

*Séptima Conferencia de Directores de Tecnología de Información, TICAL  
2017 Gestión de las TICs para la Investigación y la Colaboración, San  
José, del 3 al 5 de julio de 2017*

## **SIMULADOR PARA EL SIMULADOR: DISEÑO E IMPLEMENTACIÓN DE UN APLICATIVO MULTIMEDIAL PARA EL PROCESO DE OPERACIÓN Y MANTENIMIENTO DEL SIMULADOR DE VUELO DEL EQUIPO T-90**

José Tulio Nel Benavides Peña<sup>a</sup>, Gean Marco López Palacio<sup>b</sup>, Diana Marcela  
Rodríguez Toro<sup>c</sup>

<sup>a</sup> Fuerza Aerea Colombiana - Escuela Militar de Aviación “Marco Fidel Suárez”, Programa  
de Ingeniería Informática, Cra 8 No 58-67. Cali, Colombia  
[jtbenavides@emavirtual.edu.co](mailto:jtbenavides@emavirtual.edu.co), [tulionel@hotmail.com](mailto:tulionel@hotmail.com)

<sup>b</sup> Fuerza Aerea Colombiana - Escuela Militar de Aviación “Marco Fidel Suárez”, Programa  
de Ingeniería Informática, Cra 8 No 58-67. Cali, Colombia  
[89gmlopezp@emavirtual.edu.co](mailto:89gmlopezp@emavirtual.edu.co)

<sup>c</sup> Fuerza Aerea Colombiana - Escuela Militar de Aviación “Marco Fidel Suárez”, Programa  
de Ingeniería Informática, Cra 8 No 58-67. Cali, Colombia  
[89dmrodriguez@emavirtual.edu.co](mailto:89dmrodriguez@emavirtual.edu.co)

**Resumen.** El avión T-90 Calima es un avión construido por la Fuerza Aérea Colombiana (FAC) y la Corporación de la Industria Aeronáutica Colombiana (CIAC), actualmente junto con el T-41 Mescalero son utilizados para la instrucción primaria de los cadetes de la Escuela Militar de Aviación (EMAVI) y los miembros de la Fuerza Pública, EMAVI, con el fin de facilitar el aprendizaje de sus alumnos de vuelo y contribuir con el ahorro de los recursos de la nación, construye un simulador del equipo T-90 Calima tipo FTD (Flight Training Device), del cual existe un manual de operación y mantenimiento impreso. Debido a los adelantos de la tecnología se encuentra oportuno desarrollar un aplicativo multimedial didáctico y amigable para la operación y mantenimiento de dicho simulador, con el fin de que el usuario pueda operar el simulador por si solo y realizar un mantenimiento básico, teniendo resultados favorables para los alumnos de vuelo del avión T-90.

**Palabras Clave:** Multimedia, Simulador, Mantenimiento, Operación, T-90 Calima, Programación Extrema.

**Eje temático:** Soluciones TIC para la Enseñanza.

## **1 Introducción**

La Escuela Militar de Aviación Marco Fidel Suárez (EMAVI) ubicada en la ciudad de Cali Valle del Cauca, tiene la misión de Formar integralmente los futuros oficiales de la Fuerza Aérea Colombiana (FAC) y capacitar como pilotos de ala fija a oficiales de la fuerza pública. Esta formación se realiza en las aeronaves T-41D “Mescalero” y T-90 “Calima”.

Inicialmente este proceso de aprendizaje es realizado obteniendo los conocimientos básicos de funcionamiento de la aeronave mediante un curso de tierra; posteriormente se realizan las prácticas en la aeronave T-90. Dado el alto costo y demás riesgos asociados que conlleva impartir instrucción directamente en la aeronave, la industria aeronáutica desarrolla dispositivos simuladores para entrenamiento de vuelo, equipos que permiten alcanzar la destreza y pericia necesaria para una correcta operación de aeronaves civiles y militares.

En la Escuela Militar de Aviación Marco Fidel Suárez, se cuenta con un simulador del T-90, el cual requiere la asistencia técnica para mantenimiento; el Suboficial que ayudó a su construcción, es la única persona capacitada para la realización de actividades preventivas y predictivas con el equipo. Actualmente no se cuenta con este personal en el área, lo que ocasiona que ante cualquier evento de falla el equipo quede deshabilitado, ocasionando pérdida de tiempo y reprocesos en programaciones de uso.

El diseño y fabricación de un manual multimedial para la operación y el mantenimiento del simulador de vuelo del equipo T-90, brinda soluciones al proceso de mantenimiento, instrucción y entrenamiento de vuelo, que se desarrolla en la EMAVI; logrando el desarrollo de competencias, aptitudes y destrezas, a través de un aprendizaje activo, que libera al operario de tiempos y espacios fijos de capacitación, lo que optimiza los movimientos de personal y acorta los procesos de la formación de los futuros pilotos de la Fuerza Aérea de Colombiana.

