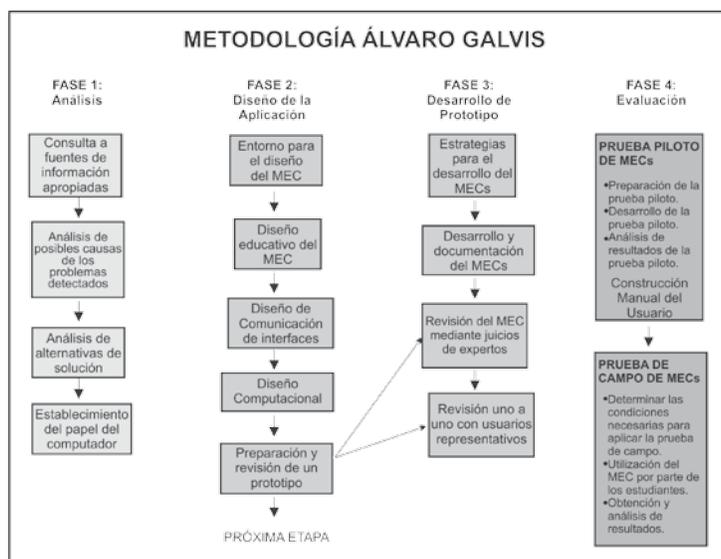


Figura 1. Modelo sistemático para selección o desarrollo de MECs propuesto por Alvaro Galvis. Fuente: Metodología Alvaro Galvis Panqueva, 1991.

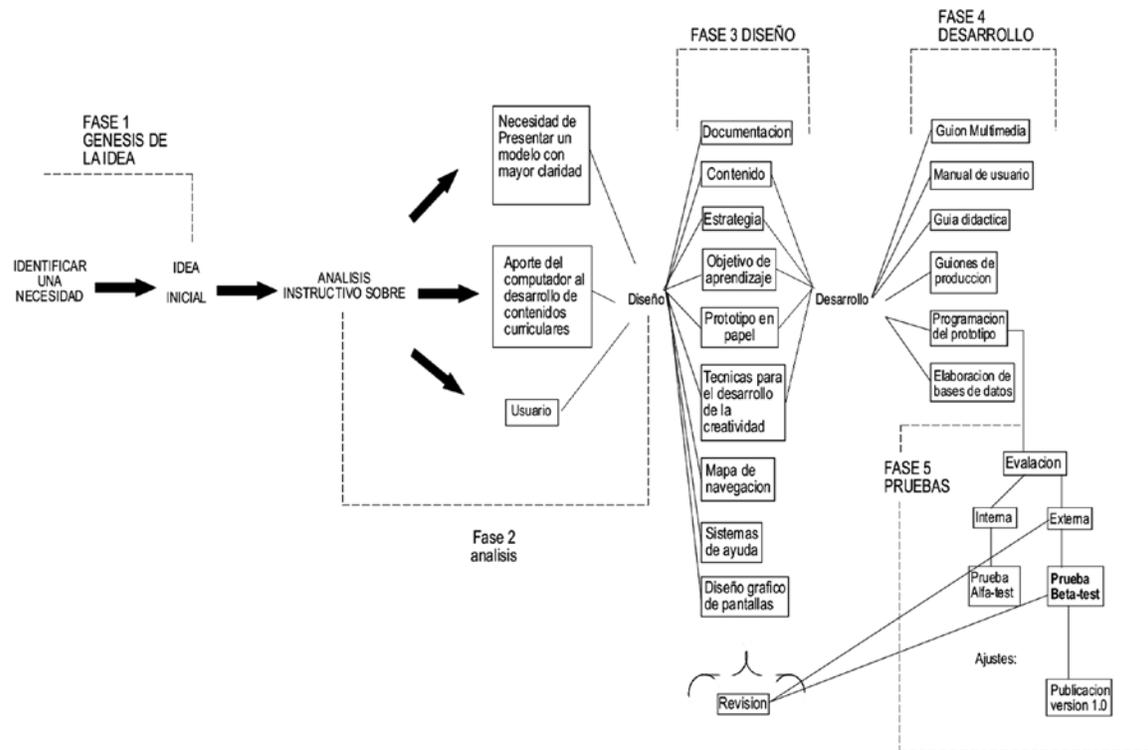


Figura 2. Metodología Pere Marquès. Fuente: Metodología Pere Marquès. 2010.

• La Metodología de Pere Marquès

Otros autores como Pere Marquès, lo definen así: "Software educativo se denomina a los programas para computadoras creados con la finalidad específica de ser utilizados como medio didáctico, es decir, para facilitar los procesos de enseñanza y de aprendizaje". La Figura 2 muestra la metodología definida por Pere Marquès.

