

**Universidad de las Ciencias Informáticas
Facultad 8**



Proceso de prueba aplicado a la Multimedia FEU

Trabajo de Diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor(a):

Yennia Rivero Aguila

Tutor(a):

Ing. Giselle Medina Martínez

Ciudad de la Habana, Julio 2007

Año 49 de la Revolución



*“La responsabilidad nuestra es luchar porque la calidad del producto que aquí se
haga sea de las mejores y la mejor posible...”*

Che

DECLARACIÓN DE AUTORÍA

Declaro ser autora de la presente tesis y reconocemos a la Universidad de las Ciencias Informáticas los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste firmo la presente a los 4 días del mes de Julio del año 2007.

DATOS DE CONTACTO

Breve Currículo:

- Graduada de Ingeniera Informática en la CUJAE, en el 2005.
- Se encuentra vinculada a la docencia de la Universidad de la Ciencias Informáticas desde el 2004.
- Profesora de la asignatura de Bases de Datos en la Facultad 8.
- Líder del Grupo de Calidad de Software de la Facultad 8.
- Ha sido miembro de tribunales de la esfera de Calidad de Software a nivel UCI.
- Actualmente se encuentra cursando su período de adiestramiento.

Ubicación: UCI, La Habana, Cuba

e-mail: gisellem@uci.cu

AGRADECIMIENTOS

Al grupo 8503 de la Universidad de las Ciencias Informáticas por su colaboración y paciencia durante los cinco años de la carrera.

Al colectivo de profesores y directivos de la Facultad 8 de la Universidad de las Ciencias Informáticas, eslabón principal para la elaboración del presente trabajo.

Al decano de la Facultad 1 de la Universidad de las Ciencias Informáticas Francisco Javier Hernandez Cao y demás directivos por la amistad y los conocimientos impartidos.

A mis amigas Yanet, Blanca, Yusnay, Yinimary, Isis, Yilennis, Telma, Adis y demás compañeras de cuarto por la paciencia y comprensión durante estos años de estudio.

A la Ing. Giselle Medina Martínez, Tutora de la tesis por se colaboración en la elaboración del presente trabajo.

A la Lic. Isabel Lombillo Mora por sus aportes metodológicos.

A Yunaika, Julio, Delia y demás familiares por su ayuda y hospitalidad en estos años de estudio.

A todas aquellas instituciones y personas que de una forma u otra han colaborado en la ejecución de la tesis y se hace imposibles mencionar aquí.

A todos muchas gracias.

DEDICATORIA

A mis padres Adrián Rivero Gómez y Marisela Aguila Pérez quiénes me han acercado al lugar en el que quiero estar mañana para que mis sueños se hagan realidad.

A mis hermanas Yailin y Yaima Rivero que me han brindado el apoyo y su amor incondicional en estos cinco años de estudio y sacrificio.

A mis tíos, abuelos y demás familiares que siempre será el motor impulsor de hechos como este.

RESUMEN

Las pruebas del software deben incluirse a lo largo de todo el ciclo de vida del desarrollo de un software pasando por requerimientos, análisis y diseño, programación, puesta en marcha y mantenimiento. Es por eso que las empresas cubanas han tomado conciencia en cuanto a la calidad en productos de software con el fin de incluirse en el mercado mundial.

El presente trabajo se enfoca en las pruebas de software como un proceso de vital importancia para asegurar la calidad del mismo. Con el objetivo de lograr una mejora de la calidad en la multimedia Federación Estudiantil Universitaria “FEU”, se elaboró un proceso de pruebas especificando las actividades a seguir para la aplicación de las pruebas. Con esto, se logró detectar en tiempo las fallas cometidas por los desarrolladores para su posterior corrección.

Para dar cumplimiento a las actividades planificadas en el proceso se trazó una estrategia de prueba especificando los tipos de pruebas a aplicar y diseñando casos de pruebas para ejecutar las mismas. Como resultados se obtuvo una lista con los errores y dificultades detectadas durante el desarrollo del proceso propuesto.

PALABRAS CLAVE

Pruebas del Software, Proceso de Prueba, Calidad, Requerimientos del Software.

TABLA DE CONTENIDOS

AGRADECIMIENTOS	I
DEDICATORIA	II
RESUMEN	III
INTRODUCCIÓN	1
CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA	3
1.1 Las pruebas en la Calidad del Software	3
1.2 Fundamento de la prueba del Software	5
1.2.1 Objetivos de las pruebas	6
1.2.2 Principios Básicos de las Pruebas de Software	6
1.2.3 Facilidad de prueba	8
1.3 Proceso de Prueba del software	8
1.3.1 Estrategia de Prueba	9
1.4 Plan de Pruebas	10
1.4.1 Estructura de un Plan de Prueba	11
1.5 Niveles de Pruebas	14
1.5.1 Pruebas de Unidad.....	14
1.5.2 Pruebas de Integración.....	14
1.5.3 Pruebas de Sistema	15
1.5.4 Prueba de Aceptación.....	17
1.5.5 Otros tipos de pruebas	18
1.6 Diseño de Casos de Pruebas	19
1.6.1 Método de Caja Blanca	19
1.6.2 Método de Caja Negra.....	20
1.7 Estado actual de las pruebas en productos Multimedia	24
1.8 Conclusiones	26
CAPÍTULO 2: PROCESO DE PRUEBAS DEL SOFTWARE	27
2.1 Proceso de prueba propuesto para la Multimedia FEU	28
2.2 Descripción del Software	29
2.3 Plan de Prueba General	30
2.3.2 Descripción de Requerimientos a probar	32

2.3.2.1	Requerimientos Funcionales	32
2.3.2.2	Requerimientos no Funcionales a probar	35
2.3.3	Definición de la Estrategia de Pruebas.	36
2.3.3.1	Pruebas de Interfaz	37
2.3.3.2	Pruebas Funcionales	37
2.3.3.3	Pruebas de Solidez	38
2.3.3.4	Pruebas de Soportabilidad.....	38
2.3.3.5	Pruebas de Resistencia (Stress)	39
2.3.4	Recursos Requeridos	39
2.3.4.1	Roles	39
2.3.4.2	Recursos.....	40
2.3.5	Actividades y tareas del proceso de prueba	40
2.3.6	Definición de los Entregables	41
2.3.7	Plazos y Calendarios	41
2.4	Procedimiento y Casos de Pruebas a Nivel de Unidad	42
2.4.1	Pruebas de Interfaz	42
2.4.1.1	Casos de Pruebas de Interfaz	42
2.5	Procedimiento y Casos de Pruebas a Nivel de Sistema	43
2.5.1	Pruebas Funcionales	43
2.5.1.1	Casos de Pruebas a los Requerimientos Generales	43
2.5.1.2	Casos de Pruebas por tema de contenido.....	54
2.5.3	Pruebas de Solidez	68
2.5.3.1	Casos de Pruebas de Solidez	68
2.5.4	Pruebas de Soportabilidad	69
2.5.4.1	Caso de prueba de Soportabilidad	69
2.5.5	Pruebas de Resistencia (Stress)	71
2.5.5.1	Caso de prueba de Resistencia	71
2.6	Conclusiones	71
CAPÍTULO 3: RESULTADOS DE LAS PRUEBAS		73
3.1	Estructura de la Bitácora de prueba	73
3.1.1	Registro de defectos y dificultades detectado a Nivel de Unidad	74
3.1.2	Registro de defectos y dificultades detectado a Nivel de Sistema.....	75
3.2	Resultados Generales de las pruebas	83
3.3	Costos de las pruebas	85
3.4	Conclusiones	87

CONCLUSIONES	88
RECOMENDACIONES	88
REFERENCIAS BIBLIOGRAFICAS	89
BIBLIOGRAFIA	91
GLOSARIO	93
ANEXOS	94

INTRODUCCIÓN

Al analizar el mundo de la informática se observa que el proceso de desarrollo del software ha evolucionado en la actualidad. A pesar de la creciente participación y de los avances producidos en las diferentes esferas de los países existen dificultades en cuanto al proceso de evaluación de los productos de software. Las necesidades y expectativas de los clientes y usuarios no son captadas satisfactoriamente, de ahí que muchos de los proyectos no llegan a cumplir sus objetivos, y como consecuencia de esto, los altos por cientos de rechazo entre ellos.