## **2 Metodología**

Se maneja la investigación descriptiva debido a que ésta permite conocer las situaciones de descripción y los procesos que se deben llevar a cabo para lograr un apropiado mantenimiento del simulador de vuelo del equipo T-90. Además, se busca mostrar nuevas estrategias y experiencias vividas por el usuario.

El enfoque es de tipo cualitativo y es desarrollado como un aplicativo multimedia tipo tutorial según su finalidad debido que funciona como un tutor virtual para mostrar diferentes procesos a quien lo use, con una navegación tipo reticular para permitir la libertad de seguir diferentes caminos hipertexto.

El desarrollo de este proyecto se basa en la metodología XP, debido a que esta se acomoda a cualquier tipo de proyecto, aplica para equipos de personas de cualquier tamaño, es una metodología fácil de implementar y aprender, presenta una notable disminución del protocolo y de jerarquías y por último introduce una autoevaluación intensiva en cada nivel.

### 3. Objetivos.

**Objetivo General.** El objetivo general del proyecto es:

- Diseñar e implementar un aplicativo multimedial para el proceso de operación y mantenimiento del simulador de vuelo del equipo T-90.

**Objetivos Específicos.** Los objetivos específicos del son:

- Identificar los requerimientos funcionales y no funcionales del simulador de vuelo del equipo T-90.
- Diseñar el aplicativo multimedial para el mantenimiento del simulador de vuelo del T-90 de manera amigable.
- Implementar y realizar pruebas para comprobar el funcionamiento del aplicativo multimedial para el mantenimiento del simulador de vuelo del equipo T-90.

### 4 Marco Conceptual

En el transcurso de desarrollo del aplicativo multimedial para el proceso de operación y del simulador del equipo T-90 en la EMAVI, se requiere que el usuario tenga conocimiento de los conceptos utilizados en el sistema de gestión de conocimiento, y de esta forma lograr tener un panorama más amplio de lo que está ofreciendo el manual para el usuario.

**Software.** El software no son sólo programas, sino todos los documentos asociados y la configuración de datos que se necesitan para hacer que estos programas operen de manera correcta. Por lo general, un sistema de software consiste en diversos programas independientes, archivos de configuración que se utilizan para ejecutar estos programas, un sistema de documentación que describe la estructura del sistema, la documentación para el usuario que explica cómo utilizar el sistema y sitios web que permitan a los usuarios descargar la información de productos recientes.

**Ingeniería de software.** Sabiendo que ingeniería hace referencia al ingenio, aplicando este principio al software, se puede afirmar que la ingeniería de software es el ingenio utilizado en los softwares ya sea desde su diseño, construcción y aplicación. “La ingeniería del software es el establecimiento y uso de principios sólidos de la ingeniería para obtener económicamente un software confiable y que funcione de modo eficiente en máquinas reales” [4]

**Programación Extrema (XP).** Es una de las metodologías del desarrollo ágil, consisten en satisfacer las necesidades del cliente adaptándose a ciertas reglas para lograr obtener un producto de calidad en poco tiempo. “la programación extrema abarca un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades del marco de trabajo: planeación, diseño, codificación y pruebas”. La Programación Extrema (XP) es posiblemente el método ágil más conocido y ampliamente utilizado. El nombre fue acuñado por Beck [4] debido a que el enfoque

fue desarrollado utilizando buenas prácticas reconocidas, como el desarrollo iterativo, y con la participación del cliente en niveles «extremos». Las fases en que esta metodología se divide son: Planificación, diseño, desarrollo o codificación y pruebas de aceptación como se muestra en el ciclo, Figura; además de esto el equipo de programación aborda la tendencia a degradar la estructura del software por los cambios que se deban realizar como suele presentarse en las metodologías de desarrollo tradicional, es decir busca posibles mejoras del software y las implementa inmediatamente, por tanto el software siempre debe ser fácil de entender y cambiar.

**Lenguaje HTML.** El lenguaje utilizado para desarrollar páginas web, el lenguaje “HTML” (HyperText Markup Lenguaje) es uno de los lenguajes más utilizados por los profesores, su popularidad se debe principalmente a su facilidad de uso para desarrollar páginas, y a las potencialidades educativas que tienen estas páginas. El lenguaje HTML, desarrollado por Tim Berners-Lee en 1991, parte del uso de comandos o etiquetas (tags) que deben incorporarse atendiendo a ciertas reglas. Este lenguaje nos permite programar las páginas web aprovechando sus diferentes posibilidades (inclusión de imágenes y/o animaciones, realización de enlaces, etc.). Las diferentes versiones que se han realizado en el lenguaje HTML han posibilitado un aumento de sus potencialidades tanto a nivel general como educativo. Html5 propone estándares para cada aspecto de la web y también un propósito claro para cada una de las tecnologías involucradas. HTML provee los elementos estructurales, CSS se encuentra concentrado en cómo volver esa estructura utilizable y atractiva a la vista, y Javascript tiene todo el poder necesario para proveer dinamismo y construir aplicaciones web completamente funcionales.