Ambos autores coinciden en las siguientes etapas de Ingeniería de Software:

- Ofrecen pautas para elaboración de evaluaciones, pruebas piloto y pruebas de campo.
- Proponen la incorporación de grupos de trabajo multidisciplinario para el desarrollo del proyecto
- El análisis es profundo en cada uno de los aspectos del producto: curricular, instruccional, necesidades de usuarios e instituciones, entre otros.
- Incorporan evaluaciones tempranas al proyecto favoreciendo la conveniencia y factibilidad del producto.
- Emplean desde el inicio el desarrollo de prototipos sometiéndolos a un mejoramiento continuo.

• Metodología de Cascada para los Videojuegos

La metodología más común en el desarrollo de videojuegos es la de cascada, que consiste en una adaptación del método de proceso de desarrollo de software de gestión empresarial llamado

también Cascada o el Ciclo de Vida Clásico o Tradicional de un software (Jiménez, 2006). En éste, cada una de las fases de desarrollo del software se suceden una tras otra así: Planeación, Análisis, Diseño, Programación, Pruebas, Producción y Mantenimiento.

Las características de cada una de las fases del proceso de desarrollo de software de cascada se relacionan a continuación:

Definición del Proyecto. Durante esta fase se resuelven las siguientes inquietudes: ¿Por qué se necesita un nuevo proyecto?, ¿Qué es lo que se desea alcanzar?, ¿La institución tiene un problema y puede ser resuelto mediante el desarrollo de un nuevo sistema de información? Igualmente, se identifican los objetivos generales, se especifica el alcance del proyecto y se define un plan del proyecto.

Análisis de Sistemas. Durante esta fase se da solución a los interrogantes de: ¿Qué hacen los sistemas ya existentes?, ¿Cuáles son sus fuerzas, debilidades, etc.?, ¿Qué debe hacer un nuevo sistema para resolver estos problemas?, ¿Qué requerimientos de información deben ser satisfechos por la solución?, ¿Cuáles son sus beneficios y sus costos? Es una fase de extensa recopilación e investigación. Durante esta fase se encuentran los requerimientos detallados del producto a desarrollar como son las entradas, sali-

das, funciones a realizar, la información a manejar, el rendimiento y las interfaces requeridas. Con esta información se construyen los modelos conceptuales de especificación del producto.

Diseño de Sistemas. Se definen las especificaciones del diseño lógico y físico de la solución de sistemas. Se definen los Diagrama de Casos de Uso, Diagrama del Modelo Físico de Datos, Diagrama de Clases, Diagrama de Secuencias.

Construcción. Con base en el diseño se construye el producto final. Este modelo no facilita la creación de un modelo del producto a construir y que sirva para evaluar y refinar los requisitos. No se produce un proceso interactivo en el que se afine un prototipo para que satisfaga las necesidades del usuario.

Pruebas. Una vez construido el producto se hacen las pruebas a cada uno de los módulos individuales, luego se prueba la interacción entre los módulos y por último se prueba el producto como un todo. Consiste en el uso y evaluación del sistema antes de que sea instalado y se encuentra en producción. Se prueba el software para estar seguros que opera adecuadamente desde el punto de vista técnico y del funcional de los negocios.

Producción. Se implementa finalmente el producto y éste puede ser tangible en el caso del hardware o intangible en el caso del software. Consiste en los pasos finales para poner al sistema nuevo o modificado en operación. Durante esta fase se adelanta la capacitación a los usuarios.

Mantenimiento. Los productos, indudablemente, sufrirán cambios después de que se entregan al usuario, debido a que se encuentran nuevos errores, o a que el producto debe adaptarse a cambios del entorno externo, o debido a que el usuario requiere una ampliación funcional o una mejora en el rendimiento del producto.

Para el desarrollo de un videojuego las fases adaptables del método de cascada son:

Especificación del juego: Se realiza un documento que especifica el juego desde la perspectiva del usuario (jugador). Es el equivalente a la fase de Planeación del ciclo de vida de un software de gestión empresarial.

La Biblia del arte y la historia: Realizado por los productores y los directores artísticos donde especifican las herramientas y los conceptos a usar desde la parte artística. Se define también la historia, el guión y

el diseño del juego desde distintos escenarios. Es el equivalente a la fase de Análisis del ciclo de vida de un software de gestión empresarial.

Especificaciones técnicas: Describe las herramientas ingenieriles como UML o diagramas de sistema, e interacción entre tareas y el código (esta es la única fase estrictamente ingenieril). Es el equivalente a la fase de Diseño del ciclo de vida de un software de gestión empresarial.

Construcción: Se inicia la programación del videojuego, la creación de modelos, texturas, niveles, arte conceptual y toda aquella tarea que dependa del diseño. Es el equivalente a la fase de Programación del ciclo de vida de un software de gestión empresarial.

Aseguramiento de calidad: El equipo de QA debe verificar que todo se contraste con lo planteado en los diseños originales de los documentos adecuadamente. Es el equivalente a la fase de Pruebas del ciclo de vida de un software de gestión empresarial.