En Cuba gran parte de los esfuerzos dedicados a la producción de software se centran en la obtención de productos con fines educativos, los que contribuyen a mejorar el proceso de aprendizaje y comunicación. Actualmente la Universidad de las Ciencias Informáticas es una de las mayores fuentes de producción de software y específicamente la facultad 8 se dedica al desarrollo de productos multimedia con fines educativos. Este proceso de desarrollo del software en ocasiones resulta muy complejo, debido a esto es necesario documentar y organizar el trabajo, para obtener un producto en menos tiempo y costo y sobre todo con calidad.

La calidad de un producto software no es un objetivo fácil de alcanzar, sin embargo puede facilitarse mediante la evaluación durante el ciclo de vida del software, de forma tal que se detecten los defectos durante el proceso de elaboración y puedan ser corregidos a tiempo. En el caso de los productos multimedia se tiene poca experiencia en la aplicación de pruebas, por lo que existen dificultades en la entrega de productos libres de defectos. Esto trae consigo resultados desfavorables, como retraso en la entrega e insatisfacción del cliente o usuario final.

Para guiar el proceso de investigación se planteó la siguiente problemática: ¿Cómo evaluar la Multimedia “FEU” de manera que logre alcanzar un alto nivel de calidad?

Para dar solución al problema planteado se trazó como objetivo general: Aplicar un Proceso de pruebas que permita detectar los errores ocurridos en el desarrollo de la Multimedia “FEU”.

Para lograr el cumplimiento del objetivo general se tuvieron en cuenta los siguientes objetivos específicos:

- Seleccionar los tipos de pruebas a aplicar en la multimedia.
- Organizar las pruebas para su ejecución en la aplicación a probar.

- Realizar un análisis de los resultados obtenidos.

Tareas desarrolladas para cumplir los objetivos:

- Estudiar las características de la Multimedia “FEU” con el fin de lograr un entendimiento de lo que se quiere probar.
- Investigar los tipos de pruebas que existen y se puedan aplicar a productos multimedia.
- Definir la estrategia de aplicación para la ejecución de las prueba.
- Diseñar los casos de pruebas para verificar el correcto funcionamiento de los requerimientos tanto funcionales como no funcionales de la multimedia.
- Obtener una lista detallada con los errores y dificultades detectadas en cada tipo de prueba para el análisis de los resultados.

EL presente trabajo está estructurado en tres capítulos:

En el primer capítulo se abordan temas como: conceptos, objetivos, principios de las pruebas de software, la relación entre la calidad de un software y las pruebas, los tipos de pruebas y la estrategia a seguir para aplicarlas. Además se tratan los componentes necesarios para que un Plan de Prueba sea exitoso.

El segundo capítulo está dedicado a describir el proceso de prueba propuesto, la descripción del producto a probar, la elaboración del Plan de pruebas de la multimedia “FEU” y el diseño y procedimiento de los casos de pruebas que serán ejecutados.

El tercer capítulo contiene los resultados que se obtuvieron al aplicar las pruebas: una lista con los defectos y dificultades detectadas, análisis de los resultados esperados y los gastos monetarios que produjo aplicar el proceso de pruebas.

CAPÍTULO 1: FUNDAMENTACIÓN TEÓRICA

Dada la importancia que el software ha adquirido en la actualidad, se ha convertido en una actividad programada, planificada y con un ciclo de vida definido. En muchas organizaciones el proceso de pruebas del software supone de un 30% a un 50% de los costes totales dedicados al desarrollo del producto. Pese a su enorme impacto en el costo y tiempo de desarrollo, es una fase que muchos de los involucrados aún no consideran. Hoy en día muchas personas opinan que el software no está bien probado antes de ser distribuido al cliente. Esta contradicción tiene su raíz en dos hechos claros: [EQUIZABAL, 2006]

- Probar software es un proceso dificultoso debido a la infinidad de comportamientos que éste puede presentar.
- Las pruebas se ejecutan sin seguir una planificación.

Este planteamiento enfatiza en dos puntos: un proceso de prueba totalmente completo es algo impensable por la cantidad de camino o acciones que puede presentar un software y segundo, las pruebas deben planificarse con el objetivo de definir actividades y responsabilidades en cada una de las fases en las que está dividido el software.

Uno de los aspectos más descuidados dentro de un proceso de desarrollo es el de las pruebas, a pesar de ser el medio por el cual se puede asegurar la calidad del producto de software, de ahí la importancia de estas para lograr que los requerimientos de los usuarios sean cumplidos correctamente.

1.1 Las pruebas en la Calidad del Software

El concepto de calidad ha estado en boga en los últimos años en diferentes áreas de la ingeniería. A pesar de esto existen aún organizaciones que no están conscientes de la importancia de la calidad, lo que implica calidad o cómo se llega a la calidad correcta de un servicio. La calidad es un factor clave en toda organización cuando se pretende avanzar hacia metas específicas orientadas a la permanencia y crecimiento de la misma dentro de la economía y la globalización actual.

Existen varios conceptos de calidad pero todos tienen como enfoque la competitividad que alcanza la organización al ofrecer productos o servicios de calidad y hacen siempre énfasis en satisfacer las necesidades de los mercados y clientes.

Según Pressman define la calidad como: [PRESSMAN, 2000]

“Concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente”.

De lo anterior se puede decir que:

- Los requerimientos son los fundamentos desde los que se mide la calidad.
- Los estándares definen un conjunto de criterios de desarrollo.
- Hay requerimientos implícitos que no se mencionan como, por ejemplo, el deseo de que un programa sea fácil de mantener y de depurar.

Según Crosby, “calidad es cumplir con los requerimientos del usuario” [CROSBY, 1979]. Esta idea llevada al contexto de sistemas de información, no es más que la medida en la que se cubran los requerimientos de los usuarios. Al respecto se puede decir que el hecho de encontrar los verdaderos requerimientos de los usuarios es un proceso complicado y es una labor que involucra mucho esfuerzo y dedicación, tanto por parte del personal, como de los usuarios mismos.

En todo proceso de software, la garantía de la calidad es una actividad de protección que se aplica cubriendo aspectos tales como: [QUESADA, 2005]

- Aplicación de metodologías técnicas
- Realización de revisiones técnicas formales.
- **Prueba del software.**
- Ajuste a los estándares.
- Control de cambios.
- Mediciones.
- Registro y realización de informes
- Auditorias

Para determinar dicho nivel de calidad se deben efectuar unas medidas o pruebas que permitan comprobar el grado de cumplimiento respecto a las especificaciones iniciales del sistema.

1.2 Fundamento de la prueba del Software.

Las pruebas son una parte muy significativa dentro de un proyecto, no sólo por su importancia en el logro de resultados correctos sino por el tiempo y recursos requeridos. El diccionario del *Institute of Engenier Electrical and Electronic* (IEEE) define a las pruebas de la siguiente manera:

“Es una actividad en la cual un sistema o uno de sus componentes se ejecuta en dos o más circunstancias previamente especificadas, los resultados se observan y registran y se realiza una evaluación de algún aspecto” [IEEE, 1990]”.

Teniendo en cuenta la definición anterior se concluye que las pruebas del software es el conjunto de técnicas con el fin de encontrar errores o fallas en un programa que permiten determinar la calidad de un producto software.

Algunas características comunes a todas las pruebas software se recogen en la tabla 1 [GUTIERREZ, 2005].

Tabla 1. Características comunes a las pruebas

- | |
|--|
| <ol style="list-style-type: none">1. Para probar necesitamos código que se pueda ejecutar.2. Para probar necesitamos saber cual es el resultado esperado.3. Una prueba es mejor que otra cuando encuentra más errores.4. Es imposible probar una aplicación al 100%.5. Las pruebas no deben dejarse para el final. |
|--|

1.2.1 Objetivos de las pruebas

En un excelente libro sobre las pruebas del software establece varias normas que pueden servir acertadamente como objetivos de las pruebas: [GLEN MYERS, 1979]

1. La prueba es un proceso de ejecución de un programa con la intención de descubrir errores.
2. Un buen caso de uso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
3. Una prueba tiene éxito si descubre un error no detectado hasta entonces

Por tanto un efecto importante de las pruebas es que da una mejor comprensión de cómo se pueden prevenir los defectos que no han sido detectados hasta el momento.