**Multimedia.** El término multimedia hace referencia al conjunto de diferentes medios de comunicación combinados como texto, gráfico, imagen, sonido, animación, video, entre otras. Tiene como fin permitir facilidad en el aprendizaje del usuario gracias a la interacción con estas aplicaciones multimedia.

**Simulador de vuelo.** “Es una réplica a tamaño real de un tipo específico de avión y la cabina, ya sea por su marca o modelo, incluye el conjunto de programas necesarios para representar la aeronave en operaciones en tierra y en vuelo, un sistema visual que proporciona el equipo y la computadora de una vista afuera de la cabina, y un sistema de fuerza que proporciona indicaciones equivalentes a la de un sistema de movimiento de tres grados de libertad.” [11]

**Adobe Muse CC.** Es una aplicación, producto de Adobe Systems, la cual crea páginas web en HTML5, utiliza las clásicas herramientas del software de Adobe y le añade otras destinadas a las zonas básicas de una web en forma WYSIWYG. Todo el proceso de trabajo está integrado en el resto de aplicaciones de la suite de Adobe y, mientras, es el propio Muse el que se encarga de crear el código necesario para hacerlo funcionar. Adobe Muse fue lanzado por primera vez el 7 de mayo de 2012, después de su lanzamiento se han creado mejoras en diferentes funciones, es un producto muy utilizado por los programadores en el mundo por eficiencia y facilidad al crear páginas web añadiendo archivos multimedia. [1]

## 5 Desarrollo del Aplicativo

El desarrollo del aplicativo multimedial para el proceso de operación y mantenimiento del simulador de vuelo del equipo t-90, suple las necesidades del Grupo de Educación Aeronáutica y del Grupo Técnico, los cuales necesitaban una aplicación fácil de usar donde estuviera consignado el funcionamiento, operación y mantenimiento del simulador del equipo T-90, esto con el fin de que los alumnos de vuelo pudieran utilizarlo sin la presencia de un operario, y poder corregir las eventuales fallas. Esto garantiza una operatividad del simulador en todo momento y un ahorro del recurso humano. Para iniciar el desarrollo de la aplicación multimedial se elige la metodología XP (programación extrema) para el desarrollo de software, siendo esta una metodología ágil, fácil de usar y muy efectiva para desarrollar aplicaciones en poco tiempo y trabajando de la mano con el cliente. A continuación, se presentan los aspectos relevantes de cada una de las fases del desarrollo del sistema:

### **Fase de Planificación.**

Esta etapa tendrá que desarrollarse de acuerdo a investigaciones preliminares ejecutadas por el equipo de desarrollo y dentro de algunas actividades a desarrollar se encuentran las historias de usuarios, programación en pareja, Release planning, Iteraciones, velocidad del proyecto, programación en pareja, reuniones diarias. Todas estas actividades permitirán tener al proyecto una distribución en cuanto al tiempo refiere y una buena utilización de los recursos que permitan un menor costo del proyecto para el debido cumplimiento de los requerimientos del cliente.

**Recolección de información.** Para la recolección de información con respecto al mantenimiento y operación del simulador se realizó la búsqueda de material bibliográfico impreso, artículos, manuales e información por medio magnético facilitada por el T1 Figueroa desarrollador y operario del simulador del equipo T-90. Uno de los artículos fue el artículo denominado DESARROLLO DE UN SIMULADOR TIPO “FLYGH T TRAINING DEVICE” DEL EQUIPO T90 CALIMA PARA LA ESCUELA MILITAR DE AVIACIÓN MARCO FIDEL SUAREZ. En este artículo se pretende mostrar cómo se concibió y desarrolló el simulador de vuelo para el equipo T90 Calima como herramienta de apoyo en la formación de los alumnos de vuelo de ala fija de la Fuerza Pública colombiana. El T1 Figueroa durante la construcción del simulador desarrolló artículos científicos de la operación y del mantenimiento básico del simulador. Se recuperaron estos documentos los cuales fueron esenciales para el desarrollo del aplicativo multimedia para el proceso de operación y mantenimiento del simulador de vuelo del equipo t-90.

**Historias de usuario.** Las historias de usuario son usadas para estimar tiempos de desarrollo de la parte de la aplicación que describen. También se utilizan en la fase de pruebas, para verificar si el programa cumple con lo que especifica la historia de usuario. Cuando llega la hora de implementar una historia de usuario, el cliente y los desarrolladores se reúnen para concretar y detallar lo que tiene que hacer dicha historia. El tiempo de desarrollo ideal para una historia de usuario es entre 1 y 3 semanas. En la figura 1 se muestra la realización de la historia de usuario Ingreso al Sistema, Adicionalmente se realizaron las historias de usuario Proceso de Operación, Proceso de Mantenimiento, Sistemas del Simulador y Buscar Fallas.

