



Universidad de las Ciencias Informáticas

Facultad 4

***Multimedia para el apoyo a la enseñanza de la
Programación Lineal en la asignatura
Investigación de Operaciones en la Universidad
de las Ciencias Informáticas***

**Trabajo de Diploma para optar por el Título de Ingeniero en Ciencias
Informáticas**

Autor:

Omar Darien Castillo González

Tutores:

Ing. Yisel Barrabía Legrá

Dr. C. Efrén Vázquez Silva

La Habana, Junio 2014

“Año 56 de la Revolución”

Declaración de autoría

Declaro ser el único autor de este trabajo y autorizo a la Universidad de las Ciencias Informáticas (UCI) a que haga el uso que estimen pertinente con el mismo.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Autor:

Omar Darien Castillo González

Tutores:

Ing. Yisel Barrabia Legrá

Dr. C. Efrén Vázquez Silva

A black and white close-up portrait of Steve Jobs. He is wearing his signature round glasses and has a slight smile. His hand is resting against his face, with his fingers near his eye. The background is dark, making his face the central focus.

**"La innovación
es lo que
distingue
a un
líder de
los demás"**

Steve Jobs

Dedicatoria

Dedico este trabajo a las personas que más quiero en este mundo, a los que se sacrificaron siempre por mí, a ustedes por ser los mejores padres y amigos les entrego todo mi amor y cariño.

A mis padres.

Agradecimientos

A mi Papá por ser un padre ejemplar, por estar ahí siempre que lo he necesitado, por apoyarme en todas mis decisiones y ser el hombre que algún día quisiera poder ser.

A mi Mamá por quererme tanto y soportarme tantas travesuras, por apoyarme aunque no estuviese de acuerdo, por ser la mejor madre del mundo y la persona más especial de mi vida.

A mi compañero de mesa Roanny por ser más que un amigo, por ayudarme en todo momento, a ti mi hermano no te debo un año, te debo la carrera.

A mis hermanos Andy y Álvaro por ser su ejemplo a seguir y su ídolo.

A Pedro, Darlenis, Edelso, Yoandris, Marcos, Alex y Fidel por ser parte de la Realeza, por pasar juntos los mejores momentos de la universidad.

A mi otro hermano Leandro por apoyarme en todo momento, por los buenos ratos que hemos pasado, por ser fanático a mí.

A Yely por ayudarme en los momentos que lo he necesitado, por ser mi hermana preferida.

A mi amiga del alma Lisbeth, por el tiempo que hemos compartido, que aunque parezca poco es más que suficiente para saber que siempre vas a ser mi hermanita linda, te quiero.

A mis tutores por apoyarme en todo lo que estuvo a su alcance, por siempre confiar en mí.

En fin a todos los que de una forma u otra contribuyeron a que hoy sea la persona que soy. Muchas gracias a todos.

Resumen

En la Universidad de las Ciencias Informáticas un tema de gran importancia dentro de la asignatura Investigación de Operaciones, es precisamente el que está relacionado con los problemas de Programación Lineal. Aunque existe una gran cantidad de información respecto a este, son insuficientes los recursos didácticos que motiven a los estudiantes, les orienten cómo abarcar todo lo que necesitan saber acerca del tema y cómo autoevaluarse, además de servirle de guía para el estudio de manera independiente. Por tales razones el presente trabajo de diploma consiste en desarrollar un producto multimedia, como medio didáctico, que sirva de apoyo al proceso de enseñanza-aprendizaje en la asignatura Investigación de Operaciones. El desarrollo del sistema fue guiado por la metodología de desarrollo Programación Extrema. Para la implementación de la aplicación se utilizó el Lenguaje de Etiquetado de Hipertexto HTML5, CCS3 apoyado en el *framework* de diseño Twitter-Bootstrap y JavaScript a través de la biblioteca JQuery. El Entorno de Desarrollo Integrado seleccionado fue NetBeans IDE y Visual Paradigm for UML fue la herramienta de modelado de *software* seleccionada. Además, se seleccionó el lenguaje de modelado OMMMA-L. Fueron realizadas las pruebas funcionales a la solución propuesta, las cuales permitieron asegurar el correcto funcionamiento de la multimedia. Se obtuvo como resultado final una multimedia que cumple con el objetivo general de la investigación.

Índice

Introducción	2
Capítulo 1 Fundamentación Teórica	7
Introducción	7
1.1 Multimedia	7
1.1.1 Hipertexto e Hipermedia	8
1.2 Multimedia Educativa	11
1.3 Investigación de Operaciones (IO)	13
1.4 Programación Lineal en la IO	14
1.5. Análisis de soluciones similares	15
1.6. Estudio de las metodologías	16
1.6.1 Proceso Unificado Racional (RUP)	16
1.6.2 Microsoft Solution Framework (MSF)	17
1.6.3 Programación Extrema (XP)	18
1.6.4 SCRUM	20
1.7 Selección de la metodología para dar solución al problema	21
1.8 Lenguajes de Modelado	22
1.8.1 Lenguaje Unificado de Modelado (UML)	22
1.8.2 Lenguaje de Modelado Orientado a Objetos de Aplicaciones Multimedia (OMMMA-L)	
.....	23
1.9 Tendencias y tecnologías actuales. Selección de las herramientas y lenguajes de desarrollo	24
1.9.1 ¿HTML5 o Flash?	24
1.9.2 Lenguajes de programación	25
1.9.3 Framework de desarrollo	28
1.9.4 Entorno de desarrollo integrado	29
1.9.5 Herramienta CASE	31
Conclusiones del capítulo	33
Capítulo 2 Descripción de la solución propuesta	35
Introducción	35
2.1 Descripción de la propuesta de solución	35
2.2 Usuarios del sistema	35
2.3 Diagrama conceptual del negocio	36
2.4 Exploración	37
2.4.1 Historia de usuario	37
2.4.2 Plantillas de historia de usuario	37
2.5 Planificación	40
2.5.1 Estimación de esfuerzo por historias de usuario	40
2.5.2 Iteraciones	41
2.5.3 Plan de duración de iteraciones	42
2.5.4 Plan de entregas	43
Conclusiones del capítulo	43
Capítulo 3 Construcción de la solución propuesta	46
Introducción	46

3.1 Diagramas de Presentación	46
3.2 Tarjetas Clase-Responsabilidades-Colaboradores (CRC)	47
3.3 Tareas de ingeniería	49
3.4 Estándares de codificación	54
3.5 Pruebas	55
3.5.1 Desarrollo dirigido por pruebas (TDD)	56
3.5.2 Pruebas unitarias	56
3.5.3 Pruebas de aceptación	57
3.6 Análisis de los resultados de las pruebas	60
Conclusiones del capítulo	61
Conclusiones generales	63
Recomendaciones	64
Referencias bibliográficas	65
Anexos de la investigación	71

Introducción

Introducción

El uso de las Tecnologías de la Información y las Comunicaciones (TIC) en apoyo al proceso de enseñanza-aprendizaje ha proporcionado nuevos canales de comunicación que se deben aprovechar. Estas fomentan la colaboración entre los alumnos, ayudan a centrarse en los aprendizajes, aumentan la motivación y el interés, favorecen la búsqueda, estimulan el desarrollo del razonamiento, la resolución de problemas, la creatividad y la capacidad de aprender a aprender. Además, han abierto posibilidades superiores en el área del aprendizaje, la investigación y la producción.

El desarrollo de *software* es una de las industrias con mayor crecimiento dentro de las TIC, el cual en sus inicios pasó por momentos críticos. Su evolución ha permitido que en la actualidad se puedan elegir diversas formas de desarrollar, distribuir y mantener *software* con cierto éxito. En Cuba se ha impulsado el desarrollo de la informática, y en particular el desarrollo de la industria del *software*, razón por la cual fue creada la Universidad de las Ciencias Informáticas (UCI) como un proyecto social de la Revolución.

La UCI desde sus inicios se ha encargado de formar profesionales calificados para producir *software* y servicios informáticos. Dentro de esta, el centro FORTES tiene entre sus objetivos principales la producción de *software* educativo, ejemplo de ello se tiene la colección Multisaber y El Navegante. Estos *software*, denominados hiperentornos de aprendizaje, definidos por el MS. C César Labañino, como sistemas informáticos basados en tecnología hipermedia que contienen una mezcla o elementos representativos de diversas tipologías de *software* educativo (Rodríguez, 2011), surgen como apoyo al proceso de enseñanza-aprendizaje en la educación primaria y secundaria.

Un *software* educativo es todo aquel programa para computadora que se desarrolla con la finalidad específica de ser utilizado como recurso didáctico en el proceso de enseñanza-aprendizaje. Este, dentro de las TIC contribuye a un proceso de enseñanza más rápido y actualmente desempeña un papel muy importante en la manera de transmitir los contenidos, así como un medio para alcanzar el conocimiento.

En la UCI se imparte la asignatura Investigación de Operaciones a los estudiantes de tercer año, en el segundo semestre; la cual está dividida en cuatro temas. Uno de ellos es Programación Lineal, de especial importancia para el apoyo a la toma de decisiones y, aunque se cuenta con el sitio de la asignatura sobre la plataforma Moodle, en cada facultad; no existe un material de consulta que facilite de manera rápida, dinámica e interactiva acceder a la

información elemental sobre esta temática en específico. Además, de manera general en los resultados tradicionales de las pruebas parciales, a los que se tuvo acceso en los informes semestrales del Departamento Metodológico Central de Ciencias Básicas, facilitados por uno de los tutores; se evidencia que los estudiantes no logran relacionar el conocimiento teórico adquirido en clases con la experiencia práctica. Por otra parte, son insuficientes los recursos didácticos utilizados para orientar a los estudiantes cómo abarcar todo lo que necesitan conocer acerca de este importante tema en la asignatura, estudiar de manera independiente y auto evaluarse; además de que los existentes ofrecen poca motivación.

Tomando en referencia la situación actual surge el **problema de investigación**: ¿Cómo apoyar el proceso de enseñanza-aprendizaje del tema Programación Lineal, en la asignatura Investigación de Operaciones? De lo planteado anteriormente se deriva como **objeto de estudio** el proceso de desarrollo de *software* con tecnología multimedia. Determinando para la investigación como **campo de acción** el proceso de desarrollo de *software* con tecnología multimedia para el apoyo del tema Programación Lineal en la asignatura Investigación de Operaciones.

El **objetivo general** de este trabajo es desarrollar una multimedia para apoyar el proceso de enseñanza-aprendizaje del tema Programación Lineal en la asignatura Investigación de Operaciones.

Los **objetivos específicos** a cumplir son:

- Construir los referentes teóricos fundamentales que sustentan la investigación.
- Realizar un estudio de las tendencias y tecnologías para el desarrollo de la multimedia.
- Realizar la implementación del *software* multimedia.
- Realizar pruebas de *software* a la propuesta de solución.

Las **tareas** a realizar para dar solución a los objetivos son:

- Estudio del estado del arte de los principales enfoques acerca de los procesos de desarrollo de aplicaciones con tecnología multimedia.
- Estudio detallado de las metodologías existentes para un proceso de modelación de una aplicación con tecnología multimedia.
- Comparación de las metodologías teniendo en cuenta sus características, ventajas y desventajas.

- Selección de una adecuada metodología, que facilite la creación y garantice la calidad del sistema.
- Estudio detallado de las herramientas que existen para la creación de una aplicación con tecnología multimedia.
- Comparación de las herramientas en cuanto a las características, ventajas y desventajas.
- Selección de las herramientas adecuadas para el proceso de creación de la aplicación.
- Desarrollo de la aplicación con las herramientas seleccionadas.
- Pruebas funcionales al producto multimedia desarrollado.

Defendiendo como idea que con el desarrollo de una multimedia que muestre de manera organizada, interactiva y con calidad la información referente a la Programación Lineal, entonces se obtendrá un material que sirva de apoyo al proceso de enseñanza-aprendizaje de dicho tema en la asignatura Investigación de Operaciones.

El **resultado** esperado de esta investigación es:

- Obtención de una multimedia para el apoyo al proceso de enseñanza-aprendizaje en la asignatura Investigación de Operaciones, específicamente en el tema de Programación Lineal.
- Realización de una correcta documentación del producto, garantizando con esta el futuro mantenimiento del sistema.

Los **métodos científicos** utilizados en la investigación estuvieron determinados por el objetivo general y las tareas de investigación previstas. A nivel teórico fueron utilizados los métodos: **analítico-sintético**, para realizar un estudio bibliográfico profundo de la teoría existente alrededor del objeto de estudio, y a partir del mismo determinar las características que tendrá la solución propuesta, además para determinar cuáles son las tecnologías y herramientas más adecuadas para el desarrollo de la solución; y el **análisis histórico-lógico**, para analizar definiciones de multimedia expresadas por distintos autores, y realizar un estudio de la evolución de estos productos desde su surgimiento hasta la actualidad. También fueron utilizados métodos empíricos, como es el caso de la **observación**, con el objetivo de tomar las mejores prácticas de sistemas similares con el mismo fin de la multimedia a desarrollar, así como las vulnerabilidades para convertirlas en ventajas de la aplicación resultante de esta investigación.

Para una mejor comprensión de la investigación, cuyo diseño metodológico se acaba de describir, se decide definir una **estructura capitular** que permita cierto grado de organización y facilite el estudio de la investigación. Los capítulos que la conforman son:

Capítulo 1: Fundamentación Teórica

En este capítulo se realiza la fundamentación del tema donde se explican algunos conceptos generales acerca del mismo, las tecnologías sobre aplicaciones multimedia, se analizan soluciones similares, además de explicar las herramientas, metodologías y lenguajes utilizados para desarrollar la aplicación.

Capítulo 2: Descripción de la solución propuesta.

El capítulo está dedicado a las dos primeras fases de la metodología de desarrollo XP definidas para la investigación: exploración y planificación, donde se describe la estructura interna del sistema mediante los artefactos generados por la metodología de desarrollo utilizada. En la fase exploración se plantean a grandes rasgos historias de usuarios que son de interés para la primera entrega del producto, el equipo de desarrollo se familiariza con las herramientas, tecnologías y prácticas que se utilizarán en el proyecto. Durante la planificación el cliente establece la prioridad de cada historia de usuario y los programadores realizan una estimación del esfuerzo para implementarlas, se realiza el Plan de Iteraciones y el Plan de Entregas del producto. El éxito del proyecto depende en gran medida de las decisiones tomadas en estas fases.

Capítulo 3: Construcción de la solución propuesta.

En este capítulo se realiza la implementación de la solución propuesta haciendo uso de la metodología y el lenguaje seleccionado. Se representan los artefactos principales que servirán de fundamento al proceso de desarrollo del producto incluyendo los diagramas de presentación, se construyen las tarjetas CRC (Clase, Responsabilidad y Colaboración) que forman parte de la fase de diseño de la metodología seleccionada y por último la descripción de las pruebas.

Capítulo 1

Fundamentación teórica

Capítulo 1 Fundamentación Teórica

Introducción

En este capítulo se abordan conceptos referentes a las tecnologías multimedia con algunas de las características más importantes que poseen y las ventajas que aporta a la esfera educacional. Se realiza una breve reseña en general del proceso de enseñanza-aprendizaje de la Investigación de Operaciones en nuestra Universidad. Se realiza un análisis de soluciones existentes. Además se identifican y caracterizan las herramientas, metodologías de desarrollo de *software*, los lenguajes de modelado y de programación a utilizar en la aplicación.

1.1 Multimedia

La integración de texto escrito, gráficas, imagen y sonido, la digitalización y la interactividad, han permitido que aparezcan diversas tecnologías. Éstas pueden ser de expresión, comunicación, información, sistematización y documentación, para dar lugar a aplicaciones en la educación, la diversión y el entretenimiento, la información, la comunicación, la capacitación y la instrucción. Esta integración ha facilitado una nueva tecnología de tipo digital que emplea la computadora, sus sistemas y periféricos, conocida generalmente como multimedia.

El concepto de multimedia se ha manejado desde hace mucho tiempo en varios campos, desde la educación hasta los medios audiovisuales. Es por ello que se dice que el término multimedia es muy difícil definir en pocas palabras, ya que existen diferentes definiciones del concepto multimedia, según distintos autores.

La palabra multimedia se puede entender como múltiples medios. Esto nos lleva a la definición de multimedia, que consiste en la integración de medios como gráficos de alta calidad, animaciones, sonido y video por medio de un computador. (Molina, 2010)

En el ámbito de la informática, la expresión multimedia se utiliza para calificar y describir las cualidades de diversos sistemas, aplicaciones, documentos y productos que emplean una combinación de textos, gráficos, imágenes, vídeos, animaciones y sonidos. Además incluyen hipervínculos, para permitir a los usuarios desplazarse por la información de forma intuitiva e interactiva. (Pinto, 2011)

Por otra parte una multimedia, podría ser denominada como una integración libre de tecnología que extiende y expande la forma en que interactuamos con una computadora, concepto que enriquece y amplía la interacción hombre-máquina, hoy en día lo vemos manifestado en

diversas aplicaciones que incluyen enciclopedias históricas, aventuras científicas animadas, libros de cuentos y novelas interactivas. (Rosas de Maidana, 2011)

En fin, en la actualidad se suele identificar como multimedia a la integración de dos o más medios de comunicación que pueden ser controlados o manipulados por el usuario en una computadora. O sea, es un sistema informático interactivo, que integra diferentes medios como el texto, el vídeo, la imagen, el sonido y las animaciones, que hacen que el usuario, mediante un ambiente agradable, se pueda informar, entretener e instruir.

Otros términos importantes asociados al término multimedia y que se relacionan mucho con este son:

- Hipertexto.
- Hipermedia.

1.1.1 Hipertexto e Hipermedia

Los términos “hipertexto” e “hipermedia” fueron acuñados por Ted Nelson en la segunda mitad de la década de los años 60 cuando trabajaba en su proyecto Xanadú. El proyecto consistía en construir un documento global único que contuviera todo el conocimiento existente mediante un sistema de enlaces. A este concepto de escritura ramificada y no lineal Nelson le denominó hipertexto. El concepto de hipermedia es una extensión del concepto de hipertexto en el que hay cabida para diferentes formatos y modos de representación, no sólo textos, sino también gráficos, imágenes, vídeos, sonido y enlaces que vinculan unas informaciones con otras, lo que permite ir ampliando la información de manera progresiva. (Vera, 2013)

El hipertexto es un conjunto de nodos de significaciones interrelacionados por conexiones entre palabras, páginas, fotografías, imágenes, gráficos, secuencias sonoras, etc. (De Oliveira, 2012)

Un buen sistema hipertexto estimula el ojeo y la búsqueda, distintos de la lectura de principio a fin. Cuando las conexiones llevan a gráficos, cuadros, secuencias de vídeo o música, el proceso se transforma en hipermedia. Por tanto, el término hipermedia viene a definir sencillamente las aplicaciones hipertexto que incluyen gráficos, audio y vídeo. (Salinas, 2007)

En rigor, el término multimedia es redundante, ya que media es en sí un plural. Hay autores que prefieren utilizar el término hipermedia en vez de multimedia. Hipermedia sería, de este modo, simplemente un hipertexto multimedia. Los documentos hipermedia pueden contener la capacidad de generar textos, gráficos, animación, sonido o vídeo en movimiento. (Salinas, 2007)

Hay que tener en cuenta que no todas las aplicaciones multimedia son hipermedia, ya que pueden ser una simple presentación de pantallas en orden secuencial. No obstante, a menudo se utilizan indistintamente los términos hipermedia y multimedia, ya que se entiende que una buena aplicación multimedia debe ser en realidad hipermedia. (Lamarca, 2013)

Además de estos términos (multimedia, hipertexto, hipermedia) existen otros los cuales resultan de gran importancia conocer sus definiciones, dado lo que significan para los mencionados anteriormente. Estos son texto, imagen, gráfico, animación, sonido y vídeo.

Texto

Se expresan mediante secuencias o cadenas de caracteres representados en la pantalla de la computadora para ser apreciados visualmente.

Según Adelaide Bianchini, en las aplicaciones de tipo multimedia los textos deben cumplir las siguientes condiciones: (Bianchini, 1992)

- Brevedad. Se debe utilizar la menor cantidad de palabras en los textos, eliminando palabras innecesarias y redundantes, e incorporando gráficos y sonidos que ilustren el posible contenido del texto.
- Tipo, estilo y tamaño de letras para mejorar la legibilidad. Los tamaños y estilos de letras deben resaltar palabras, dar importancia y diferenciar los subtítulos y títulos del resto del texto y evidenciar prioridades.
- La justificación del texto puede cambiar la categoría de la información. Si se utiliza justificación a ambos márgenes se define una información formal.

Imagen

La imagen se concibe como un soporte de la comunicación visual que materializa un fragmento de medio óptico susceptible de persistir a través del tiempo, y que constituye uno de los principales componentes de los medios de comunicación masivos. (Castellanos, 2003)

Las imágenes pueden ser de diferentes formatos: GIF, JPG, PNG, etc.

Gráfico

Cualquier cosa que pueda ser presentada en la pantalla de un computador puede emplearse como un gráfico en un documento multimedia. (Molina, 2010)

El diseño gráfico tiene como finalidad presentar textos e imágenes para mejorar la comunicación. (Bianchini, 1992). Según Bianchini esta comunicación será efectiva si se consideran las siguientes características:

- El estilo visual de los gráficos debe mantenerse consistente y encajar de una manera adecuada en toda la aplicación.
- Todos los gráficos e imágenes deben tener la misma resolución y calidad de elaboración.
- Cuando un gráfico o imagen representa o describe lo descrito en algún texto, se debe descartar el texto y dedicarse a la mejor utilización del arte gráfico: "Una imagen dice más que mil palabras".

Animación

Conjunto de imágenes que se colocan en forma secuencial para generar movimiento. Generalmente son utilizadas para efectuar demostraciones o simulaciones. Existen dos tipos de animaciones: las animaciones planas que están íntimamente relacionadas con los dibujos animados clásicos, y las animaciones en 3D las cuales están más relacionadas con la generación de Realidad Virtual.