Hay una regla universal, la cual indica que es imposible encontrar todos los defectos y que nunca hay suficiente tiempo, personal o dinero para probar todo. Se deben tomar decisiones sabias de como distribuir los recursos disponibles. [GUTIERREZ, 2005]

Esto demuestra que las pruebas no aseguran que un software está 100% libre de errores pero que al aplicarlas gran parte de los errores se detecten en tiempo. Para lograr esto, se necesita de una planificación en el inicio del proyecto especificando que debe ser probado y ejecutando los casos de prueba correspondientes a el nivel de prueba en que se encuentran.

1.2.2 Principios Básicos de las Pruebas de Software

Los principios básicos de pruebas de software a menudo parecen obvios pero por lo general son siempre ignorados, lo cual genera serias consecuencias en el proceso. Entre los principios básicos de pruebas de software tenemos los siguientes: [ESPINOZA, 2005]

- *La definición del resultado esperado a la salida del programa es una parte integrante y necesaria de un caso de prueba;* muchos errores se originan por ignorar este principio. Si el resultado esperado de un caso de prueba no ha sido predefinido, existe la posibilidad de que un resultado erróneo, se interprete como correcto.

- *Un programador debe evitar probar su propio programa;* la detección de errores es un proceso “destrutivo”. Es por ello que es difícil para una persona adoptar la actitud mental necesaria para demostrar que lo que acaba de hacer está erróneo.
- *Un área de programación no debería probar sus propios programas;* es el caso similar al anterior, aplicado a la conducta colectiva de un grupo.
- *Inspeccionar concienzudamente el resultado de cada prueba;* este es probablemente el más obvio, pero, sin embargo, a menudo dejado de lado. Empíricamente se ha encontrado que muchos de los programadores fracasan en la detección de errores, aún cuando en los resultados de las pruebas eran claramente observables.
- *Examinar un programa para comprobar que no hace lo que se supone que debe hacer es sólo la mitad del problema;* la otra mitad consiste en ver si el programa hace lo que no se supone que debe hacer. Esto significa que los programas deben ser revisados con respecto a efectos colaterales no deseados.
- *Evitar los casos de pruebas desechables a menos que el programa sea verdaderamente un programa desechable;* una práctica muy frecuente es sentarse frente al computador e inventar casos e pruebas. Los casos de prueba deben ser reinventados con cada prueba que se realice, algo que generalmente no se hace, por lo que las pruebas siguientes no son tan rigurosas como la primera vez, esto significa que si se introdujo un error en una parte ya probada, rara vez se detecta.
- *No planear el esfuerzo con la suposición tácita de que no se encontrarán errores;* este error se comete cuando se parte de la suposición de que la prueba es el proceso de mostrar que el programa funciona correctamente.
- *La probabilidad de encontrar errores adicionales en una sección del programa es proporcional al número de errores encontrados;* este fenómeno tiene su base en comprobaciones empíricas, en efecto, los módulos de un programa (igualmente probados) que han presentado más errores tienen una probabilidad más alta de presentar otros errores.
- *La prueba de software no es una ciencia exacta;* si bien existen métodos que ayudan en el proceso de la elaboración de casos de pruebas, ello requiere de todas maneras de una considerable cuota de creatividad.

1.2.3 Facilidad de prueba

De acuerdo a James Bach la facilidad de prueba es: "... simplemente lo fácil que se puede probar un programa de computadora". Como la prueba es tan profundamente difícil, merece la pena saber que se puede hacer para hacerla más sencilla. A veces los programadores están dispuestos a hacer las cosas que faciliten el proceso de prueba y una lista de comprobación de posibles puntos de diseño, características, etc., puede ser útil a la hora de negociar con ellos.

La siguiente lista de comprobación proporciona un conjunto de características que llevan a un software fácil de probar: [GUTIERREZ, 2003]

- **Operatividad.** Cuanto mejor funcione, mas eficientemente se puede probar.
- **Observabilidad.** Lo que se ve es lo que se prueba.
- **Controlabilidad.** Cuanto mejor se pueda controlar el software, mas se puede automatizar y optimizar.
- **Capacidad de descomposición.** Controlando el ámbito de las pruebas se puede aislar rápidamente los problemas y llevar a cabo mejores pruebas de regresión.
- **Simplicidad.** Cuanto menos haya que probar, se puede probar de manera más rápida.
- **Estabilidad.** Cuantos menos cambios se produzcan, menos interrupciones sufriran las pruebas.
- **Facilidad de comprensión.** Cuanta más información se tenga, más inteligentes serán las pruebas

1.3 Proceso de Prueba del software

Hoy en día, para poder competir en este mercado globalizado y altamente competido de las tecnologías de información y comunicación, la calidad no representa un valor agregado, sino más bien un prerrequisito indispensable para permanecer en el mercado. La importancia de tener un producto con calidad reside en tener un software altamente comercializado e indica que este ha sido evaluado durante todo el proceso de desarrollo.

La tendencia es iniciar las pruebas antes y dentro del proyecto, los especialistas responsables de esta actividad deben estar capacitados adecuadamente para cumplirla a cabalidad. Contar con un modelo de

ciclo de vida de desarrollo permite tener un mejor control del proceso de pruebas, dado que las actividades pueden ser planeadas y controladas consistentemente.

Un aspecto fundamental del proceso de prueba es evaluar la satisfacción de la especificación funcional del sistema o requisitos por parte del sistema construido. Para garantizar el nivel de calidad del sistema construido es necesario verificar la correcta y completa implantación de los requisitos establecidos en las etapas iniciales del desarrollo [GUTIERREZ, 2005].

Esta idea enfatiza que un proceso de prueba debe cubrir todos los requerimientos establecidos por el usuario desde la misma fase de Inicio puede hacerse parte de la planificación inicial de las pruebas cuando se define el ámbito del sistema.

Un proceso de prueba del software se define como la verificación dinámica del comportamiento del software a partir de un conjunto finito de casos de prueba [GUTIERREZ, 2005].

Esto indica que las pruebas deben comenzar en la fase misma de la definición de requerimientos y terminar con la finalización de la aplicación de los casos de prueba al sistema ejecutable. Para lograr ese objetivo se debe incorporar una serie de actividades planificadas y sistemáticas que garantizan la evaluación de los productos del software. Generalmente un proceso de pruebas del software se divide en: la planeación de las pruebas, el diseño de casos de prueba, la ejecución de pruebas y la agrupación y evaluación de los datos resultantes.

La planificación de los procesos de pruebas es una etapa de gran importancia ya que además de definir una estrategia de prueba, establece los recursos necesarios, permite realizar estimaciones que establecen una base para el control y seguimiento durante la ejecución del proyecto y además ofrece información para realizar análisis de los procesos para identificar aspectos de mejoramiento.

1.3.1 Estrategia de Prueba

Las estrategias de pruebas permiten enfocar el plan de pruebas; éste comprende la visión global del proceso de pruebas y la definición de actividades y roles involucrado en cada una de ellas.

Dentro de las características generales de la estrategia de prueba se encuentran: [PRESSMAN, 2002]

1. La prueba comienza en el nivel de módulo y trabaja "hacia fuera", hacia la Integración completa del sistema completo
2. En diferentes puntos es adecuada la utilización de técnicas de prueba distintas.
3. La prueba la lleva a cabo el que desarrolla el *software* y para grandes proyectos, un grupo de prueba independiente.
4. La prueba y la depuración son actividades diferentes, pero la depuración puede entrar en cualquier estrategia de prueba.

Al analizar lo plantado se puede concluir que las estrategias de pruebas permiten tener un control en la aplicación de las pruebas al definir según el nivel donde se encuentre diferentes técnicas de pruebas y si el producto es de gran tamaño es recomendable crear un grupo independiente para que de forma constructiva hagan explotar el *software*.