	DISEÑO E IMPLEMENTACIÓN DE UN APLICATIVO MULTIMEDIAL PARA EL PROCESO DE OPERACIÓN Y MANTENIMIENTO DEL SIMULADOR DE VUELO DEL EQUIPO T-90		Fecha
			1 <u>Abril</u> 2016
título de historia	Ingreso al sistema		
responsables	SBR. Rodríguez Toro y AF. López Palacio		
<b>Documentos de referencia</b>			
Prioridad	Alta		
Actores (Usuarios)	Administrador e invitado		
Estimación en tiempo	2 días		
Descripción:	La aplicación permitirá que el invitado y el administrador ingresen al sistema sin necesidad de un usuario o contraseña.		
Observaciones:	El Administrador es el único encargado de hacerle modificaciones al sistema.		
Pruebas a realizar			
Contacto	Nombre	SBR. Rodríguez Toro y AF. López Palacio	
	Teléfono	3176367321	
	Dirección	<a href="mailto:89gmlopezp@emavirtual.edu.co">89gmlopezp@emavirtual.edu.co</a>	
Fuente. Autores			

**Fig. 1.** Historia de Usuario, Ingreso al Sistema

**Release Planning.** Es una planificación donde los desarrolladores y clientes establecen los tiempos de implementación ideales de las historias de usuario, la prioridad con la que serán implementadas y las historias que serán implementadas en cada versión del programa. Véase figura 2, y figura 3.

	ABRIL	MAYO	JUNIO	JULIO	
	<b>FASE I: FASE DE PLANIFICACIÓN</b>				
	Iteración 1: <u>historias de usuario</u>				
	Iteración 2: Planificación y entrega				
	Iteración 3: Reuniones con el operario del simulador				
	Iteración 4: Estimación duración del proyecto				
Iteración 5: Programación en pareja					

Fig. 2. Cronograma de Iteraciones (Fase I)

HISTORIA DE USUARIO	TIEMPOS DE IMPLEMENTACION	PRIORIDAD
INGRESO AL SISTEMA	2 DIAS	ALTA
PROCESO DE OPERACIÓN	10 DIAS	ALTA
PROCESO DE MANTENIMIENTO	10 DIAS	ALTA
SISTEMA DEL SIMULADOR	10 DIAS	ALTA
BUSCAR FALLAS	10 DIAS	ALTA

Fig. 3. Tiempos de Implementación Historias de Usuario

**Análisis de requerimientos.** Se desea implementar en EMAVI un aplicativo multimedial para proceso de operación y mantenimiento del simulador de vuelo del equipo t-90, el cual debe dar las instrucciones precisas de cómo realizar el mantenimiento en un ambiente agradable, además de esto debe también indicar los pasos a seguir o procedimientos para la correcta operación del simulador de vuelo del T-90 y poder indicarle al usuario como solucionar un posible error.

**Requisitos Funcionales.** Los requisitos funcionales son aquellas funciones que el sistema debe estar en capacidad de realizar, estas funciones se logran mediante la interacción de un usuario y el sistema, y se listan a continuación:

**RF01.** El sistema permitirá revisar todos los procesos para la operación del simulador.

**RF02.** El sistema permitirá revisar todos los procesos respectivos a cómo realizar el mantenimiento.

**RF03.** El sistema permitirá revisar todos los sistemas que tiene el simulador de vuelo.

**RF04.** El sistema de acuerdo a ciertos parámetros de entrada encontrará soluciones a determinados tipos de problemas o fallas.

**Requisitos no funcionales.** Los requisitos no funcionales son aquellas acciones que no van asociadas directamente a las funciones del sistema, pero afectan a estas.

**RNF01.** Evaluar el rendimiento del software teniendo en cuenta que el sistema en que se esté implementando cumpla con todos sus requerimientos.

**RNF02.** Realizar las diferentes actualizaciones en los firewalls periódicamente para garantizar la fidelidad de los datos.

**RNF03.** El sistema debe visualizarse y funcionar correctamente en cualquier sistema operativo.

**RNF04.** El sistema debe visualizarse y funcionar en cualquier navegador web.

**RN05.** La aplicación debe ejecutarse de una manera sencilla, amigable y práctica.

### **Fase de Diseño.**

Esta etapa estará dada en la construcción del software en la cual se definirá el proceso que va a seguir el proyecto y dentro de las actividades se encuentran los diseños simples, riesgos, funcionalidad extra, tarjetas C.R.C.

**Diseño de interfaz.** Las pantallas deben permitir una forma de interacción entre el usuario y todas las funcionalidades que ofrece el sistema, cada una de ellas debe al menos presentar una funcionalidad para que su creación esté justificada. Los elementos que se deben definir para cada pantalla son: Información a presentar o recolectar, Validaciones, Relación entre datos, Flujo de páginas. Los elementos comunes entre pantallas que se podrían definir son: Encabezado (Opcional), Menú (Opcional), Zona de Contenido, Hojas de estilo (CSS), Zona de mensajes (error, éxito).

**Diseño pantalla de inicio.** Esta interfaz es la primera que va a ver el usuario, allí se van a encontrar la opción para continuar. Véase figura 4.





Fig. 4. Diseño de la Pantalla Principal.