Si se elabora una serie de gráficos con pequeñas diferencias entre ellos, y se los pasa por la pantalla en rápida sucesión, se tendrá una animación. (Molina, 2010)

Sonido

La capacidad de generar sonidos es una de las características más sobresalientes de una multimedia, ya que el sonido tiene la capacidad de atraer aún más que la imagen para resaltar cosas importantes. (Molina, 2010)

El sonido es un poderoso recurso que se puede utilizar en las aplicaciones para adornar y llamar la atención del usuario. Sin embargo, todos los excesos tienen problemas. La utilización de sonido debe restringirse y considerar las siguientes condiciones de uso: (Bianchini, 1992)

- Repeticiones de sonidos. Este recurso se utiliza para informar al usuario del cambio de un modo o escenario dentro de una aplicación, para indicar la ocurrencia de algún error, para advertirle acerca de alguna operación incorrecta o peligrosa. Sin embargo la constante repetición de un mismo sonido puede resultar molesto al usuario.
- Controlar el sonido. El usuario debe tener control suficiente para habilitar o deshabilitar los sonidos asociados a la aplicación, se le debe dar el control y la posibilidad de bajar o subir el volumen de tales sonidos.

- Discreción en el uso de sonido. La utilización de sonido debe estar asociada al esquema y estilo de la aplicación. Se debe recordar que los elementos multimedia deben motivar al usuario y mejorar su capacidad de adquisición de conocimiento y otras habilidades intelectuales, no deben ser ofensivos, ni intimidar al usuario.

Vídeo

Un video es un medio ideal para mostrar los atributos dinámicos de un concepto o proceso, en los cuales no alcanza con mostrar una descripción escrita del proceso o imágenes estáticas del mismo. (Molina, 2010)

El video da vida a la multimedia y lo convierte en un sistema completo, para fines educativos, de presentación y recreativos. Es una tecnología utilizada para capturar, grabar, procesar, transmitir y reproducir una secuencia de imágenes representativas de una escena que se encuentra en movimiento.

1.2 Multimedia Educativa

El uso de los multimedia en la educación y la formación ha provocado cambios en el proceso de aprendizaje. El estudiante deja de tener una actitud pasiva en el proceso de aprendizaje para adoptar un papel activo. La multimedia no debe pretender sustituir los tradicionales métodos de aprendizaje sino convertirse en un elemento que enriquezca este proceso de enseñanza-aprendizaje con los recursos docentes que se incorporan.

Los multimedia o Multimedia Educativas (ME), forman parte del *software* educativo y muchos lo definen como un objeto o producto que usa una combinación de medios: texto, color, gráficas, animaciones, video, sonido, en un mismo entorno, donde el estudiante interactúa con los recursos para mejorar el proceso enseñanza–aprendizaje. (Vidal, 2010)

Sus características básicas son:

- Versatilidad (adaptación a diversos contextos).
- Originalidad y uso de la tecnología avanzada.
- Capacidad de motivación.
- Calidad del entorno audiovisual.
- Calidad de los contenidos.
- Navegación e interacción.
- Adecuación a los usuarios y a su ritmo de trabajo.
- Potencialidad de los recursos didácticos.

- Incentiva el autoaprendizaje y auto iniciativa.

Además, estos multimedios tienen múltiples ventajas que conducen a elevar la calidad del proceso enseñanza-aprendizaje. Según un estudio realizado por el Dr. Pere Marqués Graells se tienen algunas de las principales ventajas de la multimedia en el ámbito educacional. (Marqués, 2010)

- Interés y motivación. Los alumnos están muy motivados y la motivación es uno de los motores del aprendizaje, ya que incita a la actividad y al pensamiento. Por otro lado, la motivación hace que los estudiantes dediquen más tiempo a trabajar y, por tanto, es probable que aprendan más.
- Interacción. Los estudiantes están permanentemente activos al interactuar con el ordenador y mantienen un alto grado de implicación en el trabajo. La versatilidad e interactividad del ordenador y la posibilidad de dialogar con él, les atrae y mantiene su atención.
- Desarrollo de la iniciativa. La constante participación por parte de los alumnos propicia el desarrollo de su iniciativa ya que se ven obligados a tomar continuamente nuevas decisiones ante las respuestas del ordenador a sus acciones. Se promueve un trabajo autónomo riguroso y metódico.
- Múltiples perspectivas e itinerarios. Los hipertextos permiten la exposición de temas y problemas presentando diversos enfoques, formas de representación y perspectivas para el análisis, lo que favorece la comprensión y el tratamiento de la diversidad.
- Aprendizaje a partir de los errores. La retroalimentación inmediata a las respuestas y a las acciones de los usuarios permite a los estudiantes conocer sus errores justo en el momento en que se producen y generalmente el programa les ofrece la oportunidad de ensayar nuevas respuestas o formas de actuar para superarlos.
- Facilitan la evaluación y control. Liberan al profesor de trabajos repetitivos. Al facilitar la práctica sistemática de algunos temas mediante ejercicios de refuerzo sobre técnicas instrumentales, presentación de conocimientos generales, prácticas sistemáticas de ortografía. Los ordenadores proporcionan informes de seguimiento y control. Facilitan la autoevaluación del estudiante.
- Actividades cooperativas. El ordenador propicia el trabajo en grupo y el cultivo de actitudes sociales, el intercambio de ideas, la cooperación y el desarrollo de la personalidad. El trabajo en grupo estimula a sus componentes y hace que discutan sobre la mejor solución para un problema, critiquen, se comuniquen los descubrimientos. Además aparece más

tarde el cansancio, y algunos alumnos razonan mejor cuando ven resolver un problema a otro que cuando tienen ellos esta responsabilidad.

- Contacto con las nuevas tecnologías y el lenguaje audiovisual. Estos materiales proporcionan a los alumnos y a los profesores un contacto con las TIC, generador de experiencias y aprendizajes. Contribuyen a facilitar la necesaria alfabetización informática y audiovisual.

En general, la multimedia educativa se ha convertido en una tecnología de gran impacto en el proceso de enseñanza-aprendizaje, en el ámbito educacional. Estos programas brindan una adecuada formación de habilidades y capacidades en los estudiantes, sirviendo como un medio de apoyo para la docencia en los diferentes niveles de enseñanzas.

1.3 Investigación de Operaciones (IO)

Como su nombre lo indica, el objetivo de esta disciplina implica investigar sobre las operaciones. En consecuencia, esta disciplina se aplica a la problemática relacionada con la conducción y la coordinación de actividades en una organización. En esencia, la naturaleza de la organización es irrelevante, por lo cual la IO ha sido aplicada de manera extensa en áreas tan diversas como manufactura, transporte, construcción, telecomunicaciones, planeación financiera, cuidado de la salud, fuerzas armadas y servicios públicos, por nombrar sólo unas cuantas. Así, la gama de aplicaciones es inusualmente amplia. (Hillier, 2006)

En la Universidad de las Ciencias Informáticas, se imparte la asignatura Investigación de Operaciones a los estudiantes de tercer año, en el segundo semestre. La asignatura tiene un carácter eminentemente práctico, aunque se sustenta sobre una base teórica que permite, ante el planteamiento de un problema, que el estudiante sea capaz de identificar los datos necesarios, reconocer sus características, plantearlo, darle solución por un método adecuado y analizar los resultados obtenidos; estos problemas deben estar referidos a situaciones prácticas, preferiblemente vinculadas a la carrera. Por estas razones se identifican como objetivos fundamentales:

- Contribuir a que los estudiantes utilicen y desarrollen una forma dialéctica de pensamiento y la apliquen consecuentemente en su enfoque sistémico de análisis.
- Consolidar en los estudiantes un estilo de trabajo que propicie una actuación independiente y creativa para la solución de los problemas que enfrentará, considerando las limitaciones que existan en el medio de trabajo.

- Identificar los elementos que caracterizan a los problemas de Programación Lineal y sus casos particulares como la Programación Discreta.
- Construir modelos matemáticos asociados a problemas de Programación Lineal y Programación Discreta.
- Analizar e interpretar la solución óptima de modelos matemáticos de Programación Lineal (casos generales y particulares) y utilizar las facilidades que brinda el análisis post-optimal.
- Utilizar eficientemente el *software* disponible para la solución de problemas de investigación de operaciones.

1.4 Programación Lineal en la IO

La Programación Lineal utiliza un modelo matemático para describir el problema. El adjetivo lineal significa que todas las funciones matemáticas del modelo deben ser funciones lineales. En este caso, la palabra programación no se refiere aquí a términos computacionales; en esencia es sinónimo de planeación. Por lo tanto, la Programación Lineal involucra la planeación de actividades para obtener un resultado óptimo; esto es, el resultado que mejor alcance la meta especificada entre todas las alternativas factibles. Este problema consiste en elegir el nivel de ciertas actividades que compiten por recursos escasos necesarios para realizarlas. Después, los niveles de actividad que se eligen dictan la cantidad de recursos que consumirá cada una de ellas. La variedad de situaciones a las que se puede aplicar esta descripción es sin duda muy grande. Sin embargo, el ingrediente común de todas estas situaciones es la necesidad de asignar recursos a las actividades mediante la elección de los niveles de éstas. (Hillier, 2006)

En la UCI, un tema de gran importancia dentro de la asignatura Investigación de Operaciones, es precisamente el tema que está relacionado con los problemas de la Programación Lineal, tema de especial importancia para el estudiante a la hora de tomar decisiones.

Entre sus objetivos fundamentales se encuentran los siguientes:

- Identificar los elementos que caracterizan a los problemas de Programación Lineal para poder reconocerlos en el contexto de una situación dada y acotar el problema.
- Formular modelos matemáticos asociados a problemas de Programación Lineal para luego aplicar un método de solución adecuado.

Logrando que el alumno cree habilidades tales como:

- Interpretar los supuestos de la Programación Lineal.
- Reconocer las variables de decisión de un problema de Programación Lineal.
- Clasificar problemas de Programación Lineal según el dominio de las variables de decisión.

- Identificar y formular los diferentes tipos de restricciones que se pueden presentar en un problema de Programación Lineal.
- Determinar la función objetivo de un problema de Programación Lineal.

1.5. Análisis de soluciones similares

Debido al exceso de contenido existente referente a la asignatura Investigación de Operaciones y la necesidad de crear un producto educativo que motive al estudiante, surgen las aplicaciones con tecnología multimedia como la más adecuada para resolver estos inconvenientes. La integración de sus medias: sonido, texto, videos e imágenes posibilitan que el proceso de enseñanza y aprendizaje sea más ameno y dinámico. Es por ello que el estudio de casos similares está basado en multimedia que apoyen este proceso en dicha asignatura.

Según el estudio realizado se encontró a nivel internacional un *software* multimedia de apoyo a los procesos de enseñanza-aprendizaje para la resolución de problemas en Investigación de Operaciones "Solución Gráfica de Máximos", desarrollada en la Facultad de Ingeniería Uniminuto (Colombia). Este proyecto busca consolidar ambientes de aprendizaje virtuales para la resolución de problemas matemáticos. Su objetivo es ser utilizado como soporte pedagógico y didáctico en la asignatura de Investigación de Operaciones en el tema de "Resolución de Problemas de Máximos en Forma Gráfica". (Castro, 2010)

En Cuba, específicamente en la Facultad de Ingeniería Industrial de la CUJAE, se cuenta con un conjunto de herramientas de ayuda al aprendizaje de la Investigación de Operaciones. Entre las herramientas que se han diseñado se encuentran: (Pérez, 2008)

- Multimedia de Aprendizaje de la Modelación Matemática. (Ver Anexo #1)
- Modele la Vida: Multimedia para la Ejercitación de la Modelación Matemática. (Ver Anexo #2)
- Simula: Multimedia para el aprendizaje y ejercitación de la Simulación. (Ver Anexo #3)
- Glosario de términos de la Investigación de Operaciones. (Ver Anexo #4)

El objetivo de estos multimedios interactivos como herramientas para el aprendizaje complementario de la Investigación de Operaciones, es permitir un rápido aprendizaje por parte del estudiante y mayor motivación por la asignatura.

En la Universidad de las Ciencias Informáticas (UCI), no se cuenta exactamente con una multimedia que ayude al aprendizaje de esta asignatura. Sin embargo existe un objeto de aprendizaje (OA), el cual representa un sitio web que se utiliza como herramienta de apoyo al

proceso de enseñanza-aprendizaje de la asignatura de Investigación de Operaciones. (López, 2013)

El sitio web contiene actividades de los cuatro temas que incluye la asignatura en la UCI. En cada uno de los temas se abordan de forma muy breve los elementos teóricos esenciales, es por ello que no brinda solución al objetivo planteado, que es poner en manos de los estudiantes un *software* interactivo que sirva de apoyo a la enseñanza sobre la Programación Lineal.

En conclusión el análisis de soluciones existentes nacionales e internacionales, evidenció que existen varios *software* con tecnología multimedia que apoyan de una manera u otra el proceso de enseñanza-aprendizaje en la asignatura Investigación de Operaciones. Estos a pesar de abordar temas relacionados con la Programación Lineal, no cumplen con el objetivo de la investigación, ya que no abarcan todo lo referente a este tema en específico. Además los contenidos tratados no están acordes con el programa de estudio de la asignatura en la Universidad. Por otra parte la información ofrecida en muchos de los casos no incluye elementos demostrativos, como imágenes, animaciones y videos, acerca del contenido, de manera que la información sea más completa.

1.6. Estudio de las metodologías

Según Pressman, el *software* se ha convertido en el elemento clave de la evolución de los sistemas y productos basados en computadoras, así como en una de las tecnologías más importantes en el ámbito mundial. La intención de la ingeniería de *software* es proporcionar un marco general para construir *software* con una calidad mucho mayor. (Pressman, 2005)

Se considera que el uso de las metodologías dentro de esta disciplina resulta una buena práctica para guiar el proceso de desarrollo de un *software*. El éxito del producto depende en gran parte de la metodología escogida, ya sea tradicional o ágil. Por estos motivos se decide hacer un estudio de las principales metodologías existentes con el objetivo de seleccionar la más adecuada para la investigación.

1.6.1 Proceso Unificado Racional (RUP)

Los verdaderos aspectos definitorios del Proceso Unificado se resumen en tres frases claves: dirigido por casos de uso, centrado en la arquitectura, e iterativo e incremental. Esto es lo que hace único al Proceso Unificado. RUP es un marco de trabajo genérico que puede especializarse para una gran variedad de sistemas de *software*, para diferentes áreas de aplicación, diferentes tipos de organizaciones, diferentes niveles de aptitud y diferentes tamaños de proyecto. Utiliza UML, siendo éste una parte esencial del mismo, además de estar

basado en componentes. Establece el contexto de las cuatro "P" del desarrollo de *software*: personas, proyecto, producto y proceso, prestándoles especial atención. (Jacobson y otros, 2000)

Cada ciclo de RUP se desarrolla a lo largo del tiempo. Este tiempo, a su vez se divide en 4 fases:

- Fase de inicio.
- Fase de elaboración.
- Fase de construcción.
- Fase de transición.

RUP propone 6 flujos de trabajo, los cuales pueden o no estar presentes en las 4 fases de vida del proyecto.

- Modelado del Negocio
- Requisitos
- Análisis y diseño
- Implementación
- Pruebas
- Despliegue

Propone también tres flujos de apoyo.

- Gestión de configuración y cambios
- Gestión de proyecto
- Ambiente o entorno

1.6.2 Microsoft Solution Framework (MSF)

Al igual que RUP, ésta se encuentra entre las principales metodologías tradicionales, las cuales centran su atención en llevar una documentación exhaustiva de todo el proyecto y en cumplir con un plan de proyecto que se define en la fase inicial del desarrollo del mismo. MSF es un compendio de las mejores prácticas en cuanto a administración de proyectos se refiere. Más que una metodología rígida de administración de proyectos, MSF es una serie de modelos que puede adaptarse a cualquier proyecto de tecnología de información. (Figuroa y otros, 2008)

Todo proyecto es separado en cinco principales fases:

- Visión y Alcance.
- Planificación.

- Desarrollo.
- Estabilización.
- Implantación.

La metodología propuesta por MSF tiene como propósito lograr entregas con un margen amplio de éxito, basados en la calidad del producto *software*, teniendo presente las necesidades del cliente y el principio de flexibilidad, así como el cumplimiento con los compromisos adquiridos, la gestión de los costos y la minimización de los riesgos inherentes en todo proyecto de desarrollo de *software*. (Pérez, 2011)

1.6.3 Programación Extrema (XP)

Es la más destacada de los procesos ágiles de desarrollo de *software* formulada por Kent Beck. La programación extrema se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad que en la previsibilidad. (Figueroa y otros, 2008).

XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de *software*, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. Esta metodología se basa en la realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico. Los principios y prácticas son de sentido común pero llevadas al extremo, de ahí proviene su nombre. (Letelier, 2006)

Entre las características esenciales de XP aparecen: (Letelier, 2006)

Las Historias de Usuario: Son la técnica utilizada en XP para especificar los requisitos del *software*. Se trata de tarjetas de papel en las cuales el cliente describe brevemente las características que el sistema debe poseer, sean requisitos funcionales o no funcionales. El tratamiento de las historias de usuario es muy dinámico y flexible, en cualquier momento historias de usuario pueden romperse, reemplazarse por otras más específicas o generales, añadirse nuevas o ser modificadas. Cada historia de usuario es lo suficientemente comprensible y delimitada para que los programadores puedan implementarla en unas semanas.

Roles XP: Entre los roles que propone XP se tienen los siguientes:

- Programador.
- Cliente.
- Encargado de pruebas (*Tester*).
- Encargado de seguimiento (*Tracker*).
- Entrenador (*Coach*).
- Consultor.
- Gestor (*Big boss*).

Proceso XP: Un proyecto XP tiene éxito cuando el cliente selecciona el valor de negocio a implementar basado en la habilidad del equipo para medir la funcionalidad que puede entregar a través del tiempo. El ciclo de desarrollo consiste en los siguientes pasos:

- 1) El cliente define el valor de negocio a implementar.
- 2) El programador estima el esfuerzo necesario para su implementación.
- 3) El cliente selecciona qué construir, de acuerdo con sus prioridades y las restricciones de tiempo.
- 4) El programador construye ese valor de negocio.
- 5) Vuelve al paso 1.

En todas las iteraciones de este ciclo tanto el cliente como el programador aprenden.

El ciclo de vida ideal de XP consiste de seis fases:

- Exploración.
- Planificación de la Entrega.
- Iteraciones.
- Producción.
- Mantenimiento.
- Muerte del Proyecto.

Para una mejor estructuración de la investigación y un mejor entendimiento, se decidió agrupar las seis fases en cuatro, sin violar el ciclo de vida de la metodología, estas fases son:

- Fase I: Exploración
- Fase II: Planificación
- Fase III: Implementación
- Fase IV: Pruebas

Prácticas XP: La aplicación disciplinada de las prácticas posibilita disminuir la mítica curva exponencial del costo del cambio a lo largo del proyecto.

- El juego de la planificación: Es un espacio frecuente de comunicación entre el cliente y los programadores.
- Entregas pequeñas: La idea es producir rápidamente versiones del sistema que sean operativas.
- Diseño simple: Se debe diseñar la solución más simple que pueda funcionar y ser implementada en un momento determinado del proyecto.
- Pruebas: La producción de código está dirigida por las pruebas unitarias.
- Refactorización: La refactorización es una actividad constante de reestructuración del código con el objetivo de remover duplicación de código, mejorar su legibilidad, simplificarlo y hacerlo más flexible para facilitar los posteriores cambios.
- Programación en parejas: Toda la producción de código debe realizarse con trabajo en parejas de programadores.
- Propiedad colectiva del código: Cualquier programador puede cambiar cualquier parte del código en cualquier momento.
- Integración continua: Cada pieza de código es integrada en el sistema una vez que esté lista. Así, el sistema puede llegar a ser integrado y construido varias veces en un mismo día.
- 40 horas por semana: Se debe trabajar un máximo de 40 horas por semana.
- Cliente in-situ: El cliente tiene que estar presente y disponible todo el tiempo para el equipo.
- Estándares de programación: XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación.
- Comentarios respecto de las prácticas: El mayor beneficio de las prácticas se consigue con su aplicación conjunta y equilibrada puesto que se apoyan unas en otras.

1.6.4 SCRUM

Es un proceso ágil y liviano que sirve para administrar y controlar el desarrollo de *software*. El desarrollo se realiza en forma iterativa e incremental. Cada ciclo o iteración termina con una pieza de *software* ejecutable que incorpora nueva funcionalidad. Esta metodología se utiliza como marco para otras prácticas de ingeniería de *software* como RUP o Programación Extrema.

SCRUM se focaliza en priorizar el trabajo en función del valor que tenga para el negocio, maximizando la utilidad de lo que se construye y el retorno de inversión. Está diseñado

especialmente para adaptarse a los cambios en los requerimientos, por ejemplo en un mercado de alta competitividad. Los requerimientos y las prioridades se revisan y ajustan durante el proyecto en intervalos muy cortos y regulares. De esta forma se puede adaptar en tiempo real el producto que se está construyendo a las necesidades del cliente. Se busca entregar *software* que realmente resuelva las necesidades, aumentando la satisfacción del cliente. (Figueroa y otros, 2008)

SCRUM promueve la colaboración con el cliente en lugar de rígida negociación de contratos. Por lo cual, es importante tener capacidad de respuesta para los cambios en lugar de seguir estrictamente una planificación, partiendo del principio que el proyecto *software* es cambiante. El propósito es que el cliente vaya observando los resultados, pueda decidir cambios en la marcha o incluso darle un giro completo al proyecto. En conclusión, la metodología SCRUM, ofrece herramientas que permiten gestionar el equipo de trabajo hasta el punto de proponer tiempos para el proceso de desarrollo de *software* y para las reuniones del equipo, con la finalidad de asegurar el cumplimiento de los objetivos del proyecto. (Pérez, 2011)

1.7 Selección de la metodología para dar solución al problema

En consideración al estudio realizado se decide escoger una metodología ágil ya que estas se adaptan perfectamente con la solución, pues entre sus principios se destaca que es más importante crear un producto *software* que funcione que escribir documentación exhaustiva. Entre ellas se decide escoger XP porque es la más destacada de los procesos ágiles de desarrollo de *software*, se cuenta con un equipo de desarrollo pequeño y un período corto de entrega. La metodología XP se destaca por el hecho de realizar entregas a corto plazo con una rapidez y calidad increíbles. Además se decide utilizar XP debido a que se adapta en gran medida, tanto al tipo de proyecto a desarrollar como a las condiciones de trabajo, ya que el cliente forma parte del equipo de desarrollo.