Las diferentes estrategias de pruebas del *software* proporcionan al desarrollador de *software* una plantilla para la prueba, estas plantillas tienen características generales, entre las se encuentran la inclusión de pruebas de bajo nivel que verifiquen que todos los pequeños segmentos de código fuente se han implementado correctamente, así como pruebas de alto nivel que validan las principales funciones del sistema frente a los requisitos del cliente.

La validación se refiere al conjunto de actividades que aseguran que el *software* implementa correctamente una función específica. La verificación se refiere a un conjunto diferente de actividades que aseguran que el *software* construido se ajusta a los requisitos del cliente [PRESSMAN, 2002].

Boehm define estos dos términos principales relacionados con la fase de prueba [BOEHM, 1981]:

- *Verificación*: responde a la pregunta ¿se está construyendo el producto correctamente?
- *Validación*: responde a la pregunta ¿se está construyendo el producto correcto? Es decir, ¿se ajusta a los requisitos del cliente?

1.4 Plan de Pruebas

Para la puesta en práctica de un proceso de prueba, la construcción de un Plan de prueba es un factor de éxito que permite la organización al entregar un *software* de mejor calidad. El objetivo fundamental de un plan de prueba es señalar el enfoque, los recursos y el esquema de actividades de prueba, así como los

elementos a probar, las características, las actividades de prueba, el personal responsable y los riesgos asociados.

Un plan de pruebas está constituido por un conjunto de pruebas. Cada prueba debe: [PRESSMAN, 2004]

- dejar claro qué tipo de propiedades se quieren probar.
- dejar claro cómo se mide el resultado
- especificar en qué consiste la prueba (hasta el último detalle de cómo se ejecuta)
- definir cual es el resultado que se espera (identificación, tolerancia,...) ¿Cómo se decide que el resultado es acorde con lo esperado?

1.4.1 Estructura de un Plan de Prueba

Cada esfuerzo o proceso de pruebas puede ser diferente y específico. La mayor parte de los proyectos informáticos (sean de nuevos desarrollos o de mantenimiento de aplicaciones) tienen un marco común para la realización de las pruebas. Uno de los componentes más importantes que debe de tener un plan de prueba son: [GONZALEZ, 2002]

1. **Descripción de Aspectos Generales.** Esta sección establece el alcance y el objetivo del Plan de Pruebas.

- **Objetivo** - Describe por qué el Plan de Pruebas fue desarrollado - cuales son sus objetivos. Esto puede incluir requerimientos de documentación, definición de estrategias de prueba, identificación de recursos y proyección de entregables.
- **Entorno o Marco** - Explica los eventos que dan origen al Plan de Pruebas. Esto puede incluir la realización de procesos mejorados, o la adición de nuevos ambientes, equipamientos, funcionalidades.
- **Arquitectura Técnica** - Diagramación de las partes que componen el sistema bajo prueba. Incluye el almacenamiento de datos y las conexiones para su transferencia y describe el objetivo de cada componente, inclusive la forma de su actualización. Se debe documentar tanto las capas, como la presentación / interfaz del usuario, la base de datos, los emisores de informes, etc. Un diagrama de alto nivel que muestre como el sistema en prueba se inserta en un contexto de automatización mayor también puede ser agregado, si el mismo está disponible.

- Especificaciones del Software y Hardware- Corresponde a una lista individualizada de todo el hardware y el software que utiliza la aplicación, incluyendo proveedores y versiones.
- Alcance - Describe brevemente los recursos que el plan requiere, las áreas de responsabilidad, las etapas y los riesgos potenciales.
- Información del proyecto - Identifica toda la información que está disponible en relación con el proyecto. La documentación del usuario, el plan de proyecto, las especificaciones del producto, los materiales para entrenamiento y las revisiones ejecutivas, son algunos ejemplos de información del proyecto.

2. **Descripción de Requerimientos.** Esta sección del Plan de Pruebas contiene una lista de todos los requerimientos que serán probados. Cualquier requerimiento no incluido en esta lista estará fuera del alcance de las pruebas.

- Requerimientos Funcionales - todas las funciones que deben ser probadas
- Requerimientos de Diseño - Las pruebas de la interfaz de usuario, las estructuras de menú u otros elementos de diseño deberían ser puestas en una lista o referenciados hacia otro documento.
- Otros Requerimientos - Cualesquiera otras exigencias que tenga la aplicación y que necesiten ser probadas.

3. **Definición de la Estrategia de Pruebas.** Use esta sección para describir como los objetivos de la prueba serán alcanzados para cada uno de los tipos de pruebas que hacen parte del plan: Unitarias, Funcionales, De integración, De sistema, De volumen, De estrés, De configuración y/o De instalación.

Para cada subconjunto requerido o definido como necesario, debe detallarse lo siguiente:

- Objetivo - El objetivo global de esta estrategia debe alcanzarse. Por ejemplo, para una prueba de sistema, este objetivo puede ser una declaración de que todos los requerimientos funcionales deben comportarse de acuerdo a lo esperado, o como quedó documentado.
- Técnica - Especifica como los casos de prueba serán desarrollados, el instrumento o herramienta usado para almacenarlos y donde pueden ser encontrados; como ellos serán ejecutados y los datos que serán usados.

- Consideraciones Especiales - Definir los datos necesarios u otras dependencias de la prueba; las condiciones de ambiente u otros aspectos que se requieren para establecer un estado conocido y estable para las pruebas.
- Diseño de Casos de prueba – Diseño del caso de prueba que será usado para la ejecución de las pruebas.
- Criterios de Terminación - Registrar los criterios que serán usados para determinar la aprobación o rechazo de pruebas y la acción que debe ser tomada con base en los resultados de la prueba.
- Herramientas - Documentar los instrumentos o herramientas que serán empleados para las pruebas. Citar al proveedor, la versión.

4. **Recursos Requeridos.** Identificar los roles las responsabilidades y los recursos que serán requeridas para la ejecución del Plan de Pruebas.

5. **Plan de proyecto.** Desarrollar un plan de proyecto mostrando a las fases, tareas y recursos. Mantenga siempre al día el plan de proyecto para reflejar acontecimientos tales como cambios de plazos (de fechas límites) o recursos disponibles.

6. **Definición de los Entregables.** Ponga en una lista cualquier entregable asociado con el esfuerzo de pruebas y donde las copias de estos entregables o documentos pueden ser localizados. Esto incluye el Plan de Pruebas en sí mismo, escenarios para prueba, casos de prueba, etc.

7. **Calendario y Plazos.** Documentar el plazo en el cual la aplicación a probar estará disponible para pruebas y el tiempo estimado para entregar el Plan de prueba, ejecutar los casos de prueba. Especifique si se proporcionará partes construidas, sobre una base regular durante el ciclo de prueba, o cuando se espera que los componentes del sistema estén listos para pruebas.

8. **Aprobación del Plan.** El Plan de Pruebas debe ser revisado por todas las partes responsables de su ejecución y aprobado por el equipo de prueba, el jefe del proyecto y el gerente de desarrollo. Obtenga las firmas de aprobación en todas las páginas del mismo.

1.5 Niveles de Pruebas

Durante el desarrollo de un software deben llevarse a cabo de manera gradual varios niveles de prueba al sistema (Nivel de Unidad, Nivel de Integración, Nivel de Sistema y Nivel de Aceptación). Muchos de los autores estudiados definen estos cuatro niveles de pruebas donde pueden ser aplicados según el nivel en el que se encuentre determinados tipos de prueba.

1.5.1 Pruebas de Unidad

Se focaliza en ejecutar cada módulo (o unidad mínima a ser probada) lo que provee un mejor modo de manejar la integración de las unidades en componentes mayores. Busca asegurar que el código funciona de acuerdo con las especificaciones y que el módulo lógico es válido. Estas pruebas suelen ser realizarlas por propio personal de desarrollo, pero evitando que sea el propio programador del módulo. [ESPINOZA, 2005]

Características:

- . Particionar los módulos en unidades lógicas fáciles de probar, por cada unidad hay que definir los casos de prueba.
- Los casos de prueba deben diseñarse de forma que se recorran todos los caminos de ejecución posibles dentro del código bajo prueba; por lo tanto el diseñador debe construirlos con acceso al código fuente de la unidad a probar (Pruebas de Caja Blanca). Además, deben diseñarse casos de pruebas para comprobar la interfaz del módulo a través de las pruebas de Caja Negra, y así, asegurar que la información externa fluye correctamente.
- Los aspectos a considerar son los siguientes: rutinas de excepción, rutinas de error, manejo de parámetros, validaciones, valores aceptados, valores límites, rangos, mensajes posibles.