**Diseño menú principal.** En esta fase se crea el menú principal en el cual el usuario va a poder elegir las opciones que desea realizar. Véase **¡Error! No se encuentra el origen de la referencia.** figura 5.



Fig. 5. Interfaz menú principal fase 1.

**Proceso de Operación Menú Principal.** Al usuario poner el cursor sobre la opción proceso de operación esta mostrara el hipervínculo el cual lo llevara a toda la información con respecto al proceso de encendido del simulador FTD. Véase figura 6. Igualmente se realizaron los diseños de Proceso de Mantenimiento, Sistemas del Simulador, figura 7 y Buscar Fallas.



Fig. 6. Proceso de Operación Menú Principal.



Fig. 7. Sistemas del Simulador.

**Diseño arquitectónico.** Involucra definir la estructura y las responsabilidades de los componentes que comprenderán la Arquitectura de Software. Los grandes sistemas siempre se descomponen en subsistemas que proporcionan conjuntos de servicios relacionados. El proceso de diseño inicial que identifica estos subsistemas y establece como se lleva a cabo su control y comunicación se llama diseño arquitectónico. La estructuración u organización se basa en la identificación de subsistemas o capas clave a desarrollar de forma independiente y en las relaciones entre subsistemas. Resulta efectivo para la comunicación entre los participantes en el proyecto y para realizar el reparto de tareas entre distintos grupos o recursos.

**Análisis de riesgo.** El riesgo siempre implica: Incertidumbre: el acontecimiento que caracteriza al riesgo puede o no puede ocurrir. Pérdida potencial: si el riesgo se convierte en una realidad, ocurrirán consecuencias no deseadas o pérdidas. El riesgo en sí mismo no es malo el riesgo es esencial para el progreso, y el fracaso es a menudo una parte fundamental del aprendizaje. Pero hay que aprender a equilibrar las posibles consecuencias negativas de riesgo frente a los beneficios potenciales de su oportunidad asociado. Ver figura 8.

TIPO DE RIESGO	IMPACTO	PLAN DE CONTINGENCIA
Cambio de los requerimientos del cliente	Alto	De acuerdo a la metodología XP, se adecua a los cambios que se presenten durante el desarrollo.
Falta de experiencia en herramientas usadas	Medio	Tutorías virtuales
Perdida de información por daño en hardware	Alto	Documentos y desarrollo en la nube
Falta de experiencia en metodología de desarrollo de software usado	Bajo	Asesoramiento virtual y aprendizaje autodidacta

**Fig. 8.** Matriz de Riesgos.

**Tarjetas CRC.** Las tarjetas CRC permiten al programador centrarse y apreciar el desarrollo orientado a objetos olvidándose de los malos hábitos de la programación procedural clásica. La tarjeta CRC Ingresar al sistema, permite que el programador tenga en cuenta que existen unos privilegios que tiene definido usuario al ingresar al sistema. Véase figura 9. Igualmente, se realizaron las tarjetas CRC Ingresar a proceso de Operación, Ingresar a Proceso de Mantenimiento, Sistemas del Simulador, Buscar Fallas.

CLASE: Ingresar al sistema	
RESPONSABILIDADES	COLABORADORES
Ingresar a los botones administrador o invitado	Administrador/Invitado
Solicitar login y passwd para el administrador	
Validar que la contraseña corresponda al administrador	
Devolver "error" en caso de que no coincidan login y password	
Enviar a ventana con opciones	

**Fig. 9.** Tarjeta CRC – Ingresar al Sistema.

### Fase de Codificación.

**Programación en pareja.** La programación en pareja es la etapa en la cual se buscará proporcionar el módulo más adecuado para obtener software de alta calidad. En esta

etapa se llevarán a cabo reuniones de los desarrolladores del proyecto con el fin del establecer en común acuerdo las necesidades reales del cliente y de poder obtener los requerimientos funcionales y no funcionales del sistema del proyecto, pactando aspectos importantes que deberá suplir el aplicativo software.

**Componentes desarrollados.** Con el diseño terminado, se procede a codificar la aplicación con respecto a las indicaciones de nuestro cliente (historias de usuario). El menú es el código más importante ya que de este se desprende toda la aplicación.

**Barra de Menú.** Este componente desarrollado como widgets de manera horizontal permite al usuario acceder a todas las opciones ofrecidas en la aplicación, véase **¡Error! No se encuentra el origen de la referencia..** En el código se evidencia las medidas de las opciones del menú, colores y la barra vertical emergente del submenú al rollover con el mouse. Véase **¡Error! No se encuentra el origen de la referencia..**



Fig. 10. Barra de Menú.



Fig. 11. Submenú Desplegable.

### Fase de pruebas de Aceptación.

Se le realizan pruebas al código de la aplicación las cuales tienen como objetivo proporcionar información objetiva e independiente sobre la calidad del producto y su buen funcionamiento, esto permite minimizar los errores y optimizar el funcionamiento de esta. A su vez se le hace entrega de la aplicación a los usuarios del simulador, los cuales dan fe que la aplicación cumple con todos los requisitos planteados en la fase de planificación.