Dentro de las metodologías ágiles analizadas XP destaca como una de las que mayor índice de agilidad posee. XP resalta también por contar con la mayor cantidad de información disponible y es con diferencia la más popular. (Canós, Letelier y Penadés, 2003)

La solución a desarrollar posee características que hacen que se ajuste a un proceso de desarrollo ágil:

1. Se cuenta con un corto período de tiempo para el desarrollo de la solución.
2. El cliente forma parte del equipo de desarrollo.
3. La dimensión de la solución es pequeña (es una 1ra versión).

4. Se necesita realizar pequeñas liberaciones para ser aprobadas por el cliente y recibir retroalimentación del mismo, lo cual requiere de un proceso de desarrollo preferiblemente ágil.

1.8 Lenguajes de Modelado

1.8.1 Lenguaje Unificado de Modelado (UML)

UML es el lenguaje más conocido y utilizado en la modelización conceptual. Esta impulsado desde la Object Management Group (OMG), una organización encargada de promover estándares relacionados con los sistemas orientados a objetos. La definición de UML propuesta por la OMG es la siguiente:

El Lenguaje de Modelado Unificado (UML) es un lenguaje gráfico para visualizar, especificar, construir y documentar los artefactos de un sistema intensivo de *software*. UML ofrece una forma estándar de representar un plano del sistema, incluyendo la conceptualización de procesos de negocio y funciones del sistema, así como también esquemas de bases de datos y componentes de *software* reutilizables. (Hernández, 2012)

Tal como lo define la OMG, sus principales funciones son:

- Visualizar: Permite expresar de una forma gráfica un sistema, de forma que cualquier persona lo pueda entender.
- Especificar: Facilita especificar cuáles son las características de un sistema antes de su construcción.
- Construir: A partir de los modelos especificados se pueden construir los sistemas diseñados.
- Documentar: Documenta los propios elementos gráficos que se pueden utilizar como documentación del sistema desarrollado, lo cual sirve para su futura revisión.

A pesar que UML presenta grandes ventajas, también posee los siguientes inconvenientes: (Berzal, 2014)

- Falta de integración con otras técnicas. (Por ejemplo: diseño de interfaces de usuario)
- UML es excesivamente complejo (y no está del todo libre de ambigüedades); "el 80% de los problemas pueden modelarse usando alrededor del 20% de UML".

Además, aunque UML permite modelar casi cualquier sistema orientado a objetos, no soporta todos los aspectos de las aplicaciones multimedia de una forma adecuada e intuitiva.

1.8.2 Lenguaje de Modelado Orientado a Objetos de Aplicaciones Multimedia (OMMMA-L)

El Lenguaje de Modelado Orientado a Objetos de Aplicaciones Multimedia, constituye una extensión de UML, que integra especificaciones de sistemas multimedia. OMMMA-L en sus siglas, está sustentado en cuatro vistas fundamentales : vista lógica, vista de presentación espacial, vista de comportamiento temporal predefinido y la vista de control interactivo, donde cada una de ellas se asocia a un tipo de diagrama en particular.

- Vista Lógica: modelada a través del Diagrama de Clases de OMMMA-L, extendido del Diagrama de Clases de UML, utilizando las mismas notaciones, pero incorporando las clases correspondientes a las medias: media continua y media discreta, generalizadas en una clase medias. Divide en dos áreas dicho diagrama: una para la jerarquía de los tipos de media y otra para la modelación de la estructura lógica del dominio de la aplicación.
- Vista de Presentación Espacial: modelada a través de los Diagramas de Presentación de OMMMA-L, los cuales son de nueva aparición en la extensión de UML, dado que este último no contiene un diagrama apropiado para esta tarea. Estos diagrama tienen el propósito de declarar las interfaces de usuario con un conjunto de estructuras delimitadas en tamaño y área, dividiéndose en objetos de visualización (texto, gráfico, video, animación) e interacción (*scrolls*, barras de menú, botones, campos de entrada y salida, hipertextos con hipervínculos). Estos diagramas de presentación pueden ser divididos en capas virtuales de presentación donde en cada uno de ellas sólo se haga referencia a una clase específica de componentes (por ejemplo, una vista para los objetos de visualización y otra para los de interacción, u otro tipo de división para la representación de los intereses de los desarrolladores.)
- Vista de Comportamiento Temporal Predefinido: modelada por el Diagrama de Secuencia de OMMMA-L, extendido a partir del diagrama de secuencia de UML. El Diagrama de secuencia modela una secuencia de una presentación predefinida dentro de una escena, donde todos los objetos dentro de un diagrama se relacionan al mismo eje del tiempo. En este diagrama se hace un refinamiento del eje del tiempo con la introducción de marcas de tiempo a través de diferentes tipos de intervalos; marcas de inicio y fin de ejecución que permite soportar su reusabilidad; marcas de activación y desactivación de demoras en objetos de tipo media, posibilitando la modelación de las tolerancias de la variación de las restricciones de sincronización para los objetos media; activación compuesta de objetos media para la agrupación de objetos concurrentemente activos.

- Vista de Control Interactivo: modelado a través del Diagrama de Estado, extendido a partir del diagrama de estado de UML, sintácticamente igual a este último, mas con la diferencia semántica de que en el orden de unir los controles interactivos y predefinidos, no interrumpidos de los objetos, las acciones internas de estados simples tienen que llevar nombres de diagrama de secuencia en vez de diagramas de estado empotrados; queriendo esto decir que el comportamiento especificado por el diagrama de secuencia se provoca automáticamente cuando se entra al estado correspondiente donde se hace referencia. (Lorente, 2006)

OMMMA-L integra dos nuevos diagramas, el mapa de navegación que le da una idea al usuario de cómo será la navegación en el producto multimedia y el diagrama de presentación, que sirve para describir la parte estática del modelo a través de una descripción intuitiva de la distribución espacial de objetos visuales de la interfaz de usuario. (Martínez, 2010)

Para el modelado del sistema fue escogido OMMMA-L, como una extensión de UML dedicada específicamente al desarrollo de *software* con tecnología multimedia. El autor se decide por este lenguaje ya que UML, no soporta con sus vistas y diagramas todos los aspectos de las aplicaciones multimedia de una forma adecuada e intuitiva. Además de que no permite que se logre una representación eficiente de todos los elementos, puesto que carece de la integración con otras técnicas para el diseño de interfaz de usuario en este tipo de aplicaciones.

1.9 Tendencias y tecnologías actuales. Selección de las herramientas y lenguajes de desarrollo

1.9.1 ¿HTML5 o Flash?

Hasta hace poco tiempo, los desarrolladores que querían incluir elementos dinámicos y animaciones en su web, prácticamente sólo podían utilizar tecnología Flash. Esta tecnología creada por Macromedia en los años noventa y después adquirida por Adobe en 2005, revolucionó el mundo web y se convirtió en el estándar multimedia de Internet. Así ha sido hasta la llegada de HTML5, al cual muchos lo consideran como su sustituto natural y el nuevo estándar para multiplicar la interactividad de los proyectos web. Hoy en día, los desarrolladores tienen que plantearse cuál de las dos opciones utilizar, ya que ambas tecnologías rivalizan en prestaciones.

La filosofía de HTML5 aboga por los estándares abiertos, que son fundamentales para impulsar la innovación e introducir en la sociedad los beneficios de las nuevas tecnologías. Los estándares abiertos tienen la capacidad de mejorar nuestras vidas, y de ello son ejemplo las

nuevas funciones multimedia de HTML5. Debido al uso extendido de determinados complementos, como Flash, las nuevas funciones multimedia de HTML5 no pueden aspirar a sustituirlos por completo, pero facilitan otros métodos para presentar contenido multimedia. La intención es no depender tanto de los complementos externos y propietarios. (Franganillo, 2010)

HTML 5 proporciona un estándar para la reproducción y representación de contenido multimedia buscando unificar los formatos del contenido incluido, eliminando la necesidad de utilizar *plugins* externos y propietarios, e intentando obtener reproducciones más efectivas. La falta de soporte para Flash por parte de dispositivos utilizados para la navegación web como el iPad, lleva a muchos desarrolladores a optar por la utilización de HTML5 buscando promocionar sus sitios web a todo el público, sin importar el dispositivo que utilice el cliente para visualizar su sitio web. (Riquelme, 2012)

La propuesta de HTML5 puede sustituir funciones que hasta ahora sólo eran posibles con Flash. Para reproducir multimedia no hace falta ningún mecanismo complejo, por lo que es razonable pensar que HTML5 desplazará a Flash como contenedor de vídeo y audio, aunque no hay argumentos sólidos para pensar que éste vaya a desaparecer. (Allaire, 2010)

En el presente trabajo de diploma el autor se decide por el estándar creado por la W3C, para el desarrollo de la multimedia, ya que el HTML5 presenta algunas ventajas sobre el muy conocido Flash. Algunas de estas ventajas por la que se selecciona HTML5 es que la reproducción de vídeo y audio sea más eficiente, consuma menos recursos y se pueda gestionar con un código abierto y transparente, sin necesidad de instalar componentes adicionales. Además, los navegadores cada vez más se adaptan a las implementaciones de HTML5 demostrando que este estándar no solo se ve como el presente, sino como el futuro de la web.

1.9.2 Lenguajes de programación

Lenguaje de Marcado de Hipertexto (HTML)

La W3C define el lenguaje HTML como “Un lenguaje comúnmente utilizado para la publicación de hipertexto en la Web y desarrollado con la idea de que cualquier persona o tipo de dispositivo pueda acceder a la información en la Web. HTML utiliza etiquetas que marcan elementos y estructuran el texto de un documento.”(W3C, 2014)

La última versión de este lenguaje es la conocida HTML5, la misma está pensada con una mayor integración con los lenguajes CSS y JavaScript. Con HTML5, HTML provee los elementos estructurales, CSS se encarga de volver esa estructura utilizable y atractiva a la

vista, y JavaScript tendrá el poder necesario para proveer dinamismo y construir aplicaciones web completamente funcionales y con mayor capacidad de interacción con el usuario. (Gauchat, 2012)

HTML5 ha revolucionado la web y no solo se ve como el presente, sino como el futuro, por las numerosas novedades que trae con respecto a la versión anterior, entre las que se encuentran:

- Nuevas etiquetas que permiten representar elementos familiares de las páginas, tal es el caso de: `<header>` para el encabezado de las páginas, `<nav>` representa un grupo de artículos introductorios o de navegación, `<section>` se utiliza para representar una sección general dentro de un documento o aplicación, como un capítulo de un libro; y `<figure>` para asignar un título a una imagen.
- Incluye etiquetas para incorporar contenido multimedia como `<audio>` para audio y `<video>` para video, las cuales permiten la reproducción por parte del navegador de este tipo de contenido. Nuevos elementos que permitirán incrustar un contenido multimedia de sonido o de vídeo, respectivamente. Es una de las novedades más importantes e interesantes en este HTML5, ya que permite reproducir y controlar vídeo y audio sin necesidad de *plugins* como el de Flash.

La decisión de seleccionar HTML5 está sustentada en las novedosas herramientas que incorpora para el desarrollo web y el hecho de ser este un estándar establecido por la W3C. Esta nueva versión del lenguaje básico de la Web proporciona mecanismos para simplificar el trabajo y facilitar la inclusión de elementos multimedia.

Hojas de Estilo en Cascada (CSS)

Las hojas de estilo son complementos de código añadidos al HTML que se encargan de la apariencia del documento. Las CSS ofrecen propiedades para ampliar el lenguaje HTML en la representación visual de las páginas web. El lenguaje de programación utilizado por los desarrolladores para codificar dichas hojas, también es denominado CSS. Es el más conocido y utilizado para definir las propiedades de formato de los diferentes elementos HTML. (Lancker, 2009)

La última versión del lenguaje CSS es la versión 3 (CSS 3). La novedad más importante que aporta CSS 3 consiste en la incorporación de nuevos mecanismos para mantener un mayor control sobre el estilo con el que se muestran los elementos de las páginas, sin tener que recurrir a trucos, que a menudo complicaban el código de las web. (Álvarez, 2011)

CCS 3 ha incorporado valiosas novedades, entre las que se encuentran: (Luca, 2011)

- Novedades en cuanto al uso del color y de la opacidad.
- Características relacionadas con la interfaz de usuario.
- Nuevas alternativas para dibujar bordes con el uso de opciones tales como color, imágenes, y radio o redondeado.
- Novedades en el trabajo con fondos, con el uso de degradados y la posibilidad de incluir múltiples imágenes.
- Uso de sombras para texto (*text shadow*).
- Incorporación de muchas novedades en lo que se refiere a flujo de texto dentro del sitio (*text overflow*).
- Da solución al problema de las limitaciones con las tipografías, con el uso de *@fontface*
- Incorporación de transición y también funciones de animación.

La selección de CSS3 como lenguaje para escribir el estilo visual de la solución se hizo teniendo en cuenta que es un estándar determinado por la W3C, además de ser muy usado y difundido por maquetadores, diseñadores y desarrolladores web.

JavaScript

JavaScript es un lenguaje de programación interpretado, por lo que no es necesario compilar los programas para ejecutarlos. En otras palabras, los programas escritos con JavaScript se pueden probar directamente en cualquier navegador sin necesidad de procesos intermedios. A pesar de su nombre, no guarda ninguna relación directa con el lenguaje de programación Java. Legalmente, JavaScript es una marca registrada de la empresa Sun Microsystems. (Eguíluz Pérez, 2009)

El amplio uso de JavaScript en el desarrollo web se debe a su gran potencia y versatilidad, esto ha hecho que sea utilizado en otros entornos como es el caso del lenguaje ActionScript de Macromedia, que acompaña el sistema Flash. Como toda tecnología tiene disímiles ventajas, pero también posee limitaciones: (Valdelli, 2014)

Entre las ventajas más importantes de JavaScript se encuentran:

- Lenguaje de scripting seguro y fiable.
- Los script tienen capacidades limitadas, por razones de seguridad.
- El código JavaScript se ejecuta en el cliente.

Y dentro de las desventajas más notables de JavaScript se encuentran:

- El código es visible y puede ser leído por cualquier usuario.

- El código debe descargarse completamente.

JavaScript no es el único lenguaje que se encuentra en la web con características similares a las mencionadas. También existe su gran contrincante: Visual Basic Script. Este lenguaje, desarrollado por Microsoft, está basado en el lenguaje Basic de esta empresa, pero solo puede utilizarse en el navegador Internet Explorer. Si se quiere que páginas dinámicas puedan ser vistas desde cualquier navegador y sistema operativo, la elección más adecuada es sin lugar a dudas JavaScript. (Mann, 2014)

1.9.3 Framework de desarrollo

Biblioteca JavaScript

JQuery: Tras realizar una búsqueda en la web se pudo determinar que la biblioteca escrita en el lenguaje JavaScript denominada JQuery, posee un amplio uso en la Internet. Implementa una serie de clases que permite programar sin preocuparse del navegador con el que está visitando el usuario, ya que funcionan de forma exacta en las plataformas más habituales.

JQuery es libre y Open Source, actúa del lado del cliente y se centra en la interacción entre el DOM (Document Object Model), JavaScript, AJAX (Asynchronous JavaScript And XML) y HTML. El objetivo de esta biblioteca es simplificar los comandos comunes de JavaScript. (Lancker, 2012)

Dentro de sus características se encuentran:

- Permite manipular fácilmente el DOM de una página web.
- Permite manejar de una manera sencilla los eventos de los elementos que forman una página web.
- Permite modificar, eliminar o acceder a los estilos visuales de una página web.
- Permite el uso de AJAX.
- Puede ser extendido a través del uso de *plugins*.
- Posee compatibilidad con navegadores como Mozilla Firefox, Internet Explorer, Safari, Opera y Google Chrome.

Una de las características más importantes de éste, es su gratuidad, y es por ello uno de los más aceptados en el desarrollo de páginas web, posee una comunidad muy grande de desarrolladores que le dan soporte, es bien documentado y cuenta con componentes para la interfaz de usuario y galerías. La ventaja principal de jQuery es que es mucho más fácil que sus

competidores. Tiene una gran cantidad de *plugins* que pueden ser adicionados fácilmente, lo cual permite ahorrar en tiempo y esfuerzo. La versión a utilizar será la 2.1.0.

Framework CSS

El uso en la actualidad de marcos de trabajo que faciliten la labor del diseño web, se ha hecho muy común entre desarrolladores. Ello contribuye con un ahorro de tiempo considerable, debido a que es esta una de las tareas que más tiempo consume en el proceso de desarrollo de *software*. Ejemplo de esta tecnología, se encuentra a Twitter-Bootstrap.

Twitter-Bootstrap: Es un *framework* de código abierto cuyo objetivo es facilitar el desarrollo de aplicaciones o páginas web teniendo una colección de plantillas CSS, HTML y *plugins* JavaScript. Los diseños creados son simples, limpios e intuitivos. Con sólo agregar algunas clases y el *markup* correcto se pueden lograr casi sin esfuerzo grupos de botones, barras de navegación, formularios, etc., todo sin tener que escribir una línea de código CSS. Las aplicaciones que utilizan Twitter-Bootstrap adaptan la interfaz dependiendo del tamaño del dispositivo en el que se visualice de forma nativa, esto se denomina diseño adaptativo. (Lerner, 2012)

La versión a utilizar será la 3.0.0. Su selección para el desarrollo de la solución se hizo posible gracias a la facilidad de uso, el ahorro de tiempo considerable y la existencia de conocimientos previos sobre esta tecnología. Twitter-Bootstrap en la aplicación se utiliza principalmente para el estilo visual, posibilitando la compatibilidad con distintos navegadores y dispositivos en los que pueda ser visualizado el contenido.

1.9.4 Entorno de desarrollo integrado

Un IDE es un programa compuesto por una serie de herramientas, tales como editor de texto, compilador, intérprete, depurador, sistema de ayuda para la construcción de interfaces gráficas de usuario, etc., que utilizan los programadores para desarrollar código. Esta herramienta puede estar pensada para su utilización con un único lenguaje de programación o bien puede dar cabida a varios de estos.

Eclipse

Eclipse es un Entorno de Desarrollo Integrado de código abierto escrito en Java en un 90 % para desarrollar aplicaciones basadas en tecnologías Java. Inicialmente fue desarrollado por IBM, pero actualmente pertenece a la Fundación Eclipse. Su funcionalidad puede ser expandida mediante el uso de *plugins* para soportar lenguajes de programación tan dispares como C/C++,

Pascal, PHP o HTML, o para dotarle de compatibilidad con SCM's (Gestión de Configuración de Software) como Subversion o Mercurial.

Se trata de un *software* muy maduro, iniciado en 2001. Entre sus características más interesantes figuran el resaltado de sintaxis, la refactorización, el autocompletado de código, la integración de Javadoc en éste y el depurador integrado, así como la integración de Ant para la construcción de proyectos, o su subsistema de gestión de tareas, Mylyn. Eclipse es multiplataforma, existen versiones para Windows, Mac OS X o Linux. Esta es una de las ventajas de los desarrollos basados en Java: al ejecutarse sobre una máquina virtual, el *software* puede ser adaptado de forma sencilla a cualquier plataforma que esté soportada por la JVM. (Bermejo, 2010)

Aptana

Aptana es un IDE para el desarrollo web de código abierto. Incorpora características completas, sincronización, administración de proyectos, funciones mediante *plugins*, posibilidad de realizar subidas y bajadas de ficheros a un servidor vía File Transfer Protocol (FTP, por sus siglas en inglés) y Secure File Transfer Protocol (SFTP, por sus siglas en inglés), soporte para los sistemas operativos Microsoft Windows, Mac OS X y Linux en sus versiones más recientes. El programa brinda diversas utilidades que favorecen el desarrollo de distintas aplicaciones web, en especial archivos HyperText Markup Language (HTML, por sus siglas en inglés). Vigila y compila los cambios que se le realicen al proyecto en tiempo real. Permite comprobar la compatibilidad de las funciones con los diferentes navegadores, multiplataforma. (Jiménez, 2012)

NetBeans IDE

NetBeans IDE es un entorno de desarrollo, una herramienta para que los programadores puedan escribir, compilar, depurar y ejecutar programas. Está escrito en Java, pero puede servir para otros lenguajes de programación. Existe además un número importante de módulos para extender el NetBeans IDE. Es un producto libre y gratuito sin restricciones de uso. El código fuente está disponible para su reutilización de acuerdo con la CDDL versión 1.0 y la GNU GPL versión 2. (Oracle Corporation©, 2014)

NetBeans IDE permite el desarrollo rápido y fácil de aplicaciones HTML5 con HTML, JavaScript y CSS, así como aplicaciones Java de escritorio, móviles y aplicaciones web. El IDE también proporciona un gran conjunto de herramientas para desarrolladores PHP y C/C++.

Se seleccionó como IDE para el desarrollo del producto Netbeans IDE en su versión 7.4, la cual fue lanzada el 15 de octubre de 2013 y es la más actualizada en los servidores de la UCI. La misma brinda un amplio soporte al desarrollo de aplicaciones web, proveyendo facilidades tanto para el completamiento de código, como para las tareas desarrolladas mediante líneas de comando. La selección fue respaldada por la familiarización y experiencia de trabajo con esta herramienta.

1.9.5 Herramienta CASE

El desarrollo de sistemas informáticos, actualmente, se encuentra centrado en la demanda de una mayor productividad y calidad. En aras de estas exigencias se ha hecho indispensable la utilización de herramientas especializadas que soporten y se adapten a las metodologías de desarrollo más utilizadas. Estas herramientas que se dirigen a mejorar la productividad y calidad son las de Ingeniería de Software Asistida por Computadora (CASE). Las herramientas CASE proporcionan una disciplina automatizada para el diseño, mantenimiento y gestión del *software*, integradas por metodologías que guían el proceso de creación de *software* definiendo un disciplinado proceso para su desarrollo. (Benigni, 2013)

Rational Rose Enterprise

Rational Rose Enterprise proporciona un conjunto de prestaciones controladas por modelo para desarrollar muchas aplicaciones de *software*, incluidas aplicaciones Ada, ANSI C++, C++, CORBA, Java, Java EE, Visual C++ y Visual Basic. El *software* permite acelerar el desarrollo de estas aplicaciones con código generado a partir de modelos visuales mediante el lenguaje UML (Unified Modeling Language). Rational Rose Enterprise ofrece una herramienta y un lenguaje de modelado común para simplificar el entorno de trabajo y permitir una creación más rápida de *software* de calidad. (IBM, 2014)

- Modelado de las aplicaciones más comunes.
 - Admite patrones de análisis, ANSI C++, Rose J y Visual C++, Enterprise JavaBean 2.0 e ingeniería directa e inversa para algunas de las construcciones más comunes de Java 1.5.
 - Es capaz de analizar la calidad del código y de generar código gracias a las prestaciones de sincronización configurable entre el modelo y el código.
 - Permite la gestión granular y el uso de modelos con una característica de componentes de modelo que se puede controlar por separado.