1.5.2 Pruebas de Integración

Se focaliza en verificar que las interfaces entre los componentes de software funcionan correctamente, así como determinar el enfoque para avanzar desde un nivel de integración de los componentes al siguiente y las acciones a tomar cuando se descubren problemas [ESPINOZA, 2005]

Características:

- Identificar errores introducidos por la combinación de programas probados unitariamente.

- Verificar que las interfaces entre las entidades externas (usuarios) y las aplicaciones funcionan correctamente. Verificar que las especificaciones de diseño sean alcanzadas.
- Determina el enfoque para avanzar desde un nivel de integración de los componentes al siguiente.
- Decide qué acciones tomar cuando se descubren problemas.

Los tipos de pruebas de integración más importantes son: [RIOJA, 2005]

- **Pruebas de regresión:** Asegurar el correcto funcionamiento cuando se produce un cambio en un sistema que ya estaba funcionando
- **Integración incremental.** Al esquema global se van añadiendo los componentes de uno en uno. De ese modo, al aparecer errores lo más probable es que se deban al último módulo ensamblado. Dentro de este tipo de integración surgen varias estrategias:
 1. Ascendente. Se empieza con los módulos de nivel superior, y se verifica que los módulos de nivel superior llaman a los de nivel inferior de manera correcta, con los parámetros correctos.
 2. Descendente. Se empieza con los módulos de nivel inferior, y se verifica que los módulos de nivel inferior llaman a los de nivel superior de manera correcta, con los parámetros correctos.
- **Integración no incremental.** Se prueba cada módulo por separado y luego se integran todos de una vez y se prueba el programa completo

1.5.3 Pruebas de Sistema

El objetivo de las pruebas del sistema es comprobar la integración del sistema de información globalmente, verificando el funcionamiento correcto de las interfaces entre los distintos subsistemas que lo componen y con el resto de sistemas de información con los que se comunica. En la realización de estas pruebas es importante comprobar la cobertura de los requisitos, dado que su incumplimiento puede comprometer la aceptación del sistema por el equipo de operación responsable de realizar las pruebas de implantación del sistema, que se llevarán a cabo en el proceso Implantación y Aceptación del Sistema. [SANTOS, 2005]

Este nivel de pruebas se basan en técnicas del método de Caja Negra, esto es, verificar el sistema (y sus procesos internos), la interacción con las aplicaciones que lo usan vía GUI y analizar las salidas o resultados. Esencialmente, el encargado de la prueba intenta hacer fallar el programa.

Técnica: [ESPINOZA, 2005]

Ejecutar cada caso de prueba, flujo básico o función utilizando datos válidos e inválidos, para verificar que:

1. Los resultados esperados ocurren cuando se utiliza un dato válido.
2. Los mensajes de error o de advertencia aparecen en el momento adecuado, cuando se utiliza un dato inválido.
3. Cada regla de negocios es aplicada adecuadamente.

Tipos de pruebas de sistema:

- **Pruebas Funcionales:** Se enfoca a validar funcionalidades específicas provistas por servicios requeridos, métodos, o casos de uso. Estas pruebas se implementan y ejecutan a nivel de unidades, unidades integradas, aplicaciones y sistemas. [SORIANO, 2005]
- **Pruebas de Configuración:** Se enfocan en evaluar aquellos elementos configurados para diferente hardware y/o configuraciones de software. Pueden implementarse como pruebas de rendimiento del sistema. Es un tipo de prueba para soportabilidad. [SORIANO, 2005]
- **Pruebas de Instalación:** Se enfoca en evaluar que el elemento a probar se instala como se indica, en diferentes hardware y /o configuraciones de sistemas de software y bajo diferentes condiciones (tales como espacio insuficiente en disco, interrupción de electricidad). Este tipo de prueba se aplica y ejecuta sobre aplicaciones y sistemas. Es un tipo de prueba para soportabilidad
- **Pruebas de Resistencia (Stress):** Este tipo de prueba se enfoca a evaluar el comportamiento del sistema baso condiciones anormales. Stress del sistema se refiere a extrema carga, memoria insuficiente, no disponibilidad de servicios y hardware o recursos compartidos limitados. Este tipo de prueba permite comprender mejor cómo y qué áreas del sistema colapsarán, de este modo es posible planificar contingencias y actualizar el mantenimiento y planear y asignar recursos de antemano. Esto se logra a través de: [ESPINOZA, 2005]
 1. Casos de prueba que requieran el máximo de memoria o de otros recursos.
 2. Casos de prueba que puedan dar problemas con el esquema de gestión de memoria virtual.
 3. Casos de prueba que produzcan excesivas búsquedas de datos residentes en disco.

- **Prueba de recuperación:** Es una prueba del sistema que obligue al fallo del software de muchas formas y verifica que la recuperación se lleva a cabo apropiadamente. Si la recuperación es automática hay que evaluar la corrección de la reinicialización, de los mecanismos de recuperación de datos y del arranque. Si la recuperación requiere la intervención humana, hay que evaluar los tiempos medios de reparación para determinar si están dentro de los límites aceptables. [PRESSMAN. 2002]
- **Prueba de seguridad:** Intenta verificar que los mecanismos de protección incorporados en el sistema lo protegerán de accesos inapropiados.
- **Prueba de rendimiento.** Está diseñada para probar el rendimiento del software en tiempo de ejecución dentro del contexto de un sistema integrado. Esta prueba se da durante todos los pasos del proceso de prueba, incluso al nivel de unidad, se debe asegurar el rendimiento de los módulos individuales a medida que se realizan las pruebas de Caja Blanca. [ESPINOZA, 2005]
- **Pruebas de usabilidad:** Estas pruebas requieren que se pruebe requisitos de usabilidad, facilidad de aprendizaje y operabilidad. Se centran en recursos humanos, estética, consistencia de la interfaz de usuario, documentación para el usuario y materiales de entrenamiento.[SORIANO, 2005]

1.5.4 Prueba de Aceptación

La prueba de aceptación es ejecutada antes de que la aplicación sea instalada dentro de un ambiente de producción. La prueba de aceptación es generalmente desarrollada y ejecutada por el cliente o un especialista de la aplicación y es conducida a determinar como el sistema satisface sus criterios de aceptación validando los requisitos que han sido levantados para el desarrollo, incluyendo la documentación y procesos de negocio. Basado en esta prueba el cliente determina si acepta o rechaza el sistema. [RIOJA, 2005]

Estas pruebas están destinadas a probar que el producto está listo para el uso operativo, suelen ser un subconjunto de las Pruebas de Sistema y sirven para que el usuario pueda validar si el producto final se ajusta a los requisitos fijados, es decir, si el producto está listo para ser implantado para el uso operativo en el entorno del usuario.

Técnica: [ESPINOZA, 2005]

1. Realización de los documentos de planes de prueba de aceptación y especificación de los mismos, basados en los criterios de aceptación del cliente.
2. Los casos prueba de aceptación han de ser planificados, organizados y formalizados de manera que se determine el cumplimiento de los requisitos del sistema. Para la realización de estas pruebas se necesita disponer de los siguientes documentos:
 - Especificación de requisitos del sistema.
 - Manual de usuario.

Tipos de pruebas de Aceptación: [PRESSMAN, 2004]

- **Pruebas alfa:** Consisten en invitar al cliente a que venga al entorno de desarrollo a probar el sistema. Se trabaja en un entorno controlado y el cliente siempre tiene un experto a mano para ayudarle a usar el sistema y para analizar los resultados.
- **Pruebas beta:** Vienen después de las pruebas alfa, y se desarrollan en el entorno del cliente, un entorno que está fuera de control. Aquí el cliente se queda a solas con el producto y trata de encontrarle fallos (reales o imaginarios) de los que informa al desarrollador.