**Pruebas unitarias.** Es una forma de comprobar el correcto funcionamiento de un módulo de código. Esto sirve para asegurar que cada uno de los módulos funcione correctamente por separado.

**Prueba botón menú.** Botón horizontal con barra desplegable, con cambios de aspecto y color en estado normal, con el ratón pulsado o activo; se comprueba la aparición de la barra desplegable como submenú.

**Prueba submenú desplegable.** Submenú desplegable de manera vertical con cambios de aspecto y color en estado normal, con el rato pulsado o activo; se comprueba enlace de hipervínculo a la página correspondiente según elección del usuario.

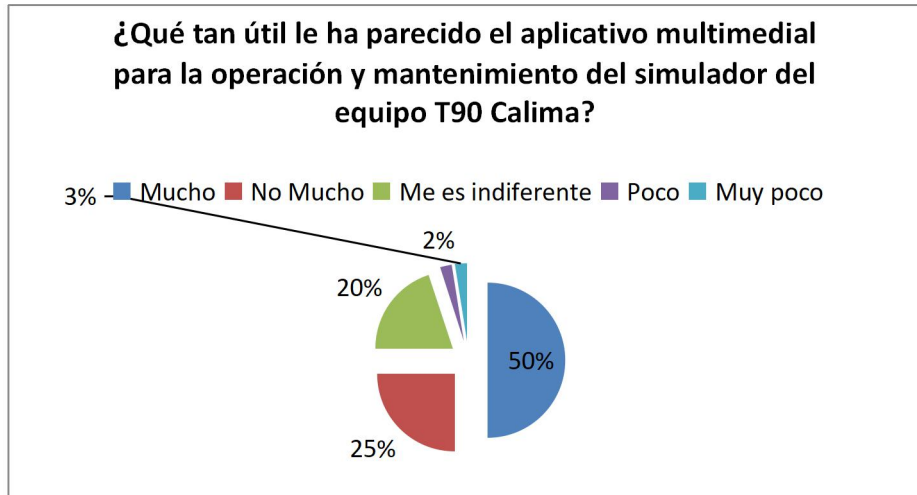
**Pruebas de integración.** Después de realizadas las pruebas unitarias, se realizan las pruebas de integración, estas con el fin de verificar que el conjunto de partes de software funciona.

**Prueba de enlace de páginas.** Prueba de enlace con la página deseada desde el menú “proceso de operación” seleccionando en el submenú “controles de vuelo”.

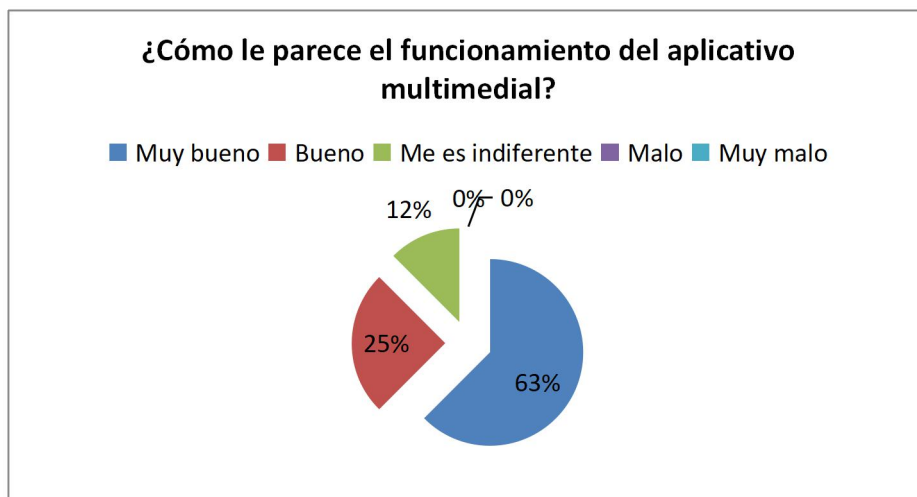
**Prueba de dirección del botón inicio al menú.** El botón de “home” permite al usuario a regresar al menú principal, a través de un hipervínculo a la página del menú.