- Proporciona el modelado UML para diseños de bases de datos.
- Permite integrar requisitos de datos y aplicación a través de diseños lógicos y físicos.
- Desarrollo de aplicaciones para la web.
 - Incluye un complemento de modelado web, que proporciona visualización, modelado y herramientas para desarrollar aplicaciones web.
 - Crea definiciones de tipos de documentos (DTD) XML para utilizarlas en las aplicaciones.
- Integración del diseño de aplicaciones con el desarrollo.
 - Permite utilizar un lenguaje para el ciclo de vida de análisis y desarrollo.
 - Se integra con otras herramientas de desarrollo de ciclo de vida de IBM Rational y con cualquier sistema de control de versiones compatible con SCC, incluido IBM Rational Clear Case.
 - Proporciona modelos e informes de publicación web para mejorar la comunicación en el equipo ampliado.

Visual Paradigm for UML

Es una herramienta para la Ingeniería de Software Asistida por Computadora que soporta el modelado mediante UML y proporciona asistencia a los analistas, ingenieros de *software* y desarrolladores, durante todas las etapas del ciclo de vida de desarrollo de un *software*.

Entre las principales facilidades que ofrece el uso de esta herramienta se encuentran: (Visual Paradigm©, 2014)

- Permite generar informes en varios formatos, entre los que se encuentran PDF, HTML y MS WORD. Ello permite estandarizar la documentación.
- Permite la generación de código a partir de diagramas para los lenguajes más populares, entre los que se encuentran Java, C# y PHP.
- Permite realizar ingeniería inversa a partir de bases de datos, soportando una amplia gama de bases de datos, entre las que se encuentran Oracle, MySQL y PostgreSQL.
- Se integra con los Entornos de Desarrollo Integrados NetBeans IDE, Eclipse, Visual Studio e IntelliJIDEA.

Se seleccionó como herramienta CASE a Visual Paradigm en su versión 8.0. La principal razón por la que fue seleccionada esta herramienta fue por la necesidad del uso de tecnologías libres

durante el desarrollo de la aplicación. Además es importante destacar que se integra con el entorno de desarrollo seleccionado por el autor y es la más conocida por el mismo.

Conclusiones del capítulo

En el capítulo se abordaron conceptos referentes a la tecnología multimedia, con algunas de las características más importantes que posee y las ventajas que aporta a la esfera educacional. Se hizo una breve reseña en general del proceso de enseñanza-aprendizaje de la IO en nuestra Universidad. Se realizó un análisis de soluciones existentes. Además se escogieron las herramientas, metodologías de desarrollo de *software*, lenguajes de modelado y lenguajes de programación a utilizar en la aplicación.

Capítulo 2

Descripción de la solución propuesta

Capítulo 2 Descripción de la solución propuesta

Introducción

En este capítulo estarán recogidos los aspectos fundamentales que describen la solución propuesta y el diagrama conceptual del negocio. Se describen las historias de usuario y las iteraciones que se crearon para la implementación de cada una de ellas.

2.1 Descripción de la propuesta de solución

El producto multimedia va dirigido a las personas interesadas en obtener conocimiento referente a la Programación Lineal, específicamente a los estudiantes de la Universidad de las Ciencias Informáticas, con el objetivo de que contribuya como complemento educativo de la asignatura Investigación de Operaciones, la cual se imparte en el segundo semestre del tercer año de la carrera.

Las presentaciones multimedia tienen muchas ventajas porque son de mayor agrado para el cliente. Cuando el usuario accede a esta se muestra una presentación y luego aparece la pantalla principal. Ya dentro de la aplicación el usuario puede moverse por todos los módulos con los que cuenta.

En el módulo Temas se encuentra la tabla de contenidos de la multimedia; el tema Programación Lineal se divide en dos subtemas, uno es Modelación y el segundo Métodos de Solución. Por otra parte el módulo cuestionario cuenta con ejercicios de autoevaluación donde el estudiante podrá poner en práctica lo aprendido. Además de estos la multimedia también cuenta con el módulo Biblioteca en el cual se puede descargar documentación asociada al tema. Por último aparecen los módulos Galería y Glosario donde el usuario puede encontrar imágenes, sonidos y videos asociados con la Programación Lineal y palabras de difícil comprensión para el usuario, con sus respectivos significados. La solución propuesta podrá servir de apoyo al proceso de enseñanza-aprendizaje en la Universidad.

2.2 Usuarios del sistema

Los usuarios del sistema o usuarios clientes son personas que se conectan al sistema para hacer uso de los servicios que este les proporciona.

Personas relacionadas con el sistema	Justificación
Usuario	Representa a los estudiantes de tercer año que cursan la carrera de Ingeniería en

	Ciencias Informáticas a quien va dirigida la aplicación, profesores de Investigación de Operaciones y a cualquier persona que desee interactuar con el sistema.
--	---

Tabla 1 Usuario del sistema

2.3 Diagrama conceptual del negocio

Un diagrama conceptual del negocio es un artefacto construido bajo las reglas de UML durante la concepción de un proyecto informático. Este modelo puede ser utilizado para capturar y expresar el entendimiento ganado sobre el negocio que nos ocupa. Se considera necesario generar este artefacto para un mejor entendimiento de la solución a desarrollar, a partir de identificar los conceptos y objetos relacionados con la solución propuesta.

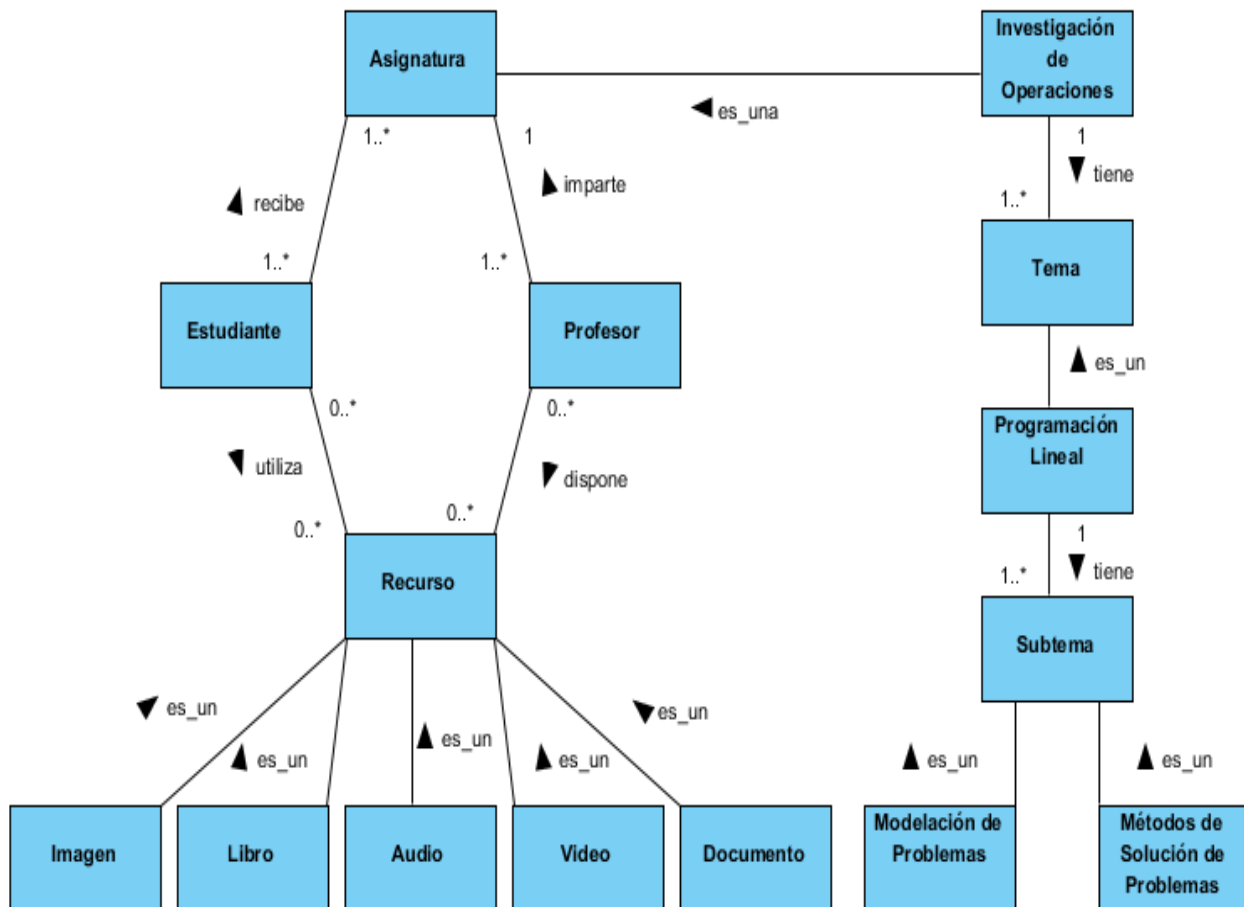


Figura 1 Diagrama Conceptual del Negocio

2.4 Exploración

Es la fase en la que se define el alcance general del proyecto. En esta fase, el cliente define lo que necesita mediante la redacción de sencillas historias de usuario. Los programadores estiman los tiempos de desarrollo en base a esta información. (Joscowicz, 2008)

Básicamente los clientes plantean a grandes rasgos las historias de usuario que les son de interés para la primera entrega del producto, a la vez que los programadores se familiarizan con las herramientas, tecnologías y prácticas que se utilizarán. Demora de pocas semanas a pocos meses, en dependencia de la magnitud y la familiaridad de los programadores con la tecnología. (Lacal, 2008)

En conclusión, la exploración es la primera fase de la metodología de desarrollo XP, en la cual se definen las historias de usuario, lo que posteriormente servirá para realizar una planificación del ciclo de vida del proyecto.

2.4.1 Historia de usuario

Las historias de usuarios son la forma en que se especifican en XP los requisitos del sistema. Estas se redactan desde la perspectiva del cliente aunque los desarrolladores pueden brindar también su ayuda en la identificación de las mismas. El contenido que ellas abarcan debe ser concreto y sencillo.

Las historias de usuarios no son más que tarjetas de papel en las cuales el cliente describe las características del sistema. En cualquier momento del desarrollo las mismas pueden ser modificadas o reemplazadas por otras más generales o específicas, o añadirse nuevas historias. Cada una es lo suficientemente entendible y está delimitada como para que los desarrolladores puedan implementarlas en unas semanas (Jeffries, 2001)

No lograr determinar todas las historias de usuario desde un inicio no es un problema, puesto que al inicio de cada iteración se registran los cambios en estas y a partir de ahí se planifica la próxima iteración. Las historias se descomponen en tareas de programación que son asignadas a los programadores para implementarlas durante una iteración.

2.4.2 Plantillas de historia de usuario

XP no propone un formato específico para las plantillas de historias de usuarios, deja los detalles a consideración del equipo de desarrollo. Se decide definir una plantilla que contenga el nombre de la HU, el número, el usuario que realiza la acción, la estimación del tiempo para

realizar la actividad, la prioridad, el riesgo, la iteración en que se encuentra, una breve descripción de la HU y un apartado para agregar alguna observación en caso de ser necesario.

Historia de usuario	
Número: Número de la HU	Nombre: El nombre de la HU, sirve para identificarla fácilmente entre los desarrolladores y los clientes.
Usuario: El usuario del sistema que utiliza o protagoniza la historia.	
Prioridad: Qué tan importante es para el cliente.	Riesgo: Qué tan difícil es para el desarrollador.
Estimación: Permite estimar duración de implementación.	Iteración: La iteración a la que corresponde.
Descripción: La descripción de la historia, detallando las operaciones del usuario y opcionalmente las respuestas del sistema.	
Observación: Algunas observaciones de interés, como glosario, información sobre usuarios.	

Tabla 2 Plantilla de HU

A continuación se detallan las historias de usuarios identificadas en este proceso, confeccionadas para el desarrollo de la solución propuesta.

Historia de usuario	
Número: 1	Nombre: Mostrar presentación de la multimedia.
Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 3 días	Iteración: 1
Descripción: El sistema muestra la presentación del <i>software</i> multimedia Programación Lineal. Al concluir la presentación, el sistema muestra la pantalla principal.	
Observación:	

Tabla 3 HU-Mostrar presentación de la multimedia

Historia de usuario	
Número: 2	Nombre: Permitir la navegabilidad por cualquier parte del

contenido del producto.	
Usuario: Cliente	
Prioridad: Media	Riesgo: Medio
Estimación: 3 días	Iteración: 2
Descripción: Da la opción al usuario de moverse dentro de la aplicación. Permite ir desde cualquier lugar de ella a cualquier otro que se desee.	
Observación:	

Tabla 4 HU-Permitir la navegabilidad por cualquier parte del producto

Historia de usuario	
Número: 3	Nombre: Mostrar el contenido de los temas que componen el producto.
Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 5 días	Iteración: 1
Descripción: Se brinda la posibilidad a los usuarios de la aplicación ver el contenido de la misma agrupados en los diferentes subtemas. El sistema habilita los botones que permiten la navegabilidad entre los subtemas del producto.	
Observación:	

Tabla 5 HU-Mostrar el contenido de los temas que componen el producto

Historia de usuario	
Número: 4	Nombre: Permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos.
Usuario: Cliente	
Prioridad: Media	Riesgo: Medio
Estimación: 5 días	Iteración: 2

Descripción:

Se presentan los ejercicios para comprobar los conocimientos adquiridos en el producto. El sistema le muestra al usuario un ejercicio y habilita el botón *Revisar*. Luego el sistema le muestra las respuestas correctas y la calificación obtenida; y habilita el botón *Continuar*.

Observación:

Tabla 6 HU-Permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos

Fueron descritas por parte del cliente un total de 14 HU, en este epígrafe se presenta una muestra de las mismas, las restantes se encuentran en el Anexo #6 de la presente investigación.

2.5 Planificación

La metodología XP plantea la planificación como un diálogo continuo entre las partes involucradas en el proyecto.

La planificación es una fase corta, en la que el cliente, los gerentes y el grupo de desarrolladores acuerdan el orden en que deberán implementarse las historias de usuario, y, asociadas a éstas, las entregas. Típicamente esta fase consiste en una o varias reuniones de planificación. El resultado de esta fase es un Plan de Entregas. Una vez acordado este cronograma, comienza una fase de iteraciones, en dónde en cada una de ellas se desarrolla, prueba e instala unas pocas historias de usuarios (Joscowicz, 2008)

En esta fase los desarrolladores y el cliente establecen los tiempos de implementación de cada historia de usuario, la prioridad con que serán desarrolladas y las historias que serán implementadas en cada versión del proyecto. Se realizará un plan de entregas que debe ser lo antes posibles, y con cada iteración, el cliente recibirá una nueva versión.

2.5.1 Estimación de esfuerzo por historias de usuario

Para un buen desarrollo de la solución propuesta, se realizó una estimación por cada una de las historias de usuarios que se identificaron.

Historias de usuario	Días estimados
HU1. Mostrar presentación de la multimedia.	3
HU2. Permitir la navegabilidad por cualquier parte del contenido del producto.	3
HU3. Mostrar el contenido de los temas que componen el producto.	5

HU4. Permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos.	5
HU5. Permitir el control del sonido de la aplicación.	3
HU6. Mostrar los créditos de la multimedia.	3
HU7. Seleccionar la sección Galería de Imágenes.	4
HU8. Seleccionar la sección Galería de Videos.	4
HU9. Seleccionar la sección Glosario de Términos.	3
HU10. Mostrar palabras calientes.	3
HU11. Seleccionar la sección Biblioteca.	4
HU12. Permitir descargar media.	3
HU13. Mostrar a pantalla completa.	3
HU14. Salir del sistema.	3

Tabla 7 Estimación de esfuerzo por HU

2.5.2 Iteraciones

Esta es la fase principal en el ciclo de desarrollo de XP. Las funcionalidades son desarrolladas en esta fase, generando al final de cada una, un entregable funcional que implementa las historias de usuario asignadas a la iteración. Cada historia de usuario se traduce en tareas específicas de programación. Asimismo, para cada historia de usuario se establecen las pruebas de aceptación. Estas pruebas se realizan al final del ciclo en el que se desarrollan, pero también al final de cada uno de los ciclos siguientes, para verificar que subsiguientes iteraciones no han afectado a las anteriores (Joscowicz, 2008)

Luego de ser identificadas y descritas todas las historias de usuario y estimado el esfuerzo propuesto para la realización de cada una de ellas, se procede a la planificación de la etapa de implementación del sistema. Este plan describe cuáles historias de usuario serán implementadas para cada iteración del sistema y las posibles fechas para estas liberaciones. En el sistema se deciden realizar 3 iteraciones:

- Iteración 1: Para esta primera iteración se implementarán cuatro historias de usuario de prioridad alta, para ello se escogieron las historias de usuario número 1, 3, 7, 8. Estas hacen referencia a mostrar la presentación de la multimedia, mostrar el contenido de los temas que componen el producto, a la galería de imágenes y a la galería de videos. Al final de esta iteración se agregaran tres días para realizar la refactorización del código de la

primera entrega del producto. Además se tendrá la primera versión de prueba, la cual será mostrada al cliente con el objetivo de obtener una retroalimentación.

- Iteración 2: En esta iteración se implementarán dos historias de usuario de prioridad alta y tres de media. Las historias de usuario escogida para esta iteración fueron la número 9, 11, 2, 4 y 13. Estas hacen referencia a la sección del glosario de términos, acceder a la sección Biblioteca, la navegabilidad por cualquier parte del producto, permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos y mostrar la multimedia a pantalla completa. A esta iteración se le agregaran 3 días para realizar la refactorización del código de la segunda entrega del producto. La versión de prueba referente a esta iteración será mostrada al cliente con el objetivo de realizar cambios en base a la opinión del mismo.
- Iteración 3: En esta última iteración se implementarán las historias de usuario número 14, 5, 6, 10, 12. Estas hacen referencia a salir de la aplicación, al sonido de la aplicación, a los créditos de la multimedia, mostrar las palabras calientes y permitir descargar las medias. Al final de esta iteración serán agregados 3 días para realizar la refactorización del código de la tercera entrega del producto. Como resultado de esta se tendrá la versión 1.0 del producto final. A partir de este momento el sistema será puesto a prueba por un período de tiempo para evaluar el desempeño del mismo.

2.5.3 Plan de duración de iteraciones

Como parte del ciclo de vida de un proyecto utilizando la Metodología XP se crea el plan de duración de cada una de las iteraciones. Este plan tiene como finalidad mostrar la duración de cada iteración, así como el orden en que serán implementadas las historias de usuario en cada una de las mismas. Está prohibido intentar adelantarse e implementar cualquier cosa que no esté planificada en la iteración en curso. Habrá suficiente tiempo para añadir la funcionalidad extra cuando sea realmente importante según el plan de entregas.

Iteraciones	Orden de las Historias de usuario a implementar	Duración total de las
Iteración 1	<p>HU1.Mostrar presentación de la multimedia.</p> <p>HU3. Mostrar el contenido de los temas que componen el producto.</p> <p>HU7. Seleccionar la sección Galería de Imágenes.</p> <p>HU8. Seleccionar la sección Galería de Videos.</p>	3 semanas

Iteración 2	<p>HU9. Seleccionar la sección Glosario de Términos.</p> <p>HU11. Seleccionar la sección Biblioteca.</p> <p>HU2. Permitir la navegabilidad por cualquier parte del contenido del producto.</p> <p>HU4. Permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos.</p> <p>HU13. Mostrar a pantalla completa.</p>	3 semanas
Iteración 3	<p>HU14. Salir del sistema.</p> <p>HU5. Permitir el control del sonido de la aplicación.</p> <p>HU6. Mostrar los créditos de la multimedia.</p> <p>HU10. Mostrar palabras calientes.</p> <p>HU12. Permitir descargar media.</p>	3 semanas

Tabla 8 Plan de duración de iteraciones

2.5.4 Plan de entregas

El Plan de entregas o cronograma de entregas, establece cuáles serán las historias de usuario agrupadas para conformar una entrega, y el orden de las mismas. El cliente decide qué historias de usuario colocar en la entrega y cuáles se implementarán después, mientras que los programadores se encargan de las estimaciones correspondientes a cada entrega.

Entregable	Final 1ra iteración(4 ^{ta} semana de marzo)	Final 2da iteración(3 ^{ra} semana de marzo)	Final 3ra iteración(2 ^{da} semana de mayo)
Multimedia para los Problemas de Programación	Versión 0.1	Versión 0.2	Versión 1.0

Tabla 9 Plan de entregas

Conclusiones del capítulo

Durante el desarrollo de este capítulo se describió la solución propuesta. Además se hizo referencia a los artefactos generados por las fases de Exploración y Planificación de la

metodología XP, asegurando una implementación por etapas correctamente descrita y detallada. En estas fases aparecen factores fundamentales para el desarrollo del sistema, como el diálogo entre el cliente y los desarrolladores, además se documentaron las historias de usuario necesarias, se priorizaron y se colocaron en las iteraciones donde debían ser implementadas, al igual que se generó el Plan de Entregas con las fechas en las que se publicaran cada versión del sistema.

Capítulo 3

Construcción de la solución propuesta

Capítulo 3 Construcción de la solución propuesta

Introducción

La Metodología XP plantea que la implementación de un *software* debe realizarse de forma iterativa, obteniendo al culminar cada iteración un producto funcional que debe ser probado y mostrado al cliente para incrementar la visión de los desarrolladores. En este capítulo se muestran los Diagramas de Presentación correspondientes a la vista de presentación espacial que propone el lenguaje de modelado OMMMA-L, que permite observar cuáles son las interfaces que interactúan con el usuario. Además se presentan las tarjetas CRC (Clase-Responsabilidades-Colaboradores) y las tareas de ingeniería necesarias para llevar a cabo el proceso de desarrollo. En la fase de pruebas se define la estrategia de pruebas a seguir, y se generan los casos de pruebas para realizar las pruebas de aceptación.