Pruebas de juego: Sesiones con los directivos y productores donde se les muestran las características del juego y se hace realimentación que debería modificar el proceso para obtener un producto a total satisfacción.

Pruebas alfa: Una vez se determina que el juego está en una fase estable, se libera a un selecto grupo de evaluadores que determinan la calidad del juego, y plantean cambios para mejorar la experiencia del juego en general (se supone que no deben surgir cambios radicales). Pruebas beta: Se libera el juego a una gran audiencia y se determinan aspectos que gustan o disgustan del juego para el mercado en general.

Golden master: Momento en el que se libera el juego para el consumo masivo. Es el equivalente a la fase de Mantenimiento del ciclo de vida de un software de gestión empresarial.

Esta metodología hace una partición del proceso de desarrollo del software en distintas fases como se muestra en la Figura 3.

· **Desarrollo Ágil de Videojuegos**

Varias metodologías ágiles se han propuesto desde diversos sectores de la industria, a continuación se relacionan algunas de ellas:

· **a. Scrum**

Es una metodología ágil iterativa orientada a la comunicación y a los resultados, disciplina y una rápida producción de versiones visibles del juego, de mane-

ra que se puedan hacer las evaluaciones cualitativas que exigen una visión e interacción del producto. Para realizar un seguimiento adecuado al producto se plantean algunas pautas: verificación constante del producto, iteraciones de hasta 30 días, estimación de tiempo de ejecución de tareas y equipos capaces de auto-organización.

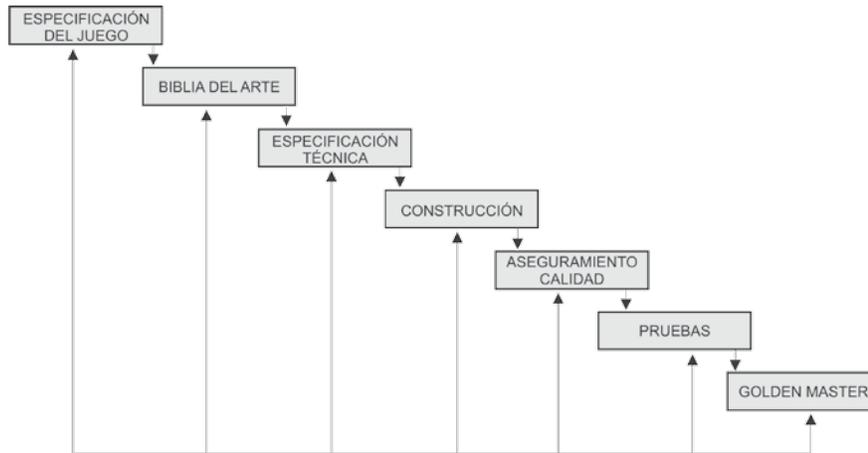


Figura 3. Metodología de Cascada para Videojuegos. Fuente: Los Autores. 2010.

Scrum plantea que se debe desarrollar entre pocas personas, un equipo de desarrollo de 5 a 8 personas es ideal, y pueden inclusive subdividirse en la medida en que lo vean pertinente, manteniendo un "scrum de scrums". Los equipos pueden dividirse de acuerdo a los enfoques que cada equipo tenga en el videojuego. Para este enfoque, el cliente es el director creativo y diseñador líder del proyecto, puesto que son los encargados de conceptualizar lo que debe ser el videojuego. Además, la realimentación se debe realizar a medida que se vaya desarrollando el videojuego y no a posteriori que pueda no serle de utilidad al estudio. Deben utilizarse unidades de prueba para las distintas funcionalidades del videojuego.

· b. Game Unified Process

Esta metodología de desarrollo fue el producto de un proyecto de realización de un videojuego tipo casino online. Durante este proceso, el patrocinante del proyecto decidió que deberían hacerse cambios a la forma de desarrollo de manera que pudieran cumplirse con los plazos, y de esta manera se decidió utilizar una combinación de dos metodologías de desarrollo utilizadas comúnmente en el software tradicional.

La primera es la utilización del Rational Unified Process el cual plantea un diseño estricto y una documentación rigurosa de cada paso y funcionalidad a implementar. Luego de dividir el equipo se decidió que

aquellos familiarizados con el proceso (ingenieros en su mayoría) utilizaran este enfoque, dándoles una organización mayor durante el proceso y el trabajo en equipo. El otro equipo adoptó la metodología extreme Programming con algunas variaciones para que personas de otras disciplinas lo pudieran aplicar, lo cual se logró en poco tiempo. En este proceso los ciclos cortos ayudaron a mantener la comunicación fluida entre equipos y el componente artístico se sentía menos restringido que con el RUP y con mayor capacidad comunicativa.