## 6. Resultados Obtenidos

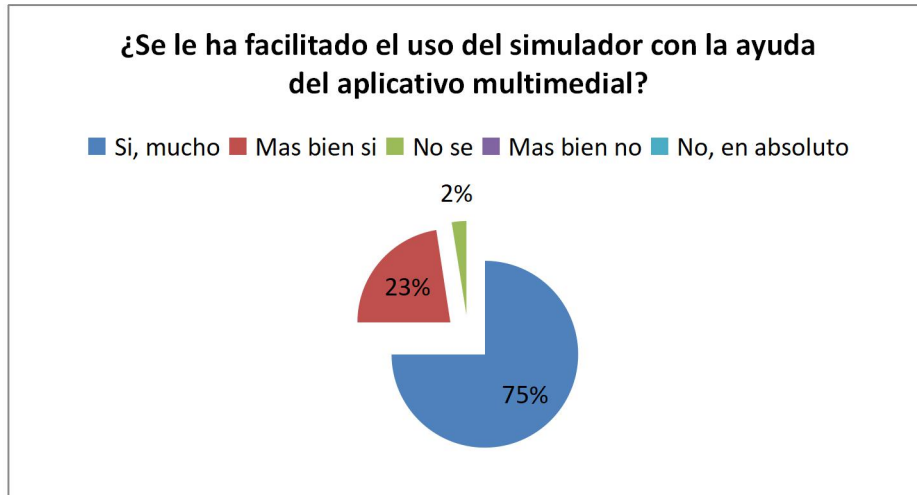
El primer objetivo se cumple en la etapa de planificación donde por medio de diferentes métodos de recolección de información como la lectura de artículos científicos elaborados entorno al objeto de estudio, la entrevista al operario del equipo simulador, donde se obtiene todo lo referente al mantenimiento del simulador, así mismo a partir de las historias de usuario se identificaron los requisitos dados por el cliente. El segundo objetivo se cumple en la etapa de diseño y codificación, donde se construyó un glosario para ayudar a los programadores y al cliente a comprender mejor la aplicación y la información consignada en ella, a su vez se analizaron los riesgos que se pueden presentar durante el desarrollo del aplicativo. Las tarjetas CRC les permitieron a los programadores centrarse y apreciar el desarrollo olvidándose de los malos hábitos de la programación, codificación todo el aplicativo según los requisitos exigidos por el cliente. El tercer objetivo se cumplió en la fase de pruebas, donde por medio de encuestas se pudo analizar el nivel de satisfacción de los cuarenta (40) alumnos de vuelo del primer y segundo turno del año 2016 de la aeronave T-90 CALIMA, beneficiarios del simulador. Estos resultados representados en las gráficas de encuestas: figura 12. a figura16.



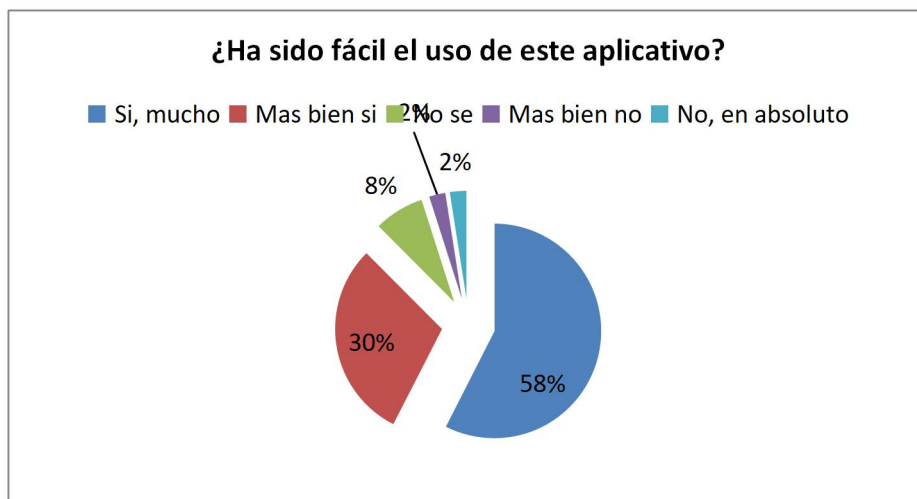
**Fig. 12.** Pregunta 1. Encuesta de Satisfacción.



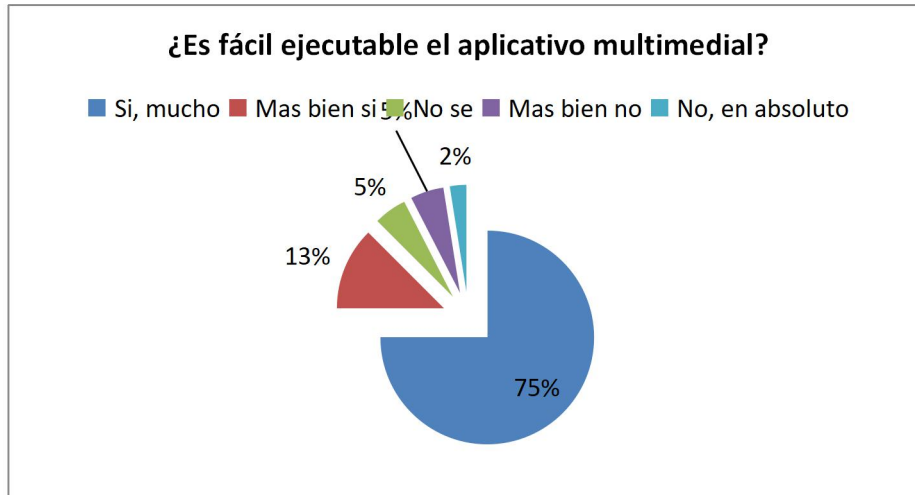
**Fig. 13.** Pregunta 2. Encuesta de Satisfacción.