3.1 Diagramas de Presentación

El Diagrama de Presentación es un nuevo artefacto del lenguaje UML incorporado a partir de la extensión planteada por OMMMA-L. Éste utiliza los Diagramas de Presentación con el propósito de declarar las interfaces de usuario con un conjunto de estructuras delimitadas en tamaño y área, dividiéndose en objetos de visualización.

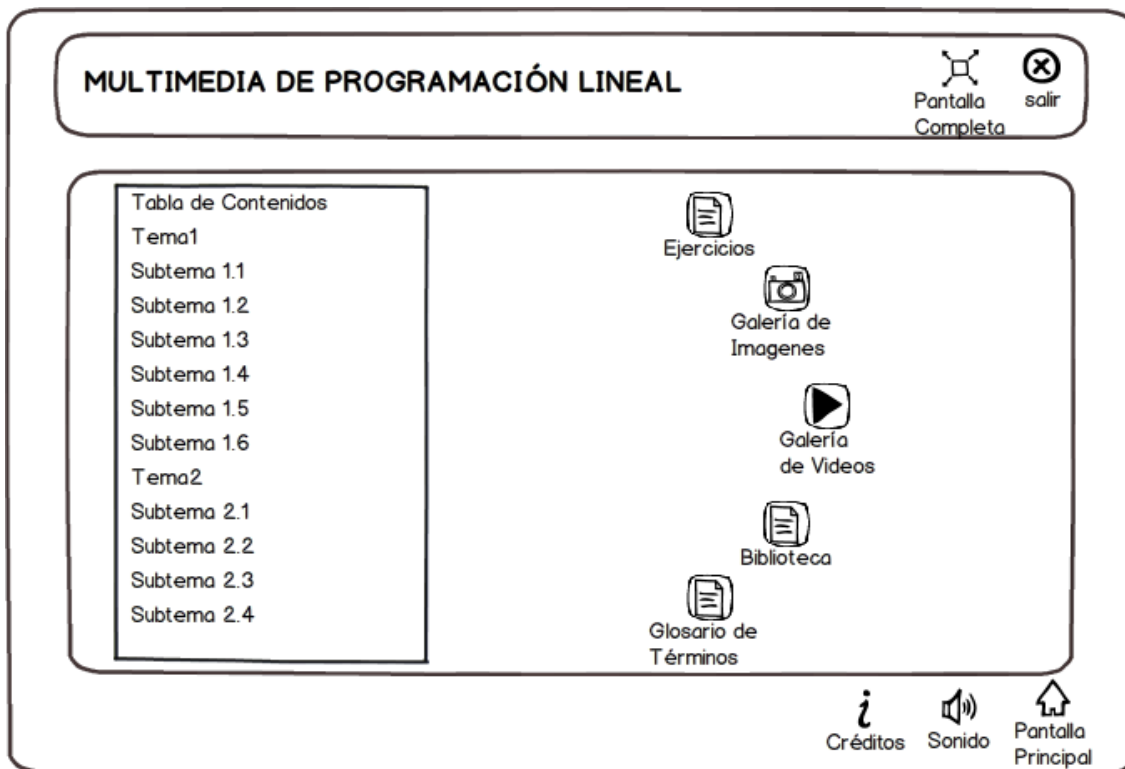


Figura 2 Diagrama de Presentación General

Fueron elaborados un total de 10 diagramas de presentación, correspondientes a las vistas de los módulos de la multimedia. En este epígrafe solo se muestra el Diagrama de Presentación General (Figura 7) de la propuesta de solución, los restantes se encuentran en el Anexo #5 de la presente investigación.

3.2 Tarjetas Clase-Responsabilidades-Colaboradores (CRC)

En la fase de producción, XP propone las tarjetas Clase Responsabilidad Colaborador (CRC) como técnica de diseño, la característica más prominente es su simpleza y ductilidad. Una tarjeta CRC establece 3 dimensiones las cuales identifican el rol de un objeto en análisis y/o diseño: nombre de la clase, responsabilidades y colaboraciones. (Casanova, 2013)

La principal funcionalidad que tienen éstas, es ayudar a dejar el pensamiento procedimental para incorporarse al enfoque orientado a objetos. Cada tarjeta representa una clase con su nombre en la parte superior, en la sección inferior izquierda están descritas las responsabilidades y a la derecha las clases que le sirven de soporte.

Nombre de la clase	
Responsabilidades	Colaboraciones

Tabla 10 Plantilla Tarjetas CRC

Una clase puede ser cualquier persona, cosa, evento, concepto, pantalla o reporte. Las responsabilidades de una clase son las cosas que conoce y que realiza, sus atributos y métodos. Los colaboradores de una clase son las demás clases con las que trabaja en conjunto para llevar a cabo sus responsabilidades. En la multimedia cada pantalla se comporta como un objeto independiente, de esta forma el mismo está formado por las siguientes clases:

- Clase Principal.
- Clase Menú Izquierdo.
- Clase Menú Derecho.
- Clase Galería.
- Clase Ejercicios.
- Clase Contenido.
- Clase Biblioteca.
- Clase Glosario.
- Clase Descargar.

Nombre de la clase: Principal

Responsabilidades	Colaboraciones
Clase principal del producto, encargada de la navegación por los diferentes módulos y controlar el dinamismo de la multimedia.	Todas las clases.

Tabla 11 Tarjeta CRC-Principal

Nombre de la clase: Menú Izquierdo	
Responsabilidades	Colaboraciones
Permite que el usuario pueda acceder al contenido que se ofrece en el producto multimedia.	Principal, Contenido.

Tabla 12 Tarjeta CRC-Menú Izquierdo

Nombre de la clase: Menú Derecho	
Responsabilidades	Colaboraciones
Permite que el usuario pueda acceder a las diferentes secciones ofrecidas en el producto.	Principal, Galería, Ejercicios, Biblioteca, Glosario.

Tabla 13 Tarjeta CRC-Menú Derecho

Nombre de la clase: Galería	
Responsabilidades	Colaboraciones
Clase encargada de mostrar las imágenes que se encuentran en la galería de imágenes, además de las funcionalidades relacionadas con la galería de videos.	Principal, Menú Derecho, Descargar.

Tabla 14 Tarjeta CRC-Galería

Nombre de la clase: Ejercicios	
Responsabilidades	Colaboraciones
Clase encargada de mostrar el contenido de los ejercicios.	Principal, Menú Derecho.

Tabla 15 Tarjeta CRC-Ejercicios

Nombre de la clase: Contenido	
Responsabilidades	Colaboraciones
Clase encargada de mostrar el contenido de los temas y de controlar el desplazamiento del texto.	Principal, Menú Izquierdo.

Tabla 16 Tarjeta CRC-Contenido

Nombre de la clase: Biblioteca	
---------------------------------------	--

Responsabilidades	Colaboraciones
Clase encargada de mostrar los documentos que se encuentran en la sección Biblioteca.	Principal, Menú Derecho, Descargar.

Tabla 17 Tarjeta CRC-Biblioteca

Nombre de la clase: Glosario	
Responsabilidades	Colaboraciones
Clase encargada de mostrar los significados de las palabras de difícil comprensión.	Principal, Menú Derecho.

Tabla 18 Tarjeta CRC-Glosario

Nombre de la clase: Descargar	
Responsabilidades	Colaboraciones
Clase encargada de descargar la media que el usuario seleccione.	Principal, Menú Derecho, Galería, Biblioteca.

Tabla 19 Tarjeta CRC-Descargar

3.3 Tareas de ingeniería

XP propone dividir cada historia de usuario en tareas de ingeniería con el objetivo de facilitar la implementación a los programadores. Cada tarea debe tener una duración de entre 1 y 3 días aproximadamente. Estas son para el uso estricto de los programadores y proporcionan una guía para un mejor desarrollo y cumplimiento de cada una de las historias de usuario. Las tareas seleccionadas fueron:

Historias de usuario	Tareas
HU1. Mostrar presentación de la multimedia.	<ul style="list-style-type: none"> Mostrar presentación de la multimedia. Mostrar pantalla principal.
HU2. Permitir la navegabilidad por cualquier parte del contenido del producto.	<ul style="list-style-type: none"> Permitir navegar globalmente dentro del producto.
HU3. Mostrar el contenido de los temas que componen el producto.	<ul style="list-style-type: none"> Crear textos de contenidos. Mostrar el contenido de temas ordenados por subtemas. Mostrar la información del contenido del

	<p>subtema seleccionado por el usuario.</p> <ul style="list-style-type: none"> • Permitir la navegación por cada uno de los subtemas que componen los temas mediante los botones correspondientes.
<p>HU4. Permitir mediante actividades de autoevaluación comprobar los conocimientos adquiridos.</p>	<ul style="list-style-type: none"> • Crear por cada tema una evaluación. • Permitir cargar ejercicios. • Mostrar y desarrollar el sistema de ejercicios. • Permitir al usuario ver la respuesta correcta, así como la evaluación obtenida por cada ejercicio.
<p>HU5. Permitir el control del sonido de la aplicación.</p>	<ul style="list-style-type: none"> • Permitir escuchar fondo musical.
<p>HU6. Mostrar los créditos de la multimedia.</p>	<ul style="list-style-type: none"> • Crear créditos.
<p>HU7. Seleccionar la sección Galería de Imágenes.</p>	<ul style="list-style-type: none"> • Crear galería de imágenes. • Mostrar imágenes.
<p>HU8. Seleccionar la sección Galería de Videos.</p>	<ul style="list-style-type: none"> • Mostrar videos. • Permitir reproducir videos.
<p>HU9. Seleccionar la sección Glosario de Términos.</p>	<ul style="list-style-type: none"> • Mostrar glosario de términos.
<p>HU10. Mostrar palabras calientes.</p>	<ul style="list-style-type: none"> • Crear vínculos en las palabras calientes.
<p>HU11. Seleccionar la sección Biblioteca.</p>	<ul style="list-style-type: none"> • Listar documentos. • Permitir leer documentos.
<p>HU12. Permitir descargar media.</p>	<ul style="list-style-type: none"> • Permitir descargar video. • Permitir descargar documento. • Permitir descargar imágenes.

HU13. Mostrar a pantalla completa.	<ul style="list-style-type: none"> • Permitir mostrar a pantalla completa.
HU14. Salir del sistema.	<ul style="list-style-type: none"> • Permitir salir desde cualquier parte del producto. • Crear ventana de comprobación de salida.

Tabla 20 HU divididas por tareas

Tarea	
Número de tarea: 1	Número de la Historia de Usuario: 1
Nombre de la tarea: Mostrar presentación de la multimedia.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 24 de febrero 2014	Fecha fin: 26 de febrero 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se brinda la posibilidad al usuario de ver la presentación de la multimedia.	

Tabla 21 Tarea #1

Tarea	
Número de tarea: 2	Número de la Historia de Usuario: 1
Nombre de la tarea: Mostrar pantalla principal.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 27 de febrero 2014	Fecha fin: 27 de febrero 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Al final de la presentación de la multimedia se debe mostrar la pantalla principal.	

Tabla 22 Tarea #2

Tarea	
Número de tarea: 3	Número de la Historia de Usuario: 2
Nombre de la tarea: Permitir navegar globalmente dentro del producto.	
Tipo de tarea: Desarrollo	Estimación: 3 días
Fecha inicio: 2 de abril 2014	Fecha fin: 4 de abril 2014
Programador responsable: Omar D. Castillo González	

Descripción:

Se ofrece la posibilidad al cliente de poder ir a cualquier parte de la multimedia en todo momento.

Tabla 23 Tarea #3

Tarea	
Número de tarea: 4	Número de la Historia de Usuario: 3
Nombre de la tarea: Crear textos de contenidos.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 28 de febrero 2014	Fecha fin: 3 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Se confeccionan los textos de contenidos que serán mostrados en las diferentes pantallas al usuario.	

Tabla 24 Tarea #4

Tarea	
Número de tarea: 5	Número de la Historia de Usuario: 3
Nombre de la tarea: Mostrar el contenido de temas ordenados por subtemas.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 4 de marzo 2014	Fecha fin: 4 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Se confecciona un menú con los diferentes temas y subtemas con que contara el producto, para una mejor comprensión y manejo de la información.	

Tabla 25 Tarea #5

Tarea	
Número de tarea: 6	Número de la Historia de Usuario: 3
Nombre de la tarea: Permitir la navegación por cada uno de los subtemas que componen los temas mediante los botones correspondientes.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 5 de marzo 2014	Fecha fin: 5 de marzo 2014

Programador responsable: Omar D. Castillo González
Descripción: Se habilitan los botones que permiten la navegabilidad entre los subtemas del producto.

Tabla 26 Tarea #6

Tarea	
Número de tarea: 7	Número de la Historia de Usuario: 3
Nombre de la tarea: Mostrar la información del contenido del subtema seleccionado por el usuario.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 6 de marzo 2014	Fecha fin: 6 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Permite mostrar el contenido del subtema seleccionado.	

Tabla 27 Tarea #7

Tarea	
Número de tarea: 8	Número de la Historia de Usuario: 4
Nombre de la tarea: Crear por cada tema una evaluación.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 7 de abril 2014	Fecha fin: 7 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción: Permite crear evaluaciones para cada uno de los temas del producto.	

Tabla 28 Tarea #8

Tarea	
Número de tarea: 9	Número de la Historia de Usuario: 4
Nombre de la tarea: Permitir cargar ejercicios.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 8 de abril 2014	Fecha fin: 8 de abril 2014
Programador responsable: Omar D. Castillo González	

Descripción:

Se confeccionan los textos donde serán cargados los ejercicios.

Tabla 29 Tarea #9

Tarea	
Número de tarea: 10	Número de la Historia de Usuario: 4
Nombre de la tarea: Mostrar y desarrollar el sistema de ejercicios.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 9 de abril 2014	Fecha fin: 9 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Se le brinda la posibilidad al usuario de responder el ejercicio presentado y se habilita el botón <i>Revisar</i> .	

Tabla 30 Tarea #10

Tarea	
Número de tarea: 11	Número de la Historia de Usuario: 4
Nombre de la tarea: Permitir al usuario ver la respuesta correcta, así como la evaluación obtenida por cada ejercicio.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 10 de abril 2014	Fecha fin: 11 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Se le muestra al usuario las respuestas correctas y la calificación obtenida, y se habilita el botón <i>Continuar</i> .	

Tabla 31 Tarea #11

Las 14 HU redactadas fueron desglosadas en un total de 27 tareas de ingeniería, en este epígrafe solo se muestran las relacionadas con las HU que se describen en el Capítulo 2, las restantes se encuentran en el Anexo #7 de la presente investigación.

3.4 Estándares de codificación

XP enfatiza la comunicación de los programadores a través del código, con lo cual es indispensable que se sigan ciertos estándares de programación. Los estándares de codificación mantienen el código legible para los miembros del equipo facilitando los cambios, de modo que éste no sea conocido por una sola persona del equipo de desarrollo, sino que otros miembros

puedan realizar cambios en el él. Por tales razones durante el proceso de implementación de un *software* es considerado una buena práctica realizar la codificación del mismo siguiendo estándares que guíen este proceso.

Los estándares de codificación más relevantes utilizados en la implementación de la propuesta de solución se describen a continuación:

- Los comentarios entre líneas se escriben comenzando con los caracteres “/*” y terminando con “*/”.
- Las líneas de código que debido a su tamaño ocupan más del espacio visible que ofrece el NetBeans IDE para escribir código, deben ser divididas en varias, preferiblemente después de una coma o un operador.
- Las variables deben ser explícitas, aunque se pueden usar abreviaturas siempre y cuando no violen este principio.
- Para el nombre de botones se utiliza el sufijo `_btn`.
- El código se encuentra tabulado a través del formato que aplica la combinación de teclas del NetBeans IDE, `Alt+Shift+F`.

Como el desarrollo de una aplicación no se termina con la entrega del producto al usuario final, sino que siempre se debe pensar en posibles mejoras, es necesario el uso de estándares, el cual ayudará a los desarrolladores tener una mejor comprensión del código, durante la implementación y en futuros mantenimientos.

3.5 Pruebas

Uno de los pilares de la Programación Extrema es el proceso de pruebas. XP anima a probar constantemente tanto como sea posible. Esto permite aumentar la calidad de los sistemas reduciendo el número de errores no detectados y disminuyendo el tiempo transcurrido entre la aparición de un error y su detección. También permite aumentar la seguridad de evitar efectos colaterales no deseados a la hora de realizar modificaciones y refactorizaciones. XP divide las pruebas del sistema en dos grupos: pruebas unitarias, encargadas de verificar el código y diseñada por los programadores, y pruebas de aceptación o pruebas funcionales destinadas a evaluar si al final de una iteración se consiguió la funcionalidad requerida diseñadas por el cliente final. (Gutiérrez, 2014)

Un equipo de desarrollo que siga el proceso de XP debe primero probar su código de forma unitaria, integrar este a la aplicación y después realizar las pruebas de aceptación.

3.5.1 Desarrollo dirigido por pruebas (TDD)

El desarrollo guiado por pruebas formalizado en XP se conoce como Test Driven Development (TDD por sus siglas en inglés). TDD es un enfoque ágil, donde para cada funcionalidad que se desea implementar, primero se escriben las pruebas y luego el código necesario para que la prueba sea exitosa. Una vez que el código cumple el test exitosamente, se amplía y continúa. De este modo, se realiza una integración continua, evitando un proceso más complejo al finalizar el proyecto.

TDD es una técnica de programación que consiste en guiar el diseño de una aplicación, por medio de pruebas unitarias. Esta cambia el orden tradicionalmente establecido, ya que primero se definen las pruebas y a partir de estas se va desarrollando la funcionalidad. El ciclo se repite de acuerdo a lo que se espera que haga el *software*, por medio de integraciones y refactorizaciones continuas del desarrollo en los casos en que las pruebas no cumplan con el requerimiento. Con esta técnica las pruebas constituyen la documentación del *software* que se está desarrollando. (Vivas, 2013)

3.5.2 Pruebas unitarias

En los últimos años, los test unitarios han ido tomando cada vez más fuerza en el proceso de desarrollo de *software* y se integraron de una forma altamente productiva. El desarrollo y ejecución de pruebas es una actividad fundamental en los proyectos de desarrollo de *software*, los cuales permiten mantener código de calidad durante el ciclo de vida del proyecto. Fundamentalmente, las pruebas unitarias representan una alternativa para encontrar y corregir la mayoría de los errores de codificación. Estas pruebas, como primer paso en el proceso de desarrollo son la base de la filosofía TDD, ya que resulta la mejor manera de producir código rápidamente y de calidad, permitiendo conducir el diseño, mediante la codificación de las citadas pruebas, antes de codificar interfaces o implementaciones. (Vivas, 2013)

XP aconseja que la realización de las pruebas unitarias al sistema se realice de forma automatizada y en etapas tempranas del desarrollo, permitiendo disminuir la ocurrencia de defectos y aprovechar las ventajas de la retroalimentación que se produce en el proceso. Aunque es extenuante la tarea de crear casos de prueba manuales pueden utilizarse complementariamente a las anteriores. Sin embargo, es válido destacar que en el caso de la aplicación en cuestión no se realizaron pruebas unitarias.

3.5.3 Pruebas de aceptación

Se denominan pruebas funcionales, a las pruebas de *software* que tienen por objetivo probar que los sistemas desarrollados, cumplan con las funciones específicas para los cuales han sido creados. Esta etapa suele ser la última etapa de pruebas y al dar conformidad sobre esta el paso siguiente es el pase a producción. A este tipo de pruebas se les denomina también pruebas de comportamiento o pruebas de caja negra, ya que los analistas de pruebas, no enfocan su atención a como se generan las respuestas del sistema, básicamente el enfoque de este tipo de prueba se basa en el análisis de los datos de entrada y en los de salida. (Oré, 2009)

Los clientes junto a un miembro del equipo de desarrollo son los responsables de verificar que los resultados de estas pruebas sean los correctos para así tomar decisiones acerca de las mismas. Una HU no se puede considerar terminada hasta que no pase los test de aceptación. Dado que la responsabilidad es grupal, es recomendable publicar los resultados de las pruebas de aceptación, de manera que todo el equipo esté al tanto de esta información.

Las pruebas de aceptación son más importantes que las pruebas unitarias dado que significan la satisfacción del cliente con el producto desarrollado y el final de una iteración y el comienzo de la siguiente. (Gutiérrez, 2014)

Seguidamente se muestra la planilla de caso de prueba de aceptación definida por el equipo de desarrollo:

Caso de prueba de aceptación	
Código: Código que identifica a la prueba.	Historia de usuario: Número de la HU relacionada con la prueba que realiza
Nombre: Nombre de la prueba que se realiza.	
Descripción: Breve descripción con el objetivo que se realiza la prueba.	
Condiciones de ejecución: Condiciones que deben cumplirse para que pueda realizarse la prueba.	
Entrada/ Pasos de ejecución: Se describen los pasos de ejecución de la prueba en cuestión.	
Resultado esperado: Proporciona las expectativas ideales para las cuáles fue pensada la prueba.	

Evaluación de la prueba: Calificación que recibe la prueba de acuerdo con los resultados obtenidos.

Tabla 32 Plantilla de Casos de Prueba

A continuación se presentan los casos de pruebas de aceptación referentes a la solución propuesta:

Caso de prueba de aceptación	
Código: HU1_CP1	Historia de usuario: 1
Nombre: Mostrar presentación de la multimedia.	
Descripción: Prueba para la funcionalidad de que el producto multimedia pueda ser inicializado.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se procede a la presentación del producto multimedia.	
Resultado esperado: Se facilita que el usuario pueda ver la presentación de la multimedia.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 33 Caso de prueba de aceptación #1

Caso de prueba de aceptación	
Código: HU1_CP2	Historia de usuario: 1
Nombre: Mostrar pantalla principal.	
Descripción: Prueba para la funcionalidad de mostrar al usuario las opciones que brinda el producto multimedia.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se procede a la pantalla principal del producto multimedia.	
Resultado esperado: Se facilita que el usuario pueda ver la pantalla principal de la aplicación.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 34 Caso de prueba de aceptación #2

Caso de prueba de aceptación	
Código: HU2_CP3	Historia de usuario: 2
Nombre: Permitir navegar globalmente dentro del producto.	