· c. Extreme Game Development (XGD)

XGD es una adaptación de eXtreme Programming (XP), una práctica muy común en el desarrollo de software ágil. Esta práctica sin embargo, se limita a su utilización por parte de ingenieros, puesto que XP está enfocado a programadores, por lo que se debió adaptar para artistas, gerentes, productores, etc. Su uso se fundamenta en cuatro principios fundamentales:

Simplicidad: lo más importante es la simplicidad, no siempre se necesita que el juego sea innovador en todos los géneros, debe tener lo necesario.

Comunicación: se debe priorizar la comunicación de forma oral, de manera que las personas se sientan más cómodas para expresarse, no necesariamente con una gran cantidad de documentos cuya utilidad puede verse reducida (en número de personas que la utilicen), sin olvidar que existen documentos indispensables tales como: las propuestas del juego, los conceptos del juego, el documento del diseño del juego, entre otras.

Realimentación: la realimentación se conoce normalmente como la experiencia que trae haber trabajado en proyectos anteriores.

Coraje: se necesita coraje para que se empiece a crear el proyecto, en ocasiones, sin que todo esté determinado, manteniendo incertidumbre por algún tiempo.

En esta metodología se utiliza un elemento análogo a los casos de uso, llamado historias de usuario, las cuales son pequeñas sentencias que determinan cosas particulares que debe tener el proyecto, y son realizadas por los distintos equipos ("debe hacerse un demo jugable", "debe realizarse un video de 30 segundos que no ocupe un espacio mayor

• Definición del Diagrama del Modelo Físico de Datos

El diagrama del modelo físico de datos permite identificar, describir y relacionar cada uno de los datos que permitirán la ejecución, almacenamiento y la distribución de la información necesaria para operar y mantener la información relacionada con el videojuego.

El diagrama del modelo físico de datos durante la Fase de Construcción del videojuego en la Metodo-

logía de Cascada para los Videojuegos, se implementará como la Base de Datos que almacenará toda la información ejecutada y generada durante la ejecución del videojuego.

El diagrama del modelo físico de datos para el videojuego "Prende Amor X Cali" se realizó empleando la herramienta de ingeniería de software asistida por computador de SYBASE o Power Designer.

La Figura 5 muestra el Diagrama del Modelo Físico de Datos para el videojuego "Prende Amor X Cali".