1.5.5 Otros tipos de pruebas

Prueba de solidez: Verifican la correspondencia entre el tema tratado, el texto y el resto de los medios que aparecen en cada pantalla contra el guión de contenidos o medias, la revisión ortográfica de los textos de la multimedia, calidad de los medios que se muestran (sonido, imágenes, videos, incluyendo diferentes resoluciones de pantalla) y el cumplimiento de las pautas trazadas por diseño. [SACHA, 2006]

Prueba de Interfaz de Usuario: Consiste en probar la interfaz de usuario para garantizar que cumple los estándares y requerimientos definidos. Usualmente se refiere a la prueba de interfaz de usuario gráfica. [MENDEZ, 2004]

1.6 Diseño de Casos de Pruebas

El principal objetivo del diseño de casos de prueba es obtener un conjunto de pruebas que tengan la mayor probabilidad de descubrir los defectos del software.

Se puede “referir a la documentación en la que se describen las entradas, condiciones y salidas de un caso de prueba” [IEEE, 1990]

Por tanto un caso de prueba es un conjunto de entradas, condiciones de ejecución y resultados esperados desarrollados para un objetivo particular como, por ejemplo, ejercitar un flujo de un programa o verificar el cumplimiento de un determinado requisito.

Para llevar a cabo este objetivo, se usan dos métodos de diseño de casos de prueba:

- Prueba de Caja Blanca
- Prueba de Caja Negra.

Las pruebas de caja blanca denominada también prueba de caja de cristal es un método de diseño de casos de prueba que utiliza la estructura de control del diseño procedimental para obtener los casos de prueba. Las pruebas de caja negra denominada también prueba de comportamiento o prueba de partición está centrada en los requisitos funcionales del software. [PRESSMAN, 2004]

Los métodos de diseño de casos de prueba ayudan a definir conjuntos de casos de prueba aplicando un cierto criterio, los casos de prueba quedarán determinados por los valores a asignar a las entradas en su ejecución. Los casos de pruebas deben verificar si el producto satisface los requerimientos del usuario, tal y como se describe en las especificación de los requerimientos y si se comporta como se desea, tal y como se describe en las especificaciones funcionales del diseño.

1.6.1 Método de Caja Blanca

Las pruebas de caja blanca tienen como objetivo recorrer la estructura del código comprobando la ejecución de todos los posibles caminos. Cuando se pasan casos de prueba al programa que se está

probando, es conveniente conocer qué porcentaje del programa se ha ejecutado, de manera que estemos próximos a asegurar que todo él es correcto (evidentemente, es imposible alcanzar una certeza del 100%).

Para desarrollar la prueba de caja blanca existen varias técnicas, entre ellas están: [PRESSMAN, 2004]

Prueba del camino básico

Esta prueba permite al diseñador de casos de prueba obtener una medida de la complejidad lógica de un diseño procedimental y usar esa medida como guía para la definición de un conjunto básico de caminos de ejecución. Los casos de prueba obtenidos del conjunto básico garantizan que durante la prueba se ejecuta por lo menos una vez cada sentencia del programa.

Prueba de condición

Es un método de diseño de casos de prueba que ejercita las condiciones lógicas contenidas en el módulo de un programa.

Prueba de flujo de datos

Se selecciona caminos de prueba de un programa de acuerdo con la ubicación de las definiciones y los usos de las variables del programa.

Prueba de bucles

Es una técnica de prueba de caja blanca que se centra exclusivamente en la validez de las construcciones de bucles.

1.6.2 Método de Caja Negra

Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro. Este tipo de prueba permite al ingeniero del software obtener un conjunto de condiciones de entrada que ejercitan completamente todos los requisitos funcionales de un programa. Es un enfoque complementario a al método de caja blanca que intenta descubrir diferentes tipos de errores.

Las pruebas de caja negra intentan encontrar errores de las siguientes categorías: [PRESSMAN, 2000]
[GLEN MYERS, 1979]

- Funciones incorrectas o ausentes.

- Errores de interfaz.
- Errores en la estructura de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Las pruebas de caja negra están especialmente indicadas en aquellos módulos que van a ser interfaz con el usuario. Se apoyan en la especificación de requisitos del módulo. Estas pruebas se diseñan para responder a las siguientes preguntas: [PRESSMAN, 2004]

- ¿Cómo se prueba la validez funcional?
- ¿Qué clases de entrada compondrán los buenos casos de prueba?
- ¿Es el sistema particularmente sensible a ciertos valores de entrada?
- ¿De que forma están aislados los límites de una clase de datos?
- ¿Qué volúmenes y niveles de datos tolerará el sistema?
- ¿Qué efectos sobre la operación del sistema tendrán combinaciones específicas de datos?

Mediante las técnicas de prueba de caja negra se obtiene un conjunto de casos de prueba que satisfacen los siguientes criterios:

- Casos de prueba que reducen, en un coeficiente que es mayor que uno, el número de casos de prueba adicionales que se deben diseñar para alcanzar una prueba razonable.
- Casos de prueba que dicen algo sobre la presencia o ausencia de clases de errores en lugar de errores asociados solamente con la prueba que se está realizando.

Para desarrollar la prueba de caja negra existen varias técnicas, entre ellas están: [PRESSMAN, 2000]

1. *Técnica de la Partición de Equivalencia*: Esta técnica divide el campo de entrada en clases de datos que tienden a ejercitar determinadas funciones del *software*.
2. *Técnica del Análisis de Valores Límites*: Esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
3. *Técnica de Grafos de Causa-Efecto*: Es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

Particiones o clases de equivalencias

Cada caso debe cubrir el máximo número de entradas. Debe tratarse el dominio de valores de entrada dividido en un número finito de clases de equivalencia que cumplan la siguiente propiedad: la prueba de un valor representativo de una clase permite suponer «razonablemente» que el resultado obtenido (existan defectos o no) será el mismo que el obtenido probando cualquier otro valor de la clase. Lo que se debe hacer: [VÁZQUEZ, 2001]

1. Identificar clases de equivalencia

Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. Las clases de equivalencia se pueden definir de acuerdo con las siguientes directrices:

- Si una condición de entrada especifica un rango, se definen una clase de equivalencia válida (dentro del rango) y dos inválidas (por debajo del rango y por encima del rango).
- Si una condición de entrada requiere un valor específico, se definen una clase de equivalencia válida (el valor específico) y dos inválidas (por debajo del valor y por encima del valor).
- Si una condición de entrada especifica un miembro de un conjunto, se definen una clase de equivalencia válida (dentro del conjunto) y una inválida (fuera del conjunto).
- Si una condición de entrada es lógica, se definen una clase válida (cierto) y una inválida (falso).

De igual forma se construyen las particiones de equivalencia para los valores del dominio de salida.

2. Crear casos de prueba correspondiente

Se pueden definir los casos teniendo en cuenta lo siguiente:

- Escribir un nuevo caso que cubra tantas clases de equivalencia válidas no cubiertas como sea posible hasta que todas las clases de equivalencia hayan sido cubiertas por casos de prueba.
- Escribir un nuevo caso de prueba que cubra una y solo una clase de equivalencia inválida hasta que todas las clases de equivalencias inválidas hayan sido cubiertas por casos de pruebas.

Análisis de valores límites (AVL)

Más que elegir «cualquier» elemento como representativo de una clase de equivalencia, se requiere la selección de uno o más elementos tal que los márgenes se sometan a prueba. La experiencia indica que los casos de prueba que exploran las condiciones límite de un programa producen un mejor resultado

para detectar defectos. Las directrices para derivar casos de prueba utilizando esta técnica son: [VAZQUEZ, 2001]

- Para una condición de entrada que especifica un rango limitado por los valores a y b , los casos de prueba deben incluir los valores a y b y los valores justo por debajo de a y justo por encima de b .
- Para una condición de entrada que especifica un número de valores, los casos de prueba deben incluir el valor mínimo, el máximo y los valores justo por debajo del mínimo y justo por encima del máximo.
- Aplicar las directrices 1 y 2 a las condiciones de salida.
- Si las estructuras de datos internas tienen límites preestablecidos, debe diseñarse un caso de prueba que ejercite la estructura en sus límites.

Grafos causa-efecto.

La técnica se basa en la exploración de las entradas o combinaciones de entradas al sistema de un cierto caso de uso, para determinar casos de prueba independientes.