Descripción: Prueba para la funcionalidad de permitir la navegabilidad en dentro del producto.
Condiciones de ejecución: El usuario debe acceder a la multimedia.
Entrada/ Pasos de ejecución: Se intenta navegar dentro del producto.
Resultado esperado: Se brinda la posibilidad de que el usuario pueda navegar dentro del producto de manera global.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 35 Caso de prueba de aceptación #3

Caso de prueba de aceptación	
Código: HU3_CP4	Historia de usuario: 3
Nombre: Mostrar el contenido de temas ordenados por subtemas.	
Descripción: Prueba para revisar la funcionalidad de acceder a un tema o subtema.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se solicita el tema o subtema deseado.	
Resultado esperado: Permite seleccionar el tema o subtema deseado.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 36 Caso de prueba de aceptación #4

Caso de prueba de aceptación	
Código: HU3_CP5	Historia de usuario: 3
Nombre: Mostrar la información del contenido del subtema seleccionado por el usuario.	
Descripción: Prueba para verificar la información de un tema o subtema.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se solicita el tema o subtema a consultar mostrándose dicho contenido.	
Resultado esperado: Permite ver toda la información del subtema seleccionado.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 37 Caso de prueba de aceptación #5

Caso de prueba de aceptación	
Código: HU4_CP6	Historia de usuario: 4
Nombre: Mostrar y desarrollar el sistema de ejercicios	

Descripción: Prueba para la funcionalidad de mostrar y resolver los ejercicios de la multimedia.
Condiciones de ejecución: El usuario debe acceder a la multimedia.
Entrada/ Pasos de ejecución: Se procede a resolver los ejercicios de la aplicación.
Resultado esperado: Se brinda la posibilidad al usuario de resolver los ejercicios con que cuenta el producto para comprobar los conocimientos adquiridos
Evaluación de la prueba: Prueba satisfactoria.

Tabla 38 Caso de prueba de aceptación #6

Caso de prueba de aceptación	
Código: HU4_CP7	Historia de usuario: 4
Nombre: Permitir al usuario ver la respuesta correcta, así como la evaluación obtenida por cada ejercicio.	
Descripción: Prueba para la funcionalidad de revisar las respuestas de los ejercicios de la multimedia.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se procede a resolver los ejercicios de la aplicación.	
Resultado esperado: Se brinda la posibilidad al usuario de ver las respuestas correctas por cada ejercicio así como la evaluación obtenida.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 39 Caso de prueba de aceptación #7

Fueron elaborados un total de 20 casos de prueba, para realizar las pruebas de aceptación. En este epígrafe solo se muestran los relacionados con las tareas de ingeniería de tipo de desarrollo elaboradas en el epígrafe 3.3, los restantes se encuentran en el Anexo #8 de la presente investigación.

3.6 Análisis de los resultados de las pruebas

Al terminar cada una de las iteraciones planificadas para el desarrollo de la multimedia, fueron realizadas las pruebas de aceptación para realizar las entregas pactadas con el cliente, en la tabla se muestran los resultados de las pruebas en cada una de las iteraciones.

	Iteración 1	Iteración 2	Iteración 3
Cantidad de No Conformidades.	8	5	6

Tabla 40 Cantidad de No Conformidades detectadas por iteración

Todas las no conformidades fueron resueltas, lo cual valida en cierto grado la calidad de la solución propuesta, logrando una mayor satisfacción por parte del cliente.

Conclusiones del capítulo

En este capítulo se modelaron los Diagramas de Presentación del producto multimedia, es decir, los diferentes escenarios con los que contará la aplicación. Además se generaron artefactos de gran importancia para llevar a cabo la implementación de la solución, tal es el caso de las tarjetas CRC y las tareas de ingenierías necesarias para desarrollar cada historia de usuario. Se plantea la estrategia de pruebas seguida por el equipo de desarrollo, y los resultados de las pruebas de aceptación las cuales permiten al cliente comprobar el correcto funcionamiento de la multimedia.

Conclusiones y Recomendaciones

Conclusiones generales

La culminación de la presente investigación permitió arribar a las siguientes conclusiones:

- Al investigar el objeto de estudio, los métodos científicos empleados posibilitaron identificar los conceptos y teorías que sustenta la investigación para el desarrollo de la solución.
- La metodología seleccionada permitió al equipo de desarrollo cumplir con el objetivo general de la investigación y satisfacer de esta forma las necesidades del cliente.
- Las herramientas y lenguajes de programación seleccionados permitieron el desarrollo de la solución propuesta.
- La validación demostró la eficacia del proceso de desarrollo llevado a cabo y además, comprobó el correcto funcionamiento de la solución.

Recomendaciones

Al término de esta investigación se recomienda:

- Poner la multimedia a disposición de todos los estudiantes que cursan el tercer año en la Universidad de las Ciencias Informáticas para que le sirva de apoyo en el estudio de la asignatura Investigación Operaciones.
- Recopilar nuevos videos e imágenes que no se encuentren disponibles en la multimedia referente al tema de Programación Lineal e incorporarlos luego a las galerías de la misma, así como otros materiales de interés.
- Enriquecer la multimedia profundizando en los temas abordados y los cuestionarios para una mejor preparación de los estudiantes.

Referencias bibliográficas

1. **PINTO, María.** ALFAMEDIA. [En línea]. 13 Abril 2011. [Accedido 3 Febrero 2014]. Disponible en: <http://www.mariapinto.es/alfamedia/cultura/entornos.htm>.
2. **RODRÍGUEZ, C. Orestes Coloma,** y otros. EL DESARROLLO DE SOFTWARE EDUCATIVO SIN COSTO DE PROGRAMACIÓN. ¿UTOPIA O REALIDAD?, 2011 *Curso*, vol. 71, p. 67.
3. **ROSAS DE MAIDANA, María Teresa.** El rol del profesional de la información y la información audiovisual y multimedia como servicio del centro de recursos para el aprendizaje (CRA) | Rosas de Maidana | Cuadernos de Documentación Multimedia. [En línea]. 2011. [Accedido 4 Febrero 2014]. Disponible en: <http://revistas.ucm.es/index.php/CDMU/article/view/38341/37097>.
4. **MOLINA, Juan Manuel Madrid.** La multimedia y sus aplicaciones: elaboración de un video promocional para la carrera de ingeniería de sistemas. Publicaciones Icesi, 2010, no 52.
5. **VERA, JULIO.** Primeros lenguajes y últimas tecnologías para la educación. Teoría de la Educación. Educación y Cultura en la Sociedad de la Información, 2013, vol. 14, no 3, p. 146-174.
6. **DE OLIVEIRA, Raquel Gomes.** LAS RUTAS HIPERTEXTUALES E HIPERMEDIATICAS DEL CIBERPERIODISMO. Razón y Palabra, 2012, no 79.
7. **LAMARCA, María Jesús.** Hipermedia/Multimedia. [En línea]. 8 Diciembre 2013. [Accedido 5 Febrero 2014]. Disponible en: <http://www.hipertexto.info/documentos/hipermedia.htm>.
8. **VIDAL LEDO, María; RODRÍGUEZ DÍAZ,** Alfredo. Multimedia Educativas. Educación Médica Superior, 2010, vol. 24, no 3, p. 430-441.
9. **MARQUÉS, Pere.** MUTIMEDIA EDUCATIVO: FUNCIONES, VENTAJAS E INCONVENIENTES. [En línea]. 3 Agosto 2010. [Accedido 6 Febrero 2014]. Disponible en: <http://peremarques.pangea.org/funcion.htm>.
10. **HENAO, Mónica and SOLÓRZANO, Bertha Alicia.** Fusión de dos tecnologías: ingeniería del conocimiento (IC) + multimedia (MM) | Henao Calad | Revista Universidad EAFIT. [En línea]. [Accedido 5 Febrero 2014]. Disponible en: <http://publicaciones.eafit.edu.co/index.php/revista-universidad-eafit/article/view/1380>.
11. **IBÁÑEZ, Jesús Salinas.** Hipertexto e hipermedia en la enseñanza universitaria. Pixel-Bit: Revista de medios y educación. 1994. No. 1, p. 2-.

12. **HILLIER, Frederick S.** and **LIEBERMAN, Gerald J.** Introducción a la investigación de operaciones. McGraw-Hill/Interamericana, 2006. ISBN 9789701056219.
13. **BIANCHINI, Adelaide.** Metodología para el desarrollo de aplicaciones educativas en ambientes multimedios. Trabajo de ascenso a la categoría de Profesor Asociado en la Escuela de Ingeniería de Sistemas, Cap. IV. Universidad Metropolitana de Caracas, 1992.
14. **ESPINOZA, Norelkys; MORALES, O.** El texto electrónico: ¿La desaparición de lo impreso o la aparición de una nueva fuente de lectura? Lectura y vida, 2002, vol. 23, no 4, p. 14-25.
15. **CASTELLANOS, Ulises.** Manual de fotoperiodismo: retos y soluciones. Universidad Iberoamericana, 2003.
16. **Glosario - Docentes y Directivos de Superior.** [En línea]. [Accedido 9 Febrero 2014]. Disponible en: <http://www.colombiaaprende.edu.co/html/directivos/1598/article-75224.html>.
17. **CASTRO, Martín Germán Zambrano.** Diseño e implementación de un software multimedia de apoyo a los procesos de enseñanza-aprendizaje para la resolución de problemas en Investigación de Operaciones solución gráfica de máximos. Revista Inventum [En línea]. 16 Marzo 2010. Vol. 0, no. 8. [Accedido 9 Mayo 2014]. Disponible en: <http://biblioteca.uniminuto.edu/ojs/index.php/Inventum/article/view/33>.
18. **PÉREZ VERGARA, Ileana, GARZA RÍOS, Rosario, ARGUELLES, Leydier, PÉREZ JIMÉNEZ, Surelys and GARCÍA PÉREZ, Linette.** DESARROLLO DE MULTIMEDIAS PARA EL APRENDIZAJE DE LA INVESTIGACIÓN DE OPERACIONES. [En línea]. 2008. [Accedido 24 Febrero 2014]. Disponible en: <http://revistas.mes.edu.cu/greenstone/collect/repo/import/repo/20100204/978959161165901059.pdf>.
19. **LÓPEZ REYES, Yoansy, GARCÍA LLANES, Hector, FERIA CHAPMAN, Yunior and EDGHILL MARTÍNEZ, Yanet.** HERRAMIENTA DE APOYO AL PROCESO DE ENSEÑANZA - APRENDIZAJE DE LA INVESTIGACIÓN DE OPERACIONES EN LA UCI | Congreso Internacional COMPUMAT 2013. [En línea]. 1 Junio 2013. [Accedido 9 Mayo 2014]. Disponible en: https://compumat.uci.cu/sites/default/files/public/p1301-ponencia-1602_0.pdf.
20. **PRESSMAN, Roger S.** Ingeniería del Software: un enfoque práctico. quinta. La Habana: Editorial Félix Varela, 2005.

21. **JACOBSON, Ivar** and RATIONAL SOFTWARE CORPORATION. El Proceso Unificado de Desarrollo de Software [En línea]. Madrid: Addison, 2000. ISBN 84-7829-036-2. Disponible en: <http://bibliodoc.uci.cu/pdf/8478290362.pdf>. 005.1-Jac-P.
22. **FIGUEROA, Roberth G.; SOLÍS, Camilo J.; CABRERA, Armando A.** Metodologías Tradicionales vs. Metodologías Ágiles. Universidad Técnica Particular de Loja, Escuela de Ciencias en Computación. (En línea), Disponible en: <http://adonisnet.files.wordpress.com/2008/06/articulo-metodologia-de-sw-formato.doc>.
23. **PÉREZ, Oiver Andrés.** Cuatro enfoques metodológicos para el desarrollo de Software RUP – MSF – XP - SCRUM. Revista Inventum [En línea]. 10 Junio 2011. Vol. 0, no. 10. [Accedido 11 Febrero 2014]. Disponible en: <http://biblioteca.uniminuto.edu/ojs/index.php/Inventum/article/view/9>.
24. **LETELIER, Patricio.** Metodologías ágiles para el desarrollo de software: eXtreme Programming (XP). 15 Abril 2006. [Accedido 11 Febrero 2014]. Disponible en: http://www.cyta.com.ar/ta0502/b_v5n2a1.htm.
25. **EGUÍLUZ PÉREZ, Javier.** Introducción a JavaScript. 2009.
26. **GAUCHAT, Juan Diego.** El gran libro de HTML5, CSS3 y Javascript [En línea]. S.I.: Marcombo, 2012. ISBN 8426717829. Disponible en: <http://books.google.com.cu/books?id=szDMIRzwzUC&lpg=PP1&dq=html5&pg=PA31#v=onepage&q=html5&f=false>.
27. **W3C.** W3C Oficina Española. [En línea]. [Accedido 1 Febrero 2014]. Disponible en: <http://www.w3c.es/Divulgacion/a-z/#h>.
28. **MANN, Mik.** Programación en JavaScript [En línea]. Mikel Angoar, [no date]. [Accedido 1 Febrero 2014]. Disponible en: http://www.google.com.cu/books?id=_FiRF6sW1cIC&lpg=PA1&ots=EXXhRwfsUW&dq=Caracter%C3%ADsticas%20de%20javascript&lr&hl=es&pg=PA1#v=onepage&q=Caracter%C3%ADsticas%20de%20javascript&f=false.
29. **VALDELLI, Ilario.** JAVASCRIPT: Aspectos y características generales. HTMLPOINT.com [En línea]. 2014. [Accedido 1 Febrero 2014]. Disponible en: http://www.htmlpoint.com/javascript/corso/js_02.htm.
30. **Alvarez, Miguel Angel** .Manual de CSS 3. 2011.
31. **LANCKER, Luc Van.** XHTML y CSS - Los nuevos estándares del código fuente [En línea]. Ediciones ENI, 2009. ISBN 9782746047426. Disponible en: http://www.google.com.cu/books?id=_REZ8UA0efQC&lpg=PA20&ots=ysM3ldAI_n&dq=XHTML%20y%20CSS%20

%20Los%20nuevos%20estandares%20del%20codigo%20fuente.%20&lr=lang_es%7Clang_en&hl=es&pg=PA20#v=onepage&q=XHTML%20y%20CSS%20%20Los%20nuevos%20estandares%20del%20codigo%20fuente.&f=false.

32. **LUCA, Damián De.** HTML5: entienda el cambio, aproveche su potencial [En línea]. S.I.: USERSHOP, 2011. ISBN 9789871773794. Disponible en: <http://books.google.com/cu/books?id=DGwjKIXWPm8C&lpg=PA29&dq=ventajas%20de%20css&hl=es&pg=PA30#v=onepage&q=ventajas%20de%20css&f=false>.
33. **Entornos de Desarrollo Integrado.** [En línea] [Citado el: 15 de 12 de 2013.] <http://petra.euitio.uniovi.es/~i1667065/HD/documentos/Entornos%20de%20Desarrollo%20Integrado.pdf>.
34. **ORACLE CORPORATION©.** Bienvenido a NetBeans y www.netbeans.org, Portal del IDE Java de Código Abierto. [En línea]. [Accedido 12 Febrero 2014]. Disponible en: https://netbeans.org/index_es.html.
35. **VISUAL PARADIGM©.** Visual modeling tool for building enterprise applications. In: [En línea]. [Accedido 17 febrero 2014]. Disponible en: <http://www.visual-paradigm.com/product/vpuml/features/>.
36. **LANCKER, Luc Van.** JQUERY. EL FRAMEWORK JAVASCRIPT DE LA WEB 2.0 [En línea]. Ediciones ENI, 2012. [Accedido 22 Marzo 2014]. ISBN 9782746072589. Disponible en: <http://books.google.com/cu/books?id=SsTUoWkg588C&lpg=PA18&dq=ventajas%20de%20jquery&pg=PA15#v=onepage&q=ventajas%20de%20jquery&f=false>.
37. **LERNER, Reuven M.** At the forge: Twitter bootstrap. Linux Journal, 2012 . vol. 2012, no 218, p. 6.
38. **JEFFRIES Ron, ANDERSON, Ann and HENDRICKSON, Chet.** Extreme Programming Installed [En línea]. Addison-Wesley Professional, 2001. ISBN 9780201708424. Disponible en: http://www.google.com/cu/books?id=5ZuPjdO8LLoC&lpg=PR13&ots=OHXBtdUX_H&dq=Extreme%20Programming%20Explained.%20Embrace%20Change&lr&pg=PR13#v=onepage&q=Extreme%20Programming%20Explained.%20Embrace%20Change&f=false.
39. **JOSKOWICZ, José.** Reglas y prácticas en Extreme Programming. Universidad de Vigo. España, 2008. <http://iie.fing.edu.uy/~josej/docs/XP%20-%20Jose%20Joskowicz.pdf>.
40. **LACAL, Daynel Marmol.** Introducción a XP. Serie Científica, 2008, vol. 1, no 8. <http://publicaciones.uci.cu/index.php/SC/article/view/78/79>.

41. **FRANGANILLO, Jorge. H.** 3. Html5: el nuevo estándar básico de la Web. 2010. www.academia.edu/download/30940232/html5.pdf.
42. **ALLAIRE, Jeremy.** “The future of Web content: html5,Flash and mobile apps”. TechCrunch, 5 febrero 2010. <http://techcrunch.com/2010/02/05/the-future-of-webcontent-html5-flash-mobile-apps>.
43. **RIQUELME, Linda.** Trabajo Práctico [En línea]. 2012. Universidad Católica Nuestra Señora de la Asunción. [Accedido 4 Abril 2014]. Disponible en: <http://www.jeuazarru.com/docs/HTML5.pdf>.
44. **CANÓS, Joseph; LETELIER, Patricio y PENADÉS, M^a Carmen.** Metodologías Ágiles en el desarrollo de Software. Universidad Politécnica de Valencia, Valencia, 2003. Disponible en: http://noqualityinside.com.ar/nqi/nqifiles/XP_Agil.pdf.
45. **IBM.** IBM - Rational Rose Enterprise. [En línea]. 3 Mayo 2014. [Accedido 3 Mayo 2014]. Disponible en: <http://www-03.ibm.com/software/products/es/enterprise>.
46. **BENIGNI, Gladys and CELADA, Jairo.** MOOMH CASE. HERRAMIENTA AUTOMATIZADA PARA LA AYUDA AL DESARROLLO DE SISTEMAS HIPERMEDIALES. Revista de la Facultad de Ingeniería Universidad Central de Venezuela. 27 Julio 2013. Vol. 23, no. 4, p. 99–110.
47. **JIMÉNEZ, Lianet Ballester.** Arquitectura de software para el producto “Mis Mejores Cuentos”. Serie Científica, 2012, vol. 5, no 5. [Accedido 12 Febrero 2014]. Disponible en: <http://publicaciones.uci.cu/index.php/SC/article/viewFile/875/539>.
48. **BERMEJO CALVO, Álvaro.** Extracción de características para recuperación de imágenes basada en contenido. [En línea]. 2010. [Accedido 3 Mayo 2014]. Disponible en: <http://eciencia.urjc.es/handle/10115/5533>.
49. **HERNÁNDEZ, Gómez and MARÍA, José.** Visualización de esquemas UML con HTML5. [En línea]. 9 Julio 2012. [Accedido 4 Mayo 2014]. Disponible en: <http://upcommons.upc.edu/pfc/bitstream/2099.1/15599/1/82470.pdf>.
50. **BERZAL, Fernando.**UML. El lenguaje de modelado unificado. [En línea]. [Accedido 14 Febrero 2014]. Disponible en: <http://pwp.etb.net.co/witorres/poo/3E-UML.pdf>.
51. **MARTINEZ IGARZA, Mailen and BARROSO HIDALGO, Noidis.** Producto multimedia vida y obra de Julio Antonio Mella. [En línea]. 2010. [Accedido 4 Mayo 2014]. Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_02832_10/1/TD_02832_10.pdf.

52. **LORENTE RODRÍGUEZ, Abel Ernesto, DÍAZ DOMÍNGUEZ, Alexey and MARTÍNEZ PÉREZ, Yancy.** Plantilla para el Montaje Dinámico de los Productos de la Colección Multisaber. [En línea]. 2006. [Accedido 4 Mayo 2014]. Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_0140_06/1/TD_0140_06.pdf.
53. **CASANOVA MUTIS, Alejandro and RODRÍGUEZ , Alain Osorio.** Herramienta generadora de ficheros de mapeo para la persistencia de esquemas de objetos relacionales. [En línea]. 13 Junio 2013. Vol. 6, no. 6. Disponible en: <http://publicaciones.uci.cu/index.php/SC/article/view/1156/690>.
54. **DELGADO CARMONA, Luz Elena and ECHEVERRY TOBÓN, Luís Miguel.** Caso práctico de la metodología ágil XP al desarrollo de software [En línea]. Thesis. Universidad Tecnológica de Pereira, 2007. [Accedido 8 Mayo 2014]. Disponible en: <http://recursosbiblioteca.utp.edu.co/dspace/bitstream/11059/794/1/0053E18cp.pdf>.
55. **GUTIÉRREZ, J. J, ESCALONA, M. J., MEJÍAS, M. and TORRES, J.** PRUEBAS DEL SISTEMA EN PROGRAMACIÓN EXTREMA [En línea]. Department de Lenguajes y Sistemas Informáticos University of Sevilla. [Accedido 5 Mayo 2014]. Disponible en: http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf.
56. **VIVAS, Luis, CAMBARIERI, Mauro, GARCÍA MARTÍNEZ, Nicolás, PETROFF, Marcelo and MUÑOZ ABBATE, Horacio.** Un marco de trabajo para la Integración de Arquitecturas de Software con Metodologías Ágiles de Desarrollo. In: XVIII Congreso Argentino de Ciencias de la Computación [En línea]. Octubre 2013. [Accedido 5 Mayo 2014]. Disponible en: http://sedici.unlp.edu.ar/bitstream/handle/10915/31759/Documento_completo.pdf?sequence=1.
57. **ORÉ, Ing. Alexander.** Pruebas Funcionales - Software Testing and QA. [En línea]. 2009. [Accedido 5 Mayo 2014]. Disponible en: http://www.calidadyssoftware.com/testing/pruebas_funcionales.php.

Anexos de la investigación

Anexo #1: Pantalla principal de la Multimedia de Aprendizaje de Modelación Matemática.



Figura 3 Pantalla principal de la Multimedia de Aprendizaje de Modelación Matemática.