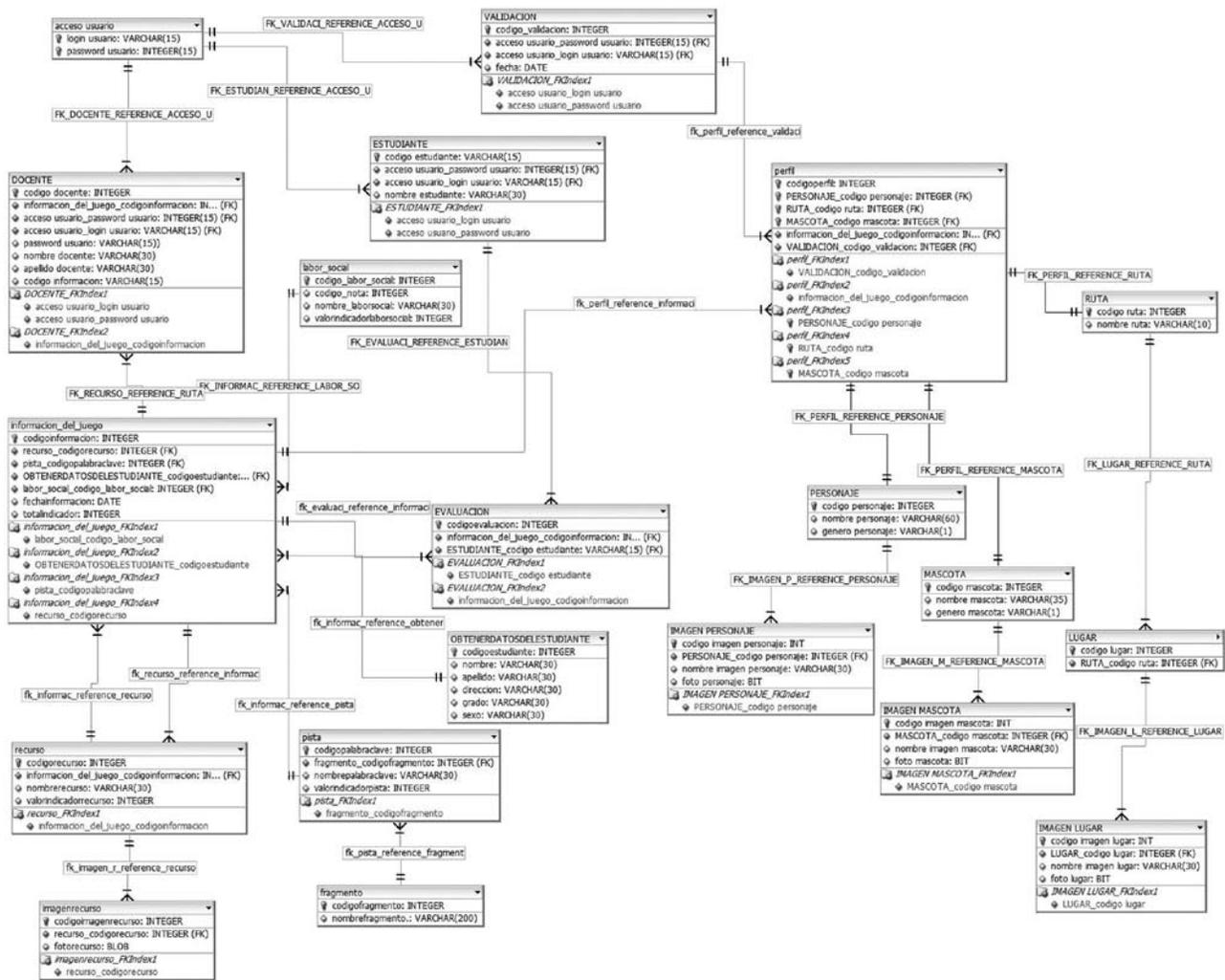


Figura 5. Diagrama Modelo Físico de Datos del Videojuego "Prende Amor X Cali". Fuente: Los Autores, 2010.

• Definición del Diagrama de Clases

El diagrama de clases es un esquema estático que representa subconjuntos del modelo físico de la base de datos que soportan la ejecución de las funcionalidades del software. Este diagrama permite identificar y describir las características y métodos u operacio-

nes de las clases para su manipulación, como las diferentes relaciones entre éstas.

Los diagramas de clases para el videojuego "Prende Amor X Cali" se realizaron empleando la herramienta de ingeniería de software asistida por computador de Sybase o Power Designer.

La Figura 6, muestra el Diagrama de Clases, a manera de ejemplo, de la funcionalidad o subconjunto de las funciones del videojuego "Prende Amor X Cali" de Obtener Datos del Juego de todos los componentes del juego involucrados durante el desarrollo del juego para una ruta específica, respectivamente.

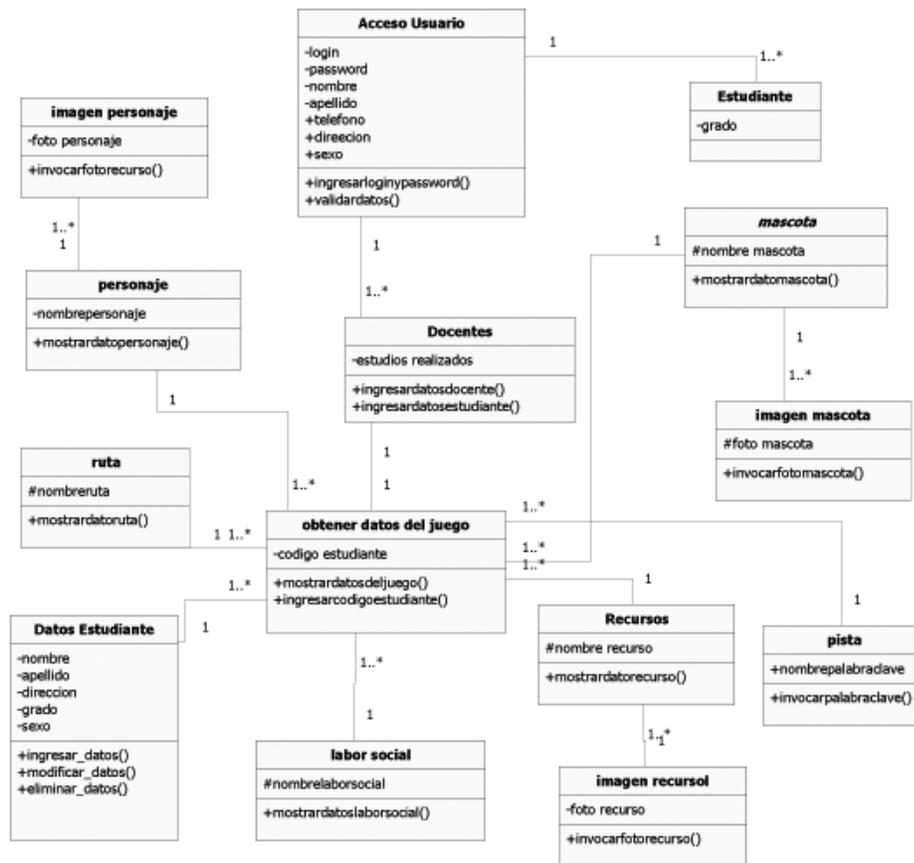


Figura 6. Diagrama de Clases de la Funcionalidad Obtener Datos del Juego "Prende Amor X Cali". Fuente: Los Autores, 2010.