Se utilizan grafos booleanos para representar las relaciones existentes entre las entradas y salidas al sistema. Las salidas son los llamados efectos. En el grafo booleano se representan los operadores lógicos: *and*, *or*, *or exclusivo* y la negación. Cada grafo comienza con las entradas posibles (entradas que el usuario puede realizar sobre el sistema) y termina por eventos observables (salidas que se pueden comprobar externamente tal como mensajes o salidas en pantalla o actualizaciones en bases de datos).

Es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones. Esta técnica sigue cuatro pasos: [VASQUEZ, 2001]

- Se listan para un módulo las causas (condiciones de entrada) y los efectos (acciones), asignando un identificador a cada uno de ellos.
- Se desarrolla un grafo de causa-efecto.
- Se convierte el grafo en una tabla de decisión.

- Las reglas de la tabla de decisión se convierten a casos de prueba.

Conjetura de errores

Se enumera una lista de posibles equivocaciones típicas que pueden cometer los desarrolladores y de situaciones propensas a ciertos errores. [PRESSMAN, 2004]

- El valor cero es una situación propensa a error tanto en la salida como en la entrada
- En situaciones en las que se introduce un número variable de valores, conviene centrarse en el caso de no introducir ningún valor y en el de un solo valor. También puede ser interesante un lista que tiene todos los valores iguales
- Es recomendable imaginar que el programador pudiera haber interpretado algo mal en la especificación.
- También interesa imaginar lo que el usuario puede introducir como entrada a un programa

Pruebas de comparación.

En los casos en que la aplicación es crítica a veces se generan dos versiones de la misma y se prueban de forma independiente, asegurando de ese modo que los resultados no sólo son correctos, sino que son idéntico sen cualquier circunstancia. [PRESSMAN, 2004]

1.7 Estado actual de las pruebas en productos Multimedia

Se realizó un estudio de la situación actual -tanto a nivel nacional como internacional- sobre las pruebas realizadas a aplicaciones multimedia. Aunque muchas de las empresas tienen poca experiencia en este tipo de aplicación, no se realizan pruebas de forma desorganizada, un ejemplo de esto, son las empresas mencionadas a continuación.

Internacional

La Fundación Parque Tecnológico del Software (**ParqueSoft**) cuenta con un modelo estratégico de apoyo a través de 5 estrategias. Dentro de los cuatro grandes programas que actualmente adelanta la Estrategia de Calidad ParqueSoft se encuentra **Calidad Productos de Software** (Programa Pruebas de Usabilidad Multimedia). Este programa conformó un equipo para definir herramientas de trabajo, entre ellas se destaca el estándar de usabilidad por tipo de multimedia. En el equipo participan ingenieros de sistemas, diseñadores gráficos e ingenieros de pruebas. El Programa Pruebas de Usabilidad Multimedia para niños evalúa mediante la observación en uso, las multimedia desarrolladas con un grupo de niños entre los 6 y 14 años que utilizan las multimedia como usuarios finales y que permite que el ingeniero de pruebas líder, evalúe un conjunto de criterios previamente definidos, logrando encontrar aspectos de mejoramiento en el software asociados a la calidad en uso de los productos. [ARENAS, 2006]

El proyecto **Imagenio** elaborado por Telefonía I+D se desenvuelve en un entorno multimedia definiendo ente una de sus fases del ciclo de vida en: Viabilidad, Definición de requisitos, Análisis de sistemas, Diseño de arquitectura, Diseño detallado y construcción, Integración de sistemas, Pruebas de sistemas, Pruebas de certificación y Soporte de campo. Solo las Pruebas de sistemas se dedica a realizar la verificación interna del producto, antes de realizar la entrega final al cliente, entre una de sus actividades está un Plan de prueba que va a recopilar los casos de pruebas a ejecutar para cada funcionalidad y las pruebas de robustez realizadas a las aplicaciones graficas. [TELEFONICA, 2007]

Nacional

La Empresa Cubana de Tecnología de la Información y Servicios Telemáticos (**CITMATEL**) tiene creado un grupo de especialistas de calidad. Para el desarrollo de multimedia han creado una metodología basada en RUP, bajo esta metodología realizan pruebas de funcionalidad y otras basadas en el conocimiento y experiencia de los probadores. Los productos probados en esta empresa salen al mercado certificados bajo la norma ISO 9126.

El Ministerio de la Educación (**MINED**) es otro centro que realiza pruebas de validación a multimedia educativas, utilizando protocolos creados en el centro. La técnica que utilizan es la de Caja Negra.

En la Universidad de las Ciencias Informáticas (**UCI**), tiene creado un grupo de calidad llamado Calisof con el fin de certificar los productos desarrollados dentro de la Universidad. Las pruebas realizadas a las multimedia se realizan de forma desorganizadas por la poca documentación que es entregada al grupo y la inexistencia de un proceso que guíe la realización de las pruebas. Las pruebas que se realizan son las de funcionalidad a partir de una plantilla de no conformidades se recogen los errores que presentó la multimedia.

1.8 Conclusiones

Las pruebas del software son un elemento crítico dentro del aseguramiento de la calidad de un producto, de ahí la importancia que tienen para que estos salgan libres o con la menor cantidad de errores posibles. Para que un proceso de prueba sea efectivo se recomienda llevar a cabo una planificación en tiempo donde quede documentado tanto los elementos a probar, los recursos y responsabilidades como los resultados de las pruebas a realizar.

Se debe diseñar métodos de prueba que tengan la mayor probabilidad de encontrar el mayor número de errores con la mínima cantidad de esfuerzo y tiempo. Para el diseño de los casos de pruebas es recomendable utilizar los métodos de Caja Blanca y Caja Negra que permita validar y verificar tanto el comportamiento interno como externo del producto que se va a evaluar.

Debido que en algunas empresas -en la que se encuentra nuestra Universidad -no se tiene una vasta experiencia en cuanto a pruebas en aplicaciones multimedia se recomienda tomar en cuenta este proceso para a la hora de evaluar los productos finales salgan con el cumplimiento de todos los requisitos planteados por el cliente y así, competir en el mercado ya sea a nivel nacional o internacional.

CAPÍTULO 2: PROCESO DE PRUEBAS DEL SOFTWARE

Al analizar los procesos de pruebas realizados por las empresas tanto nacionales como internacionales se hace evidente que muchas de estas no aplican un proceso de pruebas como tal, entre uno de los problemas presentes es que no contemplan las actividades de planificación, seguimiento, medición y evaluación que permitan insertarlo en una filosofía de mejora continua. La mayoría de estas empresas aplican las pruebas en la fase de transición cuando ya el producto está listo para su entrega, esto trae consigo que los errores que pueda presentar el producto sean corregidos por los clientes o usuarios finales aumentando así tiempo del software.

Si muchas de las empresas-en la que se encuentra la Universidad- aplicaran las pruebas de software de manera sistemática, se reduciría el costo y el tiempo global de desarrollo, pues se detectarían una buena cantidad de problemas en fases tempranas del proceso. Como consecuencia de estos problemas no aplicar un proceso de prueba trae como resultado:

- Liberación de un producto inmaduro lo cual es una labor penosa y costosa de soporte a usuarios insatisfechos, así como altos costos de mantenimiento.
- Insatisfacción del cliente trae como consecuencia pérdida de imagen de la empresa, así como una baja de la credibilidad en la misma.
- Reducción de las ventas por la mala calidad del producto.

Tomando en cuenta que es una necesidad establecer un proceso de prueba para erradicar los errores que hasta ahora están presentando la entrega de los productos multimedias en la Universidad se propuso un proceso de pruebas aplicado a la multimedia de la FEU.

2.1 Proceso de prueba propuesto para la Multimedia FEU

El proceso de prueba que se propone con el objetivo de corregir los errores antes de la entrega del producto está guiado por los niveles de pruebas en los que se puede dividir un software, con adaptaciones específicas teniendo en cuenta las características que presenta la multimedia. Dicho proceso está dividido en cuatro etapas fundamentales:

Planificación de las pruebas

El principal objetivo es la planificación del proceso de prueba. Al comenzar el desarrollo se determina los objetivos, se define la estrategia de aplicación de las pruebas, las responsabilidades y se estiman el tiempo y los recursos requeridos para las pruebas. Como resultado se elabora un Plan de Prueba que servirá de guía para el proceso de prueba propuesto

Diseño de las pruebas

El propósito general es el diseño de los casos de pruebas que se aplicaran al software a probar, documentando el procedimiento de prueba a seguir para la ejecución del caso de prueba, los datos de entrada y los resultados esperados. Para el diseño de los casos de pruebas se describieron dos plantillas teniendo en cuenta el requerimiento a probar.