Anexo #2: Pantalla principal de la Multimedia de Ejercitación de Modelación Matemática, Modele la Vida.



Figura 4 Pantalla principal de la Multimedia de Ejercitación de Modelación Matemática, Modele la Vida.

Anexo #3: Pantalla principal de la Multimedia de Aprendizaje Simulación.



Figura 5 Pantalla principal de la Multimedia de Aprendizaje Simulación.

Anexo #4: Pantalla del Glosario de Términos.

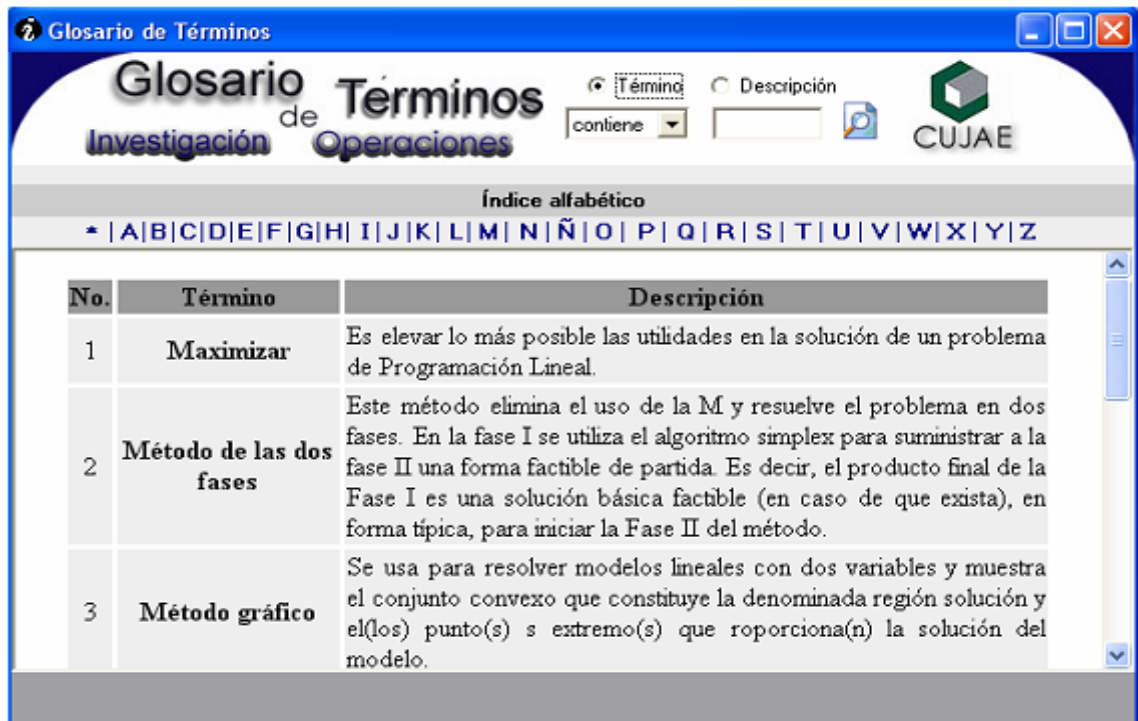


Figura 6 Pantalla del Glosario de Términos.

Anexo #5: Diagramas de presentación de la multimedia.