Definición del Diagrama de Secuencias

El diagrama de secuencias hace parte de los esquemas de comportamiento que representan la interacción en el tiempo de los descriptores de clases e interfaces que a su vez soportan las funcionalidades del software. La interacción entre los descriptores se hace a partir de pasos de mensaje de solicitudes o ejecuciones de los métodos u operaciones que permiten la manipulación de los descriptores. Se empleó igualmente Sybase o Power Designer.

La Figura 7 muestra, a manera de ejemplo, el diagrama de secuencias respectivo para la Ruta 1 del juego, que debe seleccionar el jugador del videojuego educativo, a fin de dar inicio al mismo. Este diagrama representa la funcionalidad de Gestionar el Despla-

zamiento del avatar del jugador dentro del mismo. En esta la Ruta 1, que fue la ruta que se escogió para desarrollar el Prototipo del videojuego, el jugador deberá realizar pruebas en los lugares de: Unicentro, Plaza de Toros, Hospital Universitario del Valle – H.U.V., Parque de las Banderas, Estadio Pascual Guerrero y la Biblioteca Departamental.

La Ruta 1 inicia en Pance, y entre el inicio y Unicentro, el jugador reconocerá a través de fotos las diferentes Universidades que se encuentran en el sector. Una vez en los lugares del juego, el jugador deberá cumplir un objetivo, un reto y tendrá una recompensa traducible en puntaje, por lo que para cada uno de los lugares referidos se cumplirá con lo siguiente:

- Lugar Unicentro: El objetivo es encontrar un policía, el reto es pasar un laberinto y, la recompensa es obtener una tarjeta del MIO.
- Lugar Plaza de Toros: El objetivo es encontrar un personaje del Festival de Petronio, el reto es interpretar un ritmo musical y, la recompensa es obtener un fragmento de la historia de Buziraco.
- Lugar Hospital H.U.V.: El objetivo es encontrar un

médico, el reto es vacunarse contra una posible enfermedad y, la recompensa es obtener la ayuda de un médico.

- Lugar Parque de las Banderas: El objetivo es enumerar el número de banderas, el reto es identificar unas banderas en específico y, la recompensa es obtener otro fragmento de la historia del Buziraco.
- Lugar Estadio Pascual Guerrero: El objetivo es conocer los dos equipos de fútbol de la ciudad, el reto es jugar a penales y, la recompensa es obtener una recomendación de guardar comportamiento frente a las acciones de las llamadas Barra Bravas.
- Lugar Biblioteca Departamental: El objetivo es encontrar un libro en específico, el reto es mejorar la ortografía de un texto y, la recompensa es obtener una pieza de la Cultura Calima. (ver figura 7 en la siguiente página)