Ejecución de las pruebas

Comprende el desarrollo de las pruebas donde se ejecutan los casos de pruebas diseñados para verificar si los requerimientos fueron cumplidos. Para registrar los errores encontrados en la ejecución de las pruebas se describe una plantilla donde se listan los errores y dificultades detectadas en la multimedia.

Análisis de defectos

Tiene como propósito analizar los errores y dificultades encontradas en la multimedia, se identifican los defectos teniendo en cuenta la categoría en la que se encuentra y sus probables causas. Además, se determina el tiempo y costos implicados en el proceso de prueba.

Aplicar este proceso de prueba trae consigo erradicar gran parte de los errores que puede presentar la multimedia evaluada y que una vez concluido dicho proceso y realizando la depuración a los errores detectados los clientes o usuarios finales queden satisfechos con el producto desarrollado alcanzando la multimedia un alto nivel de calidad.

2.2 Descripción del Software

El producto multimedia Federación Estudiantil Universitaria (FEU) de la Universidad de las Ciencias Informáticas contiene información referente a las actividades que lleva a cabo la Universidad. Para una mejor comprensión del tema tratado, cuenta con fotos, videos y detalles de la vida universitaria que hacen de esta institución una universidad de excelencia. Entre una de las características que presenta se encuentre la navegación plena y la salida de la aplicación desde cualquier pantalla. La multimedia está compuesta por un módulo dividido en 11 temas de contenido (Ver anexo 4), los temas son:

- Quienes somos

Contiene información sobre los inicios de la FEU en la UCI, su estructura y el funcionamiento desde la brigada.

- Docencia

Refleja la vida docente del universitario

- Producción

El tema trata sobre la vinculación estudio-trabajo que llevan los estudiantes en la Universidad, los proyectos con los que cuentan y el desarrollo de estos con diversos organismos tanto a nivel nacional como internacional.

- Batalla de ideas

Presenta información sobre la participación de la FEU de la UCI en la actual batalla de ideas, actos y marchas, además información sobre el procesamiento de encuestas realizadas en los laboratorios de la Universidad, su participación en la Misión Milagro y las opiniones de nuestra FEU ante las intervenciones de nuestro Comandante.

- Residencia

Refleja la vida de los estudiantes en la residencia, su organización y las actividades que realizan en ella.

- **Eventos**

Contiene información sobre los eventos científicos realizados en la Universidad, Copa Pascal, Mi Web por Cuba, Seminario Juvenil Martiano que culminan con la presentación de los mejores trabajos en la Jornada Científica Estudiantil.

- **Cultura**

Trata sobre las actividades culturales realizadas en la Universidad, talleres, peñas, charlas con artistas invitados. También cuenta con información del movimiento de artistas aficionados y de los festivales culturales que se efectúan cada año dentro de la institución.

- **Deporte**

Contiene información sobre las actividades realizadas en los juegos deportivos que se llevan a cabo todos los años en la Universidad y de los atletas más destacados.

- **Vida universitaria**

Contiene información sobre las elecciones del Poder Popular realizadas en el año 2005 en la Universidad, actividades recreativas, salidas, agrupaciones musicales y humorísticas que nos han visitado.

- **Visitante**

Presenta información sobre las personalidades que hasta el momento han visitado la Universidad.

- **Artículos**

Muestra artículos publicados en la intranet de la Universidad.

2.3 Plan de Prueba General

2.3.1 Descripción de Aspectos Generales

Objetivos.

Este documento describe el Plan de pruebas para la multimedia FEU. En concreto define los siguientes objetivos específicos:

- Identificar los elementos que se van a probar.
- Describir la estrategia de pruebas que se va a seguir en el proceso de prueba.
- Identificar los recursos necesarios para llevar a cabo el proceso de prueba y estima los esfuerzos que conlleva.

Especificaciones del Software (SW) y Hardware (HW)

Dependiendo de la necesidad específica para realizar la estrategia de prueba, éstas se harán en el laboratorio 206 de la facultad 8. Para el diseño y ejecución de las pruebas se contó con un probador, además fue necesario contar con recursos tangibles dentro de los cuales podemos mencionar los siguientes:

Una PC que como requerimiento de *software* debería tener:

- Windows 98 o superior

Y como requerimiento de *hardware*:

- Pentium II 300MHz o superior
- 64MB de RAM
- Lector de CD ROM
- Tarjeta de sonido
- Tarjeta de video 16 bits de colores

Alcance

Este Plan de Pruebas describe las pruebas a Nivel de Unidad y las Pruebas de Sistema que se aplicarán al software desarrollado. El objetivo es probar todos los requisitos definidos en la Descripción de requerimientos a probar aplicando pruebas de Caja Negra a todas las partes de la multimedia para verificar que las funcionalidades estén correctamente implementadas. No se hará uso del método de Caja Blanca ya que para la realización del proceso de prueba no fue entregado el código fuente del producto a evaluar.

Información del proyecto

La ingeniería de software no fue elaborada correctamente debido a que la multimedia fue desarrollada en un corto plazo. La construcción de la multimedia se realizó en la herramienta Director y se elaboró una lista con los requerimientos funcionales y no funcionales del producto. Para aplicar el proceso de prueba fue entregada la versión 1 del producto junto con el siguiente documento:

Tabla 2. Referencia

Título	Fecha	Organización	Identificador del documento
1 <i>Requisitos Funcionales y No Funcionales de la multimedia "FEU"</i>	<25/03/07>	<UCI>	<Id_Req>

2.3.2 Descripción de Requerimientos a probar.

Los elementos a probar de la multimedia están relacionados con aquellos requisitos que permitan su total funcionamiento, como por ejemplo, permitir al usuario acceder a todas las pantallas de la multimedia, cargar la presentación, videos e imágenes de la multimedia, además de los requerimientos de software y hardware que permitan ejecutar la multimedia. Cualquier requerimiento no incluido en esta lista estará fuera del alcance de las pruebas.

2.3.2.1 Requerimientos Funcionales

Requerimientos Generales.

- R1. Mostrar pantalla de inicio
- R2. Escuchar música de fondo.
 - R2.1.Desactivar el botón sonido si está activado
 - R2.2 Activar el botón sonido si está desactivado.
- R3. Permitir ir a la siguiente pantalla a través del botón siguiente.

- R4. Permitir la salida de la aplicación
- R5. Visualizar los créditos.
- R7. Permitir desplazar el scroll cuando este presente la barra de desplazamiento.
- R8. Mostrar pantalla de imágenes.
 - R8.1. Mostrar la imagen siguiente al dar clic sobre el botón correspondiente a siguiente.
 - R8.2. Mostrar la imagen anterior al dar clic sobre el botón correspondiente anterior.
 - R8.3. Permitir la salida de la pantalla de imágenes
- R9. Mostrar pantalla de video
 - R9.1. Desactivar música de fondo al visualizarse la pantalla de videos.
 - R9.2. Mostrar contenido de video.
 - R9.3. Activar pausa al video al dar clic sobre el botón correspondiente a pausa, si esta en reproducción el video.
 - R9.4. Activar reproducción al video al dar clic sobre el botón correspondiente a reproducción, si esta activado el botón pausa.
 - R9.5. Permitir la salida de la pantalla de videos

Requerimientos Funcionales por tema de contenido

- R10. Mostrar submenú de ¿Quiénes somos?
 - R10.1. Mostrar contenido de la pantalla Inicios
 - R10.2. Mostrar contenido de la pantalla Estructura
 - R10.3. Mostrar contenido de la pantalla Funcionamiento
- R11. Mostrar contenido de la pantalla Docencia
- R12. Mostrar contenido de la pantalla Producción
- R13. Mostrar submenú de Batalla de