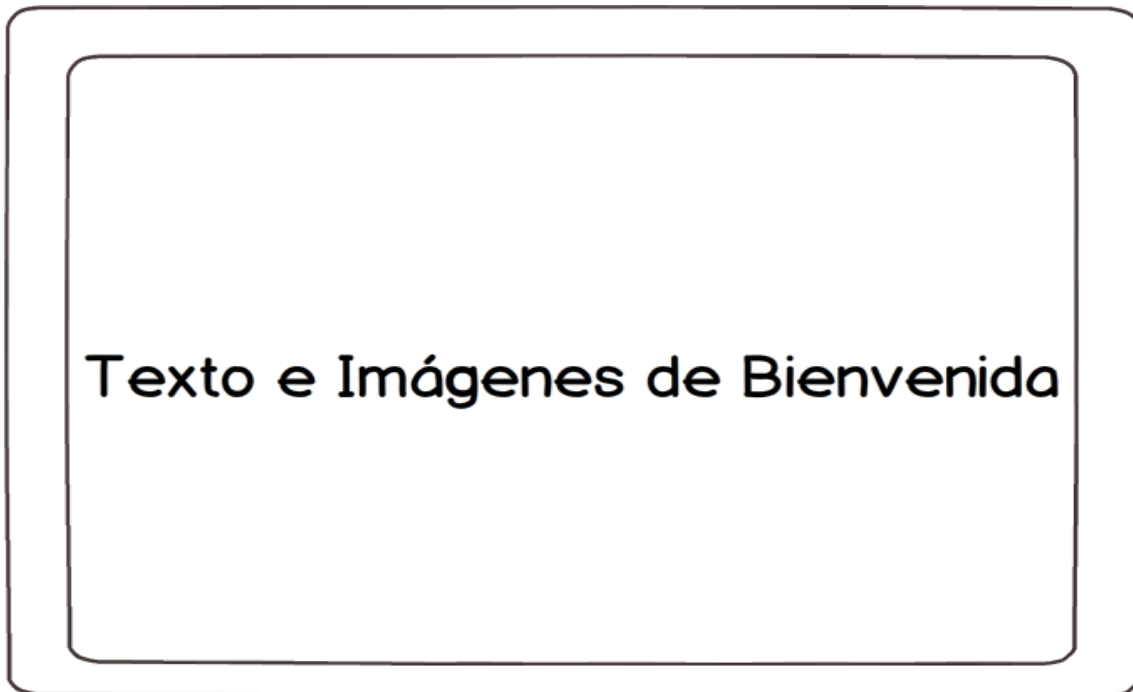


Figura 7 Diagrama de Presentación Principal

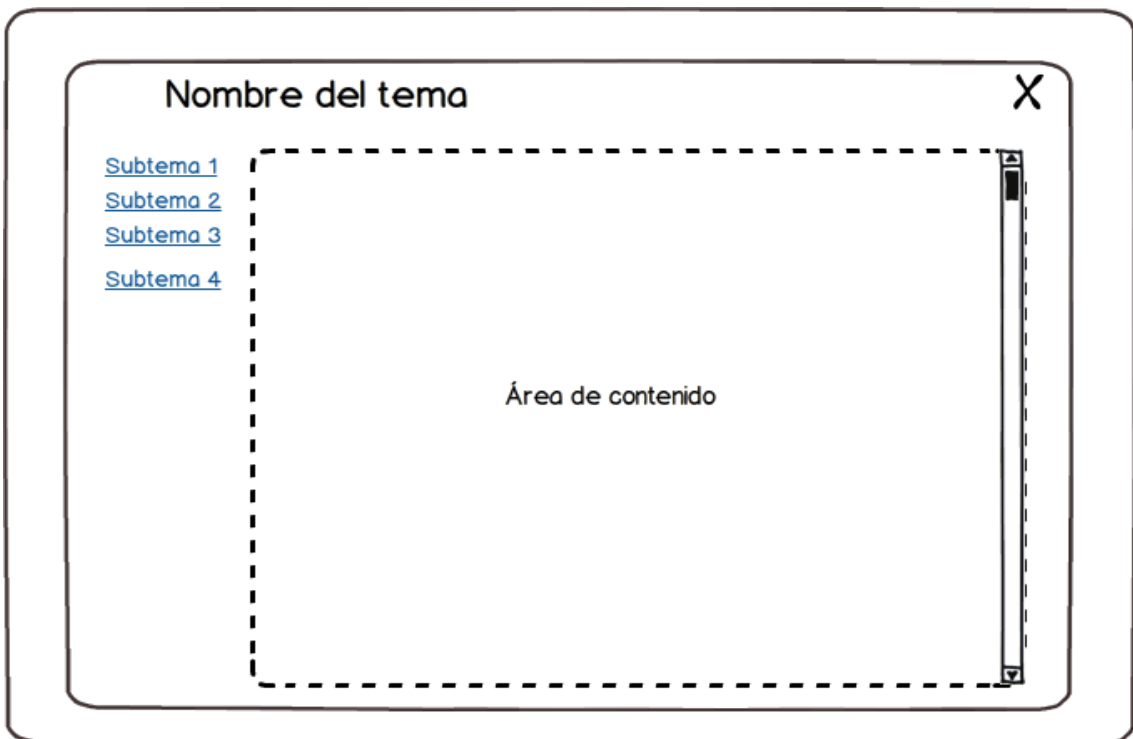


Figura 8 Diagrama de Presentación Temas

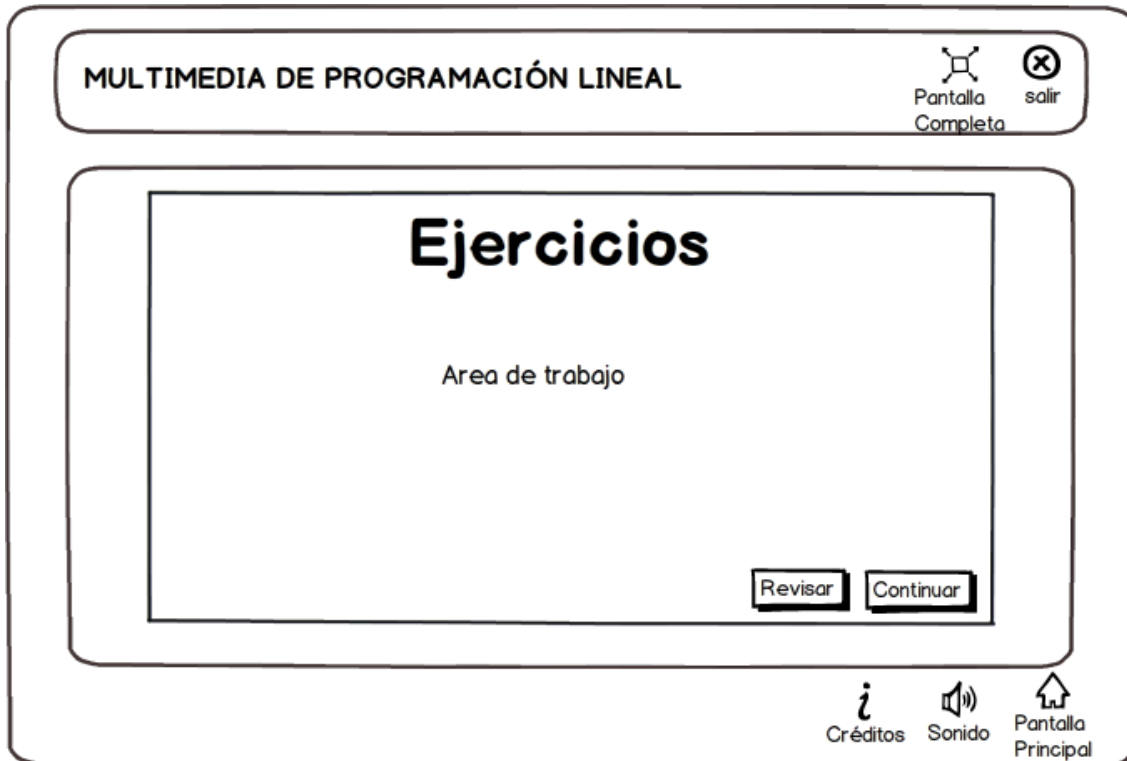


Figura 9 Diagrama de Presentación Ejercicios

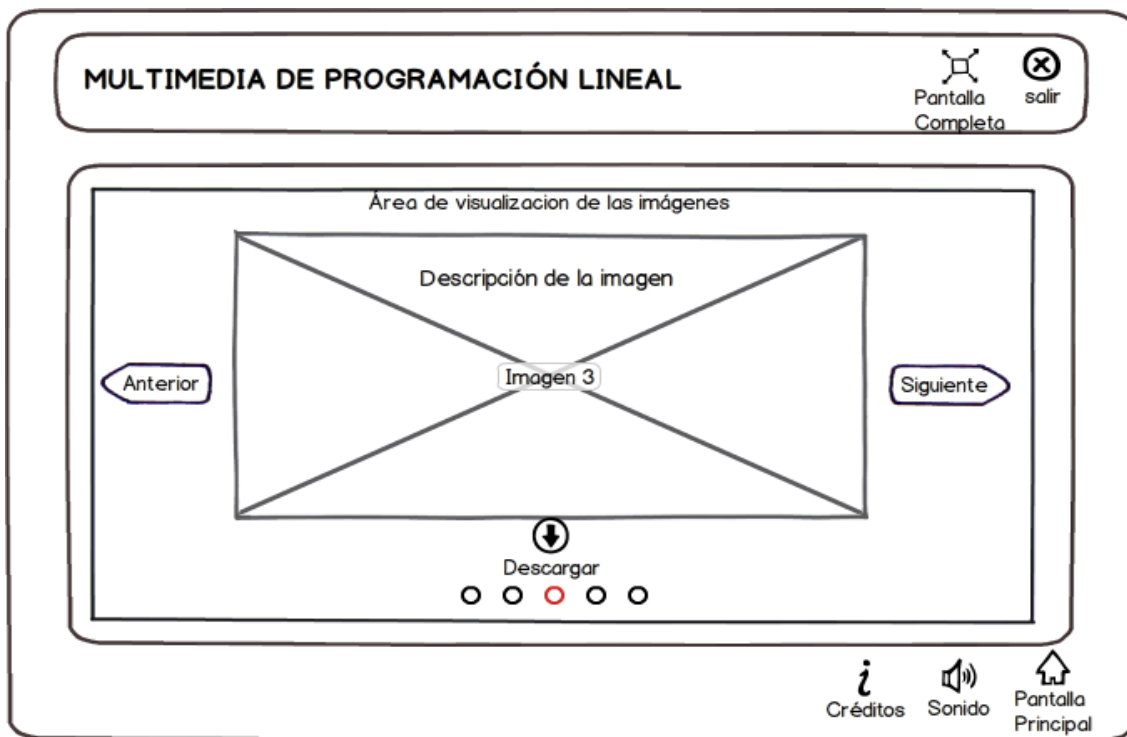


Figura 10 Diagrama de Presentación Galería de Imágenes

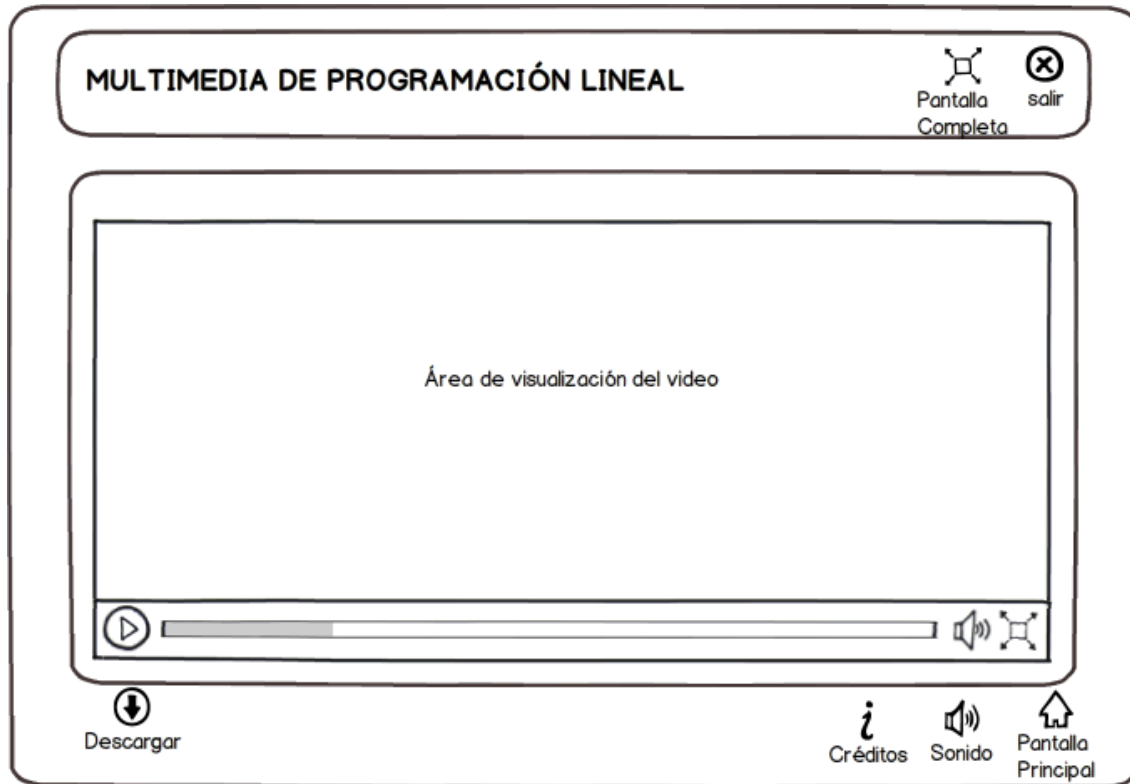


Figura 11 Diagrama de Presentación Galería de Video

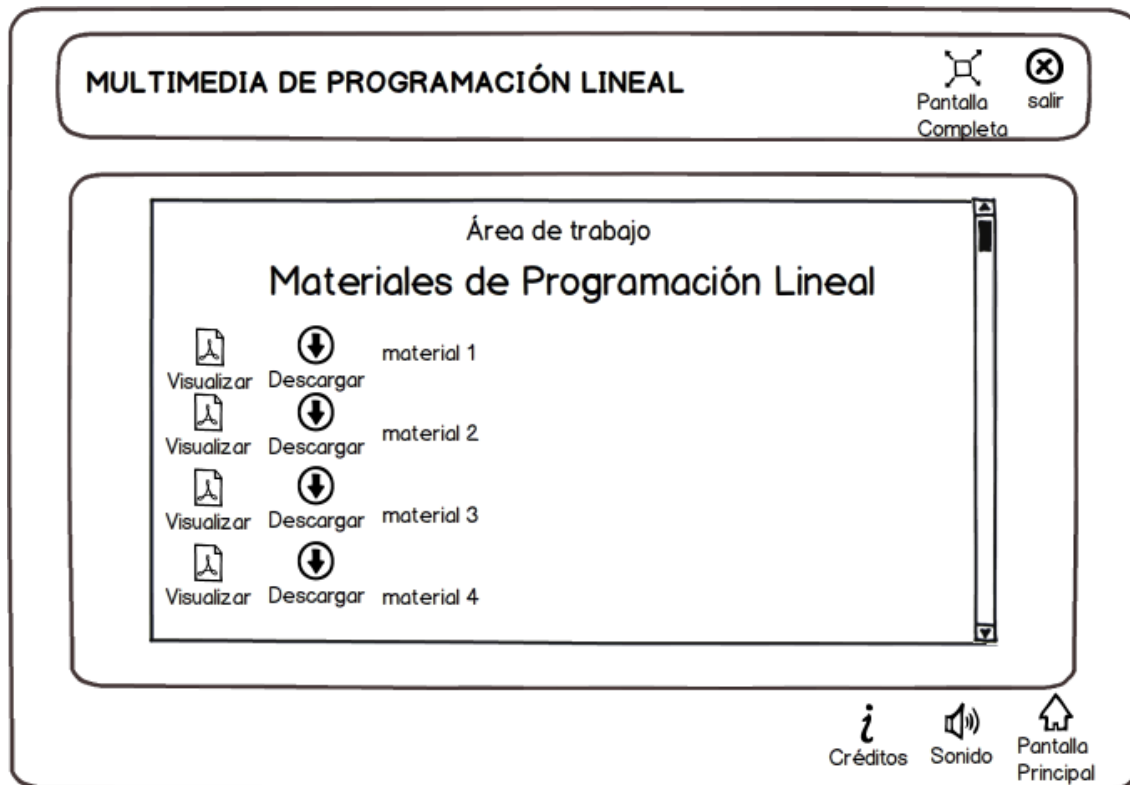


Figura 12 Diagrama de Presentación Biblioteca

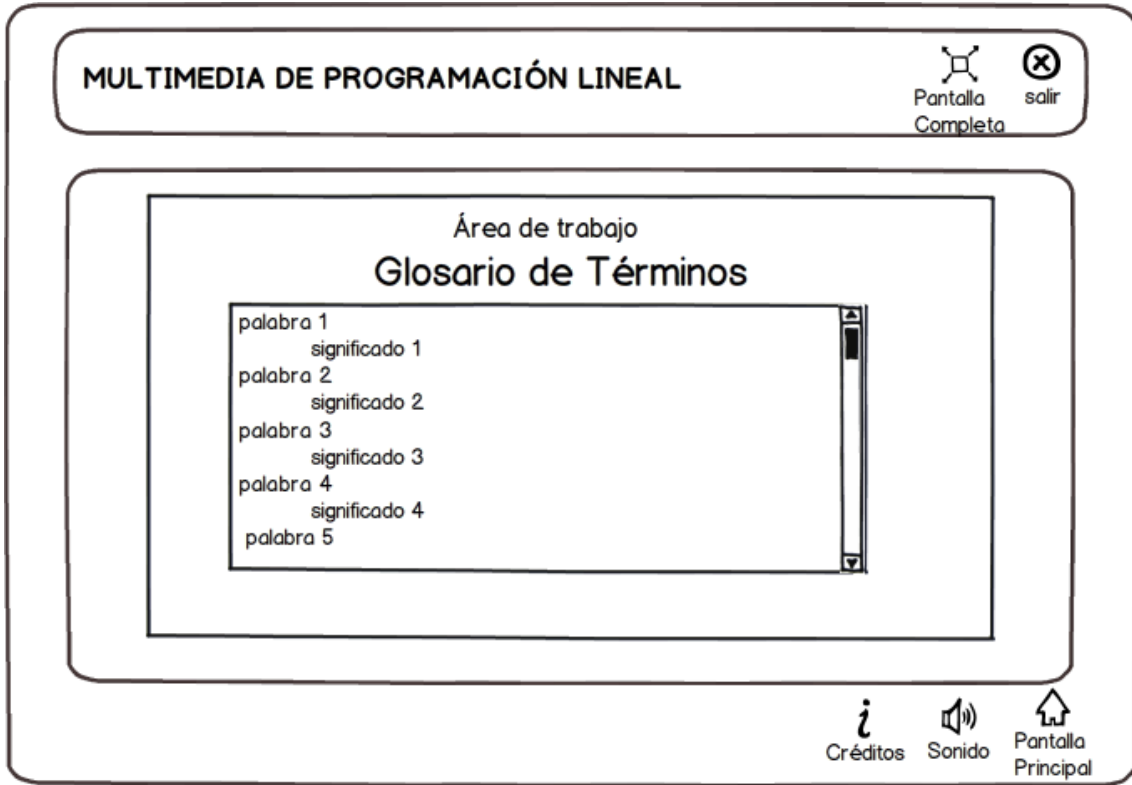


Figura 13 Diagrama de Presentación Glosario de Términos

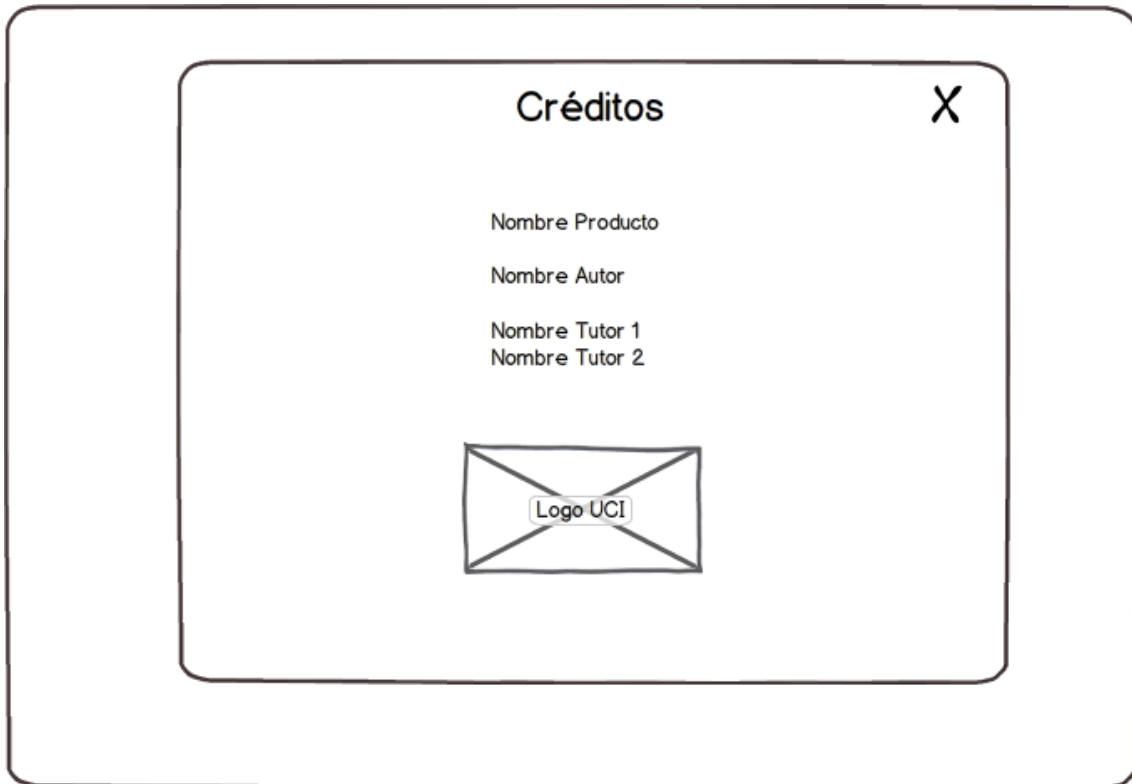


Figura 14 Diagrama de Presentación Créditos

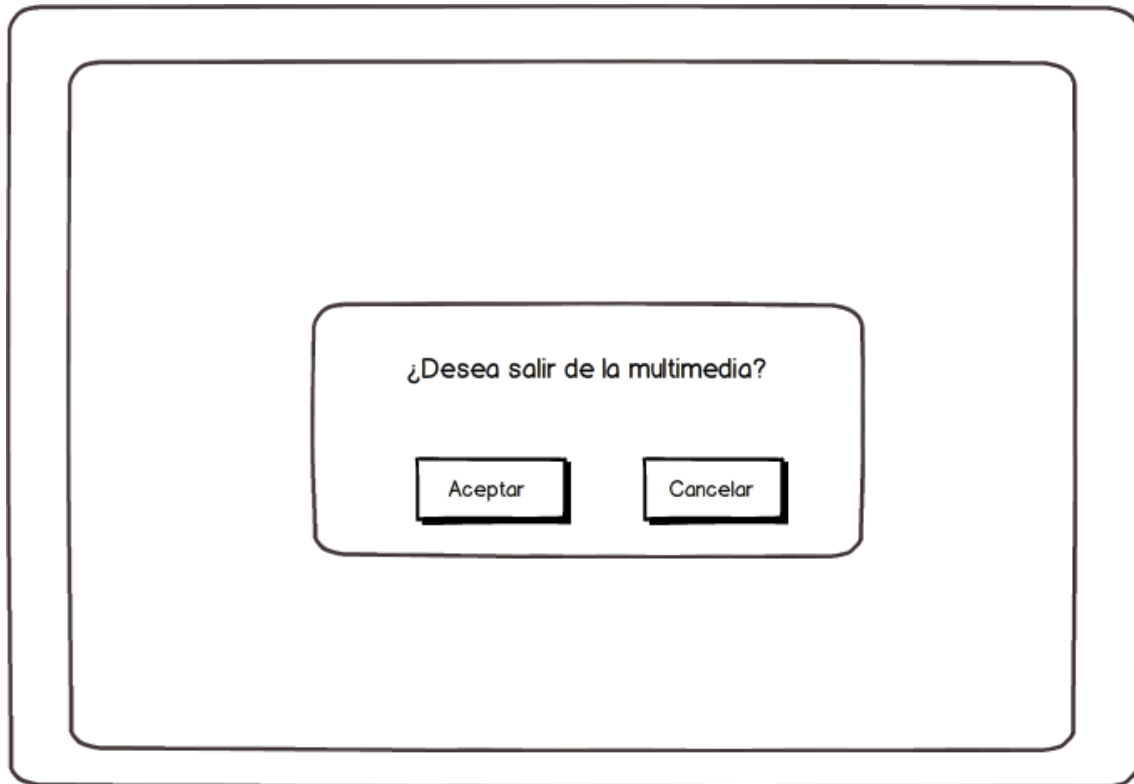


Figura 15 Diagrama de Presentación Salir

Anexo #6: Historias de usuario.

Historia de usuario	
Número: 5	Nombre: Permitir el control del sonido de la aplicación.
Usuario: Cliente	
Prioridad: Baja	Riesgo: Bajo
Estimación: 3 días	Iteración: 3
Descripción:	
Permite al usuario escuchar un fondo musical mientras está en la multimedia. El sistema permite cambiar el estado actual del audio presionado el botón correspondiente al audio.	
Observación:	

Tabla 41 HU-Permitir el control del sonido de la aplicación.

Historia de usuario	
Número: 6	Nombre: Mostrar los créditos de la multimedia.
Usuario: Cliente	

Prioridad: Baja	Riesgo: Bajo
Estimación: 3 días	Iteración: 3
Descripción: El sistema le da la posibilidad al usuario ver los créditos de la multimedia presionando el botón <i>Información</i> .	
Observación:	

Tabla 42 HU- Mostrar los créditos de la multimedia

Historia de usuario	
Número: 7	Nombre: Seleccionar la sección Galería de Imágenes
Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 4 días	Iteración: 1
Descripción: Inicia cuando el usuario accede a la opción galería de imágenes y se le muestran todas las imágenes presentes.	
Observación:	

Tabla 43 HU-Seleccionar la sección Galería de Imágenes

Historia de usuario	
Número: 8	Nombre: Seleccionar la sección Galería de Videos
Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 4 días	Iteración: 1
Descripción: Inicia cuando el usuario accede a la opción galería de videos y se le muestran los videos presentes.	
Observación:	

Tabla 44 HU-Seleccionar la sección Galería de Videos

Historia de usuario	
Número: 9	Nombre: Seleccionar la sección Glosario de Términos

Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 3 días	Iteración: 2
Descripción: Se muestran palabras de poco entendimiento para el usuario.	
Observación:	

Tabla 45 HU-Seleccionar la sección Glosario de Términos

Historia de usuario	
Número: 10	Nombre: Mostrar palabras calientes
Usuario: Cliente	
Prioridad: Baja	Riesgo: Bajo
Estimación: 3 días	Iteración: 3
Descripción: Inicia cuando el usuario selecciona dentro de la información de los temas una palabra y se muestra una imagen correspondiente al tema.	
Observación:	

Tabla 46 HU-Mostrar palabras calientes

Historia de usuario	
Número: 11	Nombre: Seleccionar la sección Biblioteca.
Usuario: Cliente	
Prioridad: Alta	Riesgo: Alto
Estimación: 4 días	Iteración: 2
Descripción: Inicia cuando el usuario accede a la sección Biblioteca, mostrándole materiales de consulta que lo ayudará en el estudio del tema, los cuales consisten en documentos en formato PDF.	
Observación:	

Tabla 47 HU-Seleccionar la sección Biblioteca

Historia de usuario	
Número: 12	Nombre: Permitir descargar media.
Usuario: Cliente	
Prioridad: Baja	Riesgo: Bajo
Estimación: 3 días	Iteración: 3
Descripción: El sistema brinda la posibilidad de descargar cualquier tipo de media (imágenes, videos, documentos) que el usuario desee.	
Observación:	

Tabla 48 HU-Permitir descargar media

Historia de usuario	
Número: 13	Nombre: Mostrar a pantalla completa.
Usuario: Cliente	
Prioridad: Media	Riesgo: Medio
Estimación: 2 días	Iteración: 2
Descripción: El sistema permite al usuario la opción de ver la multimedia en pantalla completa.	
Observación:	

Historia de usuario	
Número: 14	Nombre: Salir del sistema.
Usuario: Cliente	
Prioridad: Media	Riesgo: Medio
Estimación: 3 días	Iteración: 3

<p>Descripción:</p> <p>El sistema permite al usuario salir desde cualquier sección que se encuentre. Se muestra una pequeña pantalla de confirmación de salida y en caso de ser aceptada se cierra la multimedia.</p>
<p>Observación:</p>

Tabla 49 HU-Salir del sistema

Anexo #7: Tareas de ingeniería.

Tarea	
Número de tarea: 12	Número de la Historia de Usuario: 5
Nombre de la tarea: Permitir escuchar fondo musical.	
Tipo de tarea: Desarrollo	Estimación: 3 días
Fecha inicio: 24 de abril 2014	Fecha fin: 28 de abril 2014
Programador responsable: Omar D. Castillo González	
<p>Descripción:</p> <p>Se ofrece la posibilidad al usuario de escuchar un fondo musical mientras este dentro de la multimedia.</p>	

Tabla 50 Tarea #12

Tarea	
Número de tarea: 13	Número de la Historia de Usuario: 6
Nombre de la tarea: Crear créditos.	
Tipo de tarea: Desarrollo	Estimación: 3 días
Fecha inicio: 29 de abril 2014	Fecha fin: 1 de mayo 2014
Programador responsable: Omar D. Castillo González	
<p>Descripción:</p> <p>Se confeccionan los créditos del producto.</p>	

Tabla 51 Tarea #13

Tarea	
Número de tarea: 14	Número de la Historia de Usuario: 7
Nombre de la tarea: Crear galería de imágenes.	
Tipo de tarea: Desarrollo	Estimación: 3 días

Fecha inicio: 7 de marzo 2014	Fecha fin: 11 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se confecciona un menú para la galería de imágenes que permitirá al usuario ver las imágenes con que cuenta la aplicación.	

Tabla 52 Tarea #14

Tarea	
Número de tarea: 15	Número de la Historia de Usuario: 7
Nombre de la tarea: Mostrar imágenes.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 12 de marzo 2014	Fecha fin: 12 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Brinda la posibilidad al usuario de ver las imágenes con una descripción.	

Tabla 53 Tarea #15

Tarea	
Número de tarea: 16	Número de la Historia de Usuario: 8
Nombre de la tarea: Mostrar videos.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 13 de marzo 2014	Fecha fin: 14 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se carga el video en pantalla.	

Tabla 54 Tarea #16

Tarea	
Número de tarea: 17	Número de la Historia de Usuario: 8
Nombre de la tarea: Permitir reproducir videos.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 17 de marzo 2014	Fecha fin: 18 de marzo 2014
Programador responsable: Omar D. Castillo González	

Descripción:

Cuando el usuario seleccione opción reproducir video. Se le ofrece la posibilidad de llevar el control del reproductor como: volumen, iniciar, detener, adelantar y atrasar.

Tabla 55 Tarea #17

Tarea	
Número de tarea: 18	Número de la Historia de Usuario: 9
Nombre de la tarea: Mostrar glosario de términos.	
Tipo de tarea: Desarrollo	Estimación: 3 días
Fecha inicio: 24 de marzo 2014	Fecha fin: 26 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Mostrar la pantalla donde se visualiza el listado de términos.	

Tabla 56 Tarea #18

Tarea	
Número de tarea: 19	Número de la Historia de Usuario: 10
Nombre de la tarea: Crear vínculos en las palabras calientes.	
Tipo de tarea: Desarrollo	Estimación: 3 días
Fecha inicio: 2 de mayo 2014	Fecha fin: 6 de mayo 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Permite que el usuario desee conocer al momento información adicional del texto en el que se encuentra.	

Tabla 57 Tarea #19

Tarea	
Número de tarea: 20	Número de la Historia de Usuario: 11
Nombre de la tarea: Listar documentos.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 27 de marzo 2014	Fecha fin: 28 de marzo 2014
Programador responsable: Omar D. Castillo González	
Descripción:	
Se cargan los documentos con formato PDF y se muestran en una lista.	

Tabla 58 Tarea #20

Tarea	
Número de tarea: 21	Número de la Historia de Usuario: 11
Nombre de la tarea: Permitir leer documentos.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 31 de marzo 2014	Fecha fin: 1 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción: Brinda la posibilidad al usuario de abrir los documentos.	

Tabla 59 Tarea #21

Tarea	
Número de tarea: 22	Número de la Historia de Usuario: 12
Nombre de la tarea: Permitir descargar video.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 7 de mayo 2014	Fecha fin: 7 de mayo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se ofrece la posibilidad de descargar los videos con los que cuenta el producto.	

Tabla 60 Tarea #22

Tarea	
Número de tarea: 23	Número de la Historia de Usuario: 12
Nombre de la tarea: Permitir descargar documento.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 8 de mayo 2014	Fecha fin: 8 de mayo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se ofrece la posibilidad de descargar los diferentes documentos con que cuenta el producto.	

Tabla 61 Tarea #23

Tarea	
Número de tarea: 24	Número de la Historia de Usuario: 12
Nombre de la tarea: Permitir descargar imágenes.	

Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 9 de mayo 2014	Fecha fin: 9 de mayo 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se ofrece la posibilidad al usuario de descargar las imágenes con que cuenta el producto.	

Tabla 62 Tarea #24

Tarea	
Número de tarea: 25	Número de la Historia de Usuario: 13
Nombre de la tarea: Permitir mostrar a pantalla completa.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 14 de abril 2014	Fecha fin: 15 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se ofrece la posibilidad al usuario de poder interactuar con la multimedia a pantalla completa.	

Tabla 63 Tarea #25

Tarea	
Número de tarea: 26	Número de la Historia de Usuario: 14
Nombre de la tarea: Permitir salir desde cualquier parte del producto.	
Tipo de tarea: Desarrollo	Estimación: 2 días
Fecha inicio: 21 de abril 2014	Fecha fin: 22 de abril 2014
Programador responsable: Omar D. Castillo González	
Descripción: Se ofrece la posibilidad al usuario de poder salir desde cualquier lugar donde se encuentre en la multimedia.	

Tabla 64 Tarea #26

Tarea	
Número de tarea: 27	Número de la Historia de Usuario: 14
Nombre de la tarea: Crear ventana de comprobación de salida.	
Tipo de tarea: Desarrollo	Estimación: 1 día
Fecha inicio: 23 de abril 2014	Fecha fin: 23 de abril 2014

Programador responsable: Omar D. Castillo González
Descripción: Se crea una pequeña ventana para confirmar la salida del sistema.

Tabla 65 Tarea #27

Anexo #8: Casos de prueba de aceptación.

Caso de prueba de aceptación	
Código: HU5_CP8	Historia de usuario: 5
Nombre: Permitir escuchar fondo musical.	
Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de escuchar un fondo musical.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta acceder al ícono del sonido, brindando la posibilidad de pausar el fondo musical.	
Resultado esperado: Se brinda la posibilidad de que el usuario pueda o no escuchar un fondo musical mientras esté dentro del producto.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 66 Caso de prueba de aceptación #8

Caso de prueba de aceptación	
Código: HU6_CP9	Historia de usuario: 6
Nombre: Mostrar los créditos de la multimedia.	
Descripción: Prueba para la funcionalidad de mostrar al usuario los créditos.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta acceder al botón Información.	
Resultado esperado: Se brinda la posibilidad de que el usuario pueda ver los créditos de la multimedia.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 67 Caso de prueba de aceptación #9

Caso de prueba de aceptación	
Código: HU7_CP10	Historia de usuario: 7
Nombre: Seleccionar Galería de Imágenes.	

Descripción: Prueba para la funcionalidad de que el usuario seleccione las diferentes imágenes contenidas dentro de la galería.
Condiciones de ejecución: El usuario debe acceder a la multimedia.
Entrada/ Pasos de ejecución: Se intenta seleccionar las imágenes contenidas dentro de la galería.
Resultado esperado: Se facilita que el usuario pueda seleccionar las imágenes contenidas en la galería.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 68 Caso de prueba de aceptación #10

Caso de prueba de aceptación	
Código: HU7_CP11	Historia de usuario: 7
Nombre: Mostrar Imágenes	
Descripción: Prueba para la funcionalidad de mostrar al usuario las imágenes contenidas dentro de la galería con una breve descripción.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta ver las imágenes contenidas dentro de la galería.	
Resultado esperado: Se facilita que el usuario pueda ver las imágenes contenidas en la galería con su descripción.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 69 Caso de prueba de aceptación #11

Caso de prueba de aceptación	
Código: HU8_CP12	Historia de usuario: 8
Nombre: Seleccionar Galería de Videos	
Descripción: Prueba para la funcionalidad de mostrar al usuario los videos contenidos dentro de la galería.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta ver los videos contenidos dentro de la galería.	
Resultado esperado: Se facilita que el usuario tenga la posibilidad de ver los videos contenidos en la galería.	

Evaluación de la prueba: Prueba satisfactoria.

Tabla 70 Caso de prueba de aceptación #12

Caso de prueba de aceptación	
Código: HU8_CP13	Historia de usuario: 8
Nombre: Permitir reproducir videos.	
Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de reproducir un determinado video de la galería.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta reproducir el video seleccionado.	
Resultado esperado: Se brinda la posibilidad al usuario de reproducir el video de la aplicación y de poder trabajar con las opciones de video.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 71 Caso de prueba de aceptación #13

Caso de prueba de aceptación	
Código: HU9_CP14	Historia de usuario: 9
Nombre: Mostrar glosario de términos	
Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de acceder al glosario de términos.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta acceder al glosario de términos.	
Resultado esperado: Se facilita que el usuario acceda al glosario de términos brindando la posibilidad de buscar un vocablo desconocido.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 72 Caso de prueba de aceptación #14

Caso de prueba de aceptación	
Código: HU10_CP15	Historia de usuario: 10
Nombre: Mostrar palabras calientes	
Descripción: Prueba para la funcionalidad de mostrar al usuario información adicional del contenido, mediante la interacción con una palabra caliente.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	

Entrada/ Pasos de ejecución: Se intenta interactuar con la palabra caliente seleccionada.
Resultado esperado: Se facilita que el usuario pueda conocer información adicional del contenido. Permite mostrar la ventana de contenido correspondiente a dicha palabra.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 73 Caso de prueba de aceptación #15

Caso de prueba de aceptación	
Código: HU11_CP16	Historia de usuario: 11
Nombre: Seleccionar Biblioteca	
Descripción: Prueba para la funcionalidad de mostrar un listado de documentos.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta acceder a la sección Biblioteca.	
Resultado esperado: Permite mostrar el listado de los materiales de consulta dándole la posibilidad de seleccionar el deseado.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 74 Caso de prueba de aceptación #16

Caso de prueba de aceptación	
Código: HU11_CP17	Historia de usuario: 11
Nombre: Permitir leer documentos.	
Descripción: Prueba para la funcionalidad de leer los documentos con formato PDF de la aplicación.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta ver los documentos.	
Resultado esperado: Se brinda la posibilidad de que el usuario observe el documento que desee y en formato PDF.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 75 Caso de prueba de aceptación #17

Caso de prueba de aceptación	
Código: HU12_CP18	Historia de usuario: 12
Nombre: Permitir descargar media.	

Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de descargar la media (imagen, documento, video) deseada.
Condiciones de ejecución: El usuario debe acceder a la multimedia.
Entrada/ Pasos de ejecución: Se intenta acceder a la opción descargar.
Resultado esperado: Se facilita que el usuario acceda a la opción descargar, brindando la posibilidad de descargar la media deseada.
Evaluación de la prueba: Prueba satisfactoria.

Tabla 76 Caso de prueba de aceptación #18

Caso de prueba de aceptación	
Código: HU13_CP19	Historia de usuario: 13
Nombre: Mostrar a pantalla completa.	
Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de mostrar la multimedia a pantalla completa.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta acceder a la opción de mostrar a pantalla completa.	
Resultado esperado: Permite que el usuario pueda navegar en la multimedia a pantalla completa.	
Evaluación de la prueba: Prueba satisfactoria.	

Tabla 77 Caso de prueba de aceptación #19

Caso de prueba de aceptación	
Código: HU14_CP20	Historia de usuario: 14
Nombre: Salir del sistema.	
Descripción: Prueba para la funcionalidad de mostrar al usuario la opción de salir de la aplicación.	
Condiciones de ejecución: El usuario debe acceder a la multimedia.	
Entrada/ Pasos de ejecución: Se intenta salir de la aplicación desde cualquier pantalla del producto multimedia.	
Resultado esperado: Permite mostrar el mensaje de confirmación de salida.	

Evaluación de la prueba: Prueba satisfactoria.

Tabla 78 Caso de prueba de aceptación #20