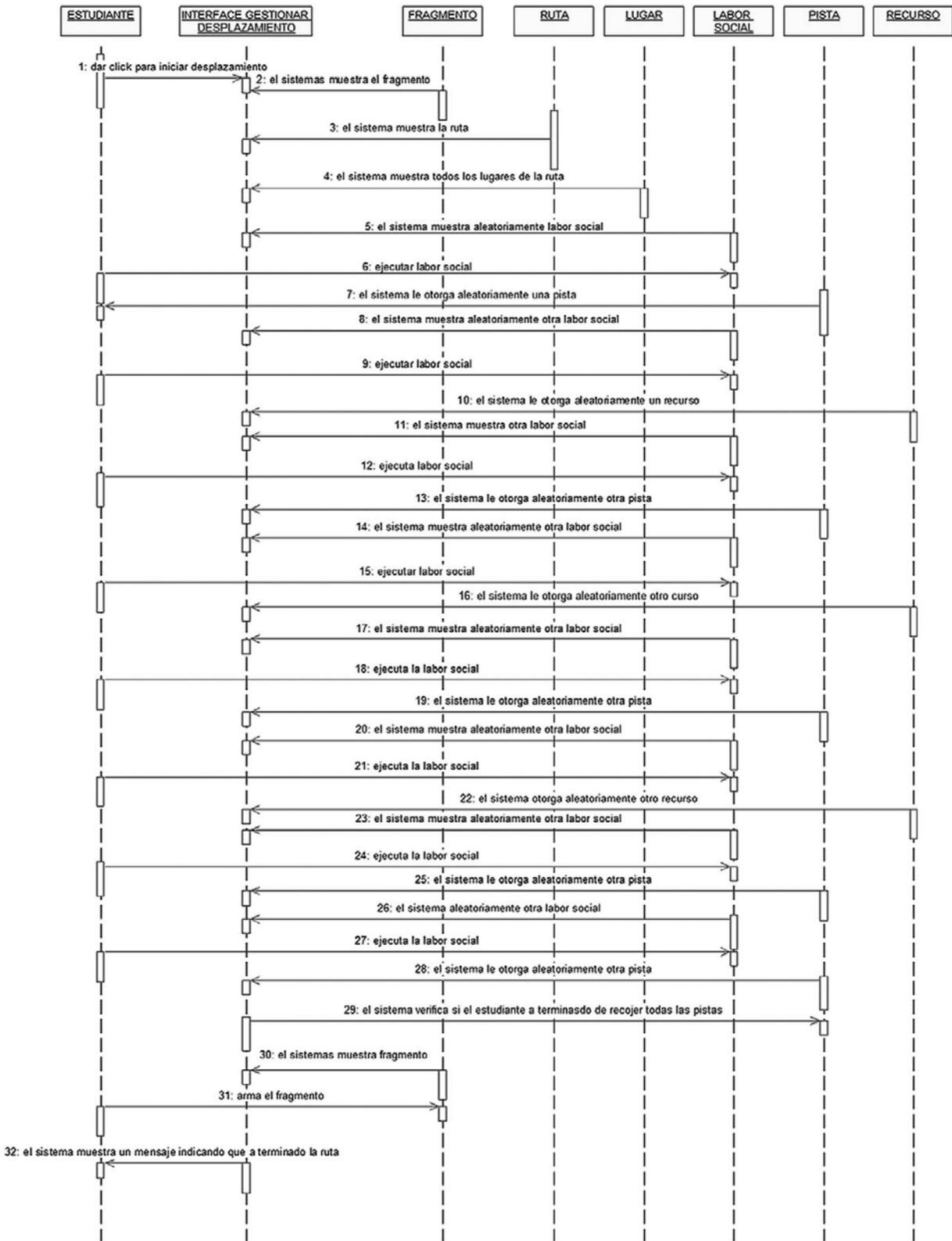


Figura 7. Diagrama de Secuencias sobre la Ruta 1. Fuente: Los Autores, 2010.

VI. Conclusiones

- La metodología más adecuada que permitió desarrollar el prototipo del proyecto "Prende Amor X Cali" fue la de Cascada para Videojuegos, ya que facilitó integrar el modelo educativo, el modelo comunicacional y el diseño computacional.
- Existe escasa información sobre la ingeniería del software aplicada al modelado y diseño de software educativo, debido al poco desarrollo de investigación en este campo.
- Se pudo constatar que un videojuego es una herramienta óptima que integra las tecnologías de información y comunicación con los procesos pedagógicos, para soportar los procesos de enseñanza – aprendizaje en el aula.
- Durante el desarrollo del videojuego se pudo evidenciar que sólo a través de un trabajo en equipo e interdisciplinario es posible la construcción del prototipo, porque es necesario gestionar el proyecto a partir del trabajo de los equipos de: Guionización, Dibujo, Ingeniería de Software, Programación y Bases de Datos.
- La construcción del prototipo de la Ruta 1 y su validación y verificación frente a las funcionalidades generales del videojuego y frente a las especificaciones técnicas de diseño, respectivamente, garantizaron el logro de la calidad del mismo y, por ende la continuidad en el desarrollo del videojuego.
- El empleo de herramientas de ingeniería de software o llamadas herramientas CASE – Computer Aided Software Engineering, como Sybase – Power Designer, facilitó el diseño del prototipo del videojuego de una manera sencilla como el traducir en modelos gráficos los requerimientos educativos en el aula y los requerimientos funcionales del software educativo.
- El videojuego "Prende Amor X Cali" demostró que es posible la intervención social en comunidades de escasos recursos, y por lo tanto, facilitar el acceso a recursos tecnológicos de informática y de comunicación de una manera lúdica.
- Las pruebas de validación de las funcionalidades del software, se han aplicado a través de las pruebas de caja blanca a las funcionalidades soportadas por el prototipo del videojuego.

- Dado que uno de los aspectos de la calidad del software consiste en realizar pruebas de verificación con respecto al diseño y construcción, se adelantaron las respectivas pruebas de caja negra al código del prototipo.