**Fig. 14.** Pregunta 3. Encuesta de Satisfacción.



**Fig. 15.** Pregunta 4. Encuesta de Satisfacción.



**Fig. 16.** Pregunta 5. Encuesta de Satisfacción.

## 7 Conclusiones

- El aplicativo multimedial para el proceso de operación y mantenimiento del simulador del equipo T90 Calima, elaborado utilizando la metodología de programación extrema y el programa Adobe Muse, brinda al usuario la posibilidad de aprender a operar y desarrollar un mantenimiento básico de este simulador bajo un ambiente multimedia en lenguaje HTML5.
- La programación extrema utilizada como metodología para el desarrollo de software se consideró propicia ya que es un método ágil, ampliamente utilizado con un ciclo fácil de comprender y basado en la simplicidad, la comunicación fluida, la retroalimentación continua y el coraje para enfrentar los cambios que se presenten durante el desarrollo.
- El aplicativo desarrollado, satisface la necesidad del grupo de educación aeronáutica, quienes utilizan este simulador para el entrenamiento de sus alumnos de vuelo y para el grupo técnico los encargados del mantenimiento de este simulador, los cuales podrán capacitarse de una manera rápida y solucionar cualquier daño en el menor tiempo posible para no afectar la operatividad del mismo.
- El aplicativo multimedial cumple los requisitos exigidos por el cliente y por sus usuarios, así mismo es muy fácil de usar y bastante amigable.
- En la implementación se entregó la aplicación terminada al T1 Figueroa, suboficial encargado del simulador, se le capacitó junto con los alumnos de vuelo, los cuales manifestaron que es una aplicación fácil de usar, no se necesitan conocimientos previos para su utilización, además mejorará la operación del simulador ya que no necesitarán de un técnico especializado para corregir una falla, esto garantiza operatividad en todo momento.



- Esta aplicación fue diseñada en un entorno web con multimedia lo cual permite que el usuario pueda acceder una manera sencilla sin importar el lugar en que se encuentre. y comprender las instrucciones para la tarea que desee realizar.
- En una encuesta realizada a 40 usuarios de la aplicación los cuales son actualmente alumnos de vuelo del equipo T-90, el 75% de los encuestados les pareció que el aplicativo es muy útil a la hora de operar o realizar algún tipo de mantenimiento en el simulador.
- Con la colaboración del T1 Figueroa se logró recolectar la información respectiva de la operación y mantenimiento del simulador del equipo T-90, a partir de fabricantes de simuladores activos y existentes actualmente en unidades aéreas en Colombia, está siendo esencial para la realización de este proyecto.

## 8 Referencias

### Agradecimientos

Los autores desean expresar su agradecimiento a la Escuela Militar de Aviación “Marco Fidel Suarez” de la ciudad de Cali por el apoyo en la realización de la presente investigación.

### Referencias

1. Adobe Muse CC. (2016) Adobe Systems incorporated. Recuperado de <http://www.adobe.com/la/products/muse.html>
2. Consuelo Belloch. (2016) Diseño y Desarrollo de aplicaciones multimedia educativas En <http://www.uv.es>: <http://www.uv.es/bellohc/pwedu6.htm>.
3. Fernando. Martínez Alonso, Loïc Martínez y Francisco Segovia. (2010). La ingeniería de software. Introducción a la ingeniería de software (75). Madrid: Barbero.
4. Fritz Bauer y Roger Pressman. (2010). Un enfoque práctico. En Ingeniería del Software (7ed p.23). México: PESSMAN
5. Fuerza Aérea Colombiana. (2016). Misión de la Fuerza Aérea Colombiana. 12/05/16, Recuperado de <https://www.fac.mil.co/misi%C3%B3n-28>.
6. Ian Somerville. (2011) Desarrollo rápido de software. Ingeniería de Software. (9 ed p. 361). Madrid: Pearson Educación. A.S.
7. Universidad Nacional De Ingeniería. Introducción A La Ingeniería De Software. (2013) Facultad de Ciencias Y Sistemas. Lima Perú.
8. José H. Canos, Patricio Letelier; y M<sup>a</sup> Carmen Penades. (2003) Metodologías ágiles en el desarrollo de software. (p 3-6) Valencia. Universidad Politécnica de Valencia.
9. Juan Gauchat. El gran libro de HTML5, CSS3 Y Javascript. Marcombo S.A. Barcelona España.
10. Martín Gutiérrez (1996), Nuevas Tecnología. Educación Multimedia Y Nuevas Tecnologías. (p. 24) Madrid España. Ediciones De La Torre.
11. U.S. Department of Transportation. Federal Aviation Administration. Airplane Simulator Qualification. (1991. P. 3-4) Ac 120-40b. Washington, D.C.

12. Yazmin González. Multimedia En La Educación, Una Necesidad. Universidad Autónoma Del Estado De Hidalgo (P 1-3)