VII. Referencias

- [1] Area, M. (2001), *Educación en la Sociedad de la Información*, Editorial Descleé de Brouwer, España.
- [2] Berrocal, V. (1999), *Micromundos*, San José, Editorial Universidad Estatal a Distancia, Costa Rica
- [3] Buratto, C. et al. (2002), "La Informática como Recurso Pedagógico – Didáctico en la Educación", disponible en: <http://www.monografias.com/trabajos10/recped/recped.shtm>, recuperado: 16 de septiembre de 2010
- [4] Buxton J. M.; Naur P. & Randell B. (1969), *Software Engineering Concepts and Techniques*. Nato Conference on Software Engineering
- [5] Calderón, J.; Díaz, W.; Angulo, Z. & Márquez, N. (2008), "Metodologías para el Desarrollo de Software Educativo", disponible en: <http://softeduc2008.files.wordpress.com/2008/11/metodologias.pdf>, recuperado: 08 de octubre de 2010.
- [6] Gómez, C.; Galvis, P. & Mariño, D. (1998), *Ingeniería de Software Educativo con modelaje orientado por objetos: un medio para desarrollar micromundos interactivos*. Informática Educativa, UNIANDES – LIDIE. Vol 11, No, 1.
- [7] Jiménez, A. (2009), *Modelación de Sistemas de Información un Enfoque Metodológico y Práctico*, Segunda edición, Cali, Editorial Universidad Santiago de Cali.
- [8] Lacueva, A. (2000), *Ciencia y Tecnología en la Escuela*, Editorial Popular, España.
- [9] Marqués, P. (2010), *Técnicas Didácticas con TIC. Libros Digitales*, Colección EDUCACIÓN DIM No. 1, Barcelona, Universidad Autónoma de Barcelona.
- [10] Meléndez Acuña, A. (1995), *Informática y Software Educativo*. Santa Fe de Bogotá, ICFES y Universidad Javeriana, pp. 18-25.
- [11] Morales Escobar, G., (1999), *La Población en Santiago de Cali*, Departamento Administrativo de Planeación – DAP, Municipio Santiago de Cali.
- [12] Naur, P. & Randell, B. (1969), *Software Engineering: a report on a conference sponsored by the nato science committee*, Nato.
- [13] Pérez, V. et al. (s. f.), "Folleto del Curso Informática Educativa", La Habana, Cuba, disponible en: <http://www.monografias.com/trabajos31/software-educativo-cuba/software-educativo-cuba.shtml>, recuperado: 18 de noviembre de 2010.
- [14] Ramos, Mirtha (s.f.), "Software Educativo. Metodología y Criterio para su Elaboración y Evaluación", Universidad Estatal a Distancia – UNED, San José, Costa

Rica, disponible en: <http://www.monografias.com/trabajos31/software-educativo-cuba/software-educativo-cuba.shtml>, recuperado: 18 noviembre 2010.

[15] Ruiz, J. *et al.* (2005), Videojuego rpg para PC "mitología colombiana", Bogotá, Universidad Católica de Colombia.

[16] Salas, M. (1999), *Una Propuesta Didáctica para la Programación con Micromundos*, San José, Editorial Universidad Estatal a Distancia, Costa Rica.

[17] Silva, S. (2006), *Informática Educativa*, Editorial Ideas Propias, España.

[18] _____ (2006), *Medios Didácticos Multimedia para el Aula*, Editorial Ideas Propias, España.

Álvaro Iván Jiménez Alzate. Ingeniero Mecánico de la Universidad Autónoma de Occidente - UAO, CAP Instructor de Empresas del SENA, Magíster en Ciencias Computacionales del ITESM de México, Autor de los libros *Modelamiento de Negocios y Nuevas Prácticas Informáticas*, *Modelación de Sistemas de Información un Enfoque Metodológico y Práctico* y *Una Visión Sistémica de la Auditoría Informática*, Docente del área de Ingeniería de Software, Director del Grupo de Investigación KIMSA reconocido por Colciencias en la categoría D, actualmente es el Decano de la Facultad de Ingeniería de UNICATOLICA, decaingenieria@unicatolica.edu.co;

Diana Patricia Rico Largo. Ingeniera Informática de UNICIENCIA, Especialista en Diseño de Ambientes de Aprendizaje de UNIMINUTO, adelanta estudios de Maestría en Gestión de Informática y Telecomunicaciones en el ICESI, Docente del área de Informática, Coordinadora del equipo de Ingeniería de Software del Proyecto "Prende Amor X Cali", diana25.r@hotmail.com

Carlos Mauricio Méndez Solarte. Estudiante perteneciente al Grupo de Investigación KIMSA e integrante del Semillero de Investigación AYNI y del equipo de Ingeniería de Software del Proyecto "Prende Amor X Cali", Egresado del Programa de Tecnología en Sistemas de la Fundación Universitaria Católica Lumen Gentium - UNICATOLICA, en el mes de agosto de 2010.

Stephany Cerón Salas. Estudiante perteneciente al Grupo de Investigación KIMSA e integrante del Semillero de Investigación AYNI y del equipo de Ingeniería de Software del Proyecto "Prende Amor X Cali", Egresado del Programa de Tecnología en Sistemas de la Fundación Universitaria Católica Lumen Gentium - UNICATOLICA, en el mes de agosto de 2010.

Steven Palechor Betancourt. Estudiante perteneciente al Grupo de Investigación KIMSA e integrante del Semillero de Investigación AYNI y del equipo de Ingeniería de Software del Proyecto "Prende Amor X Cali", Egresado del Programa de Tecnología en Sistemas de la Fundación Universitaria Católica Lumen Gentium - UNICATOLICA, en el mes de agosto de 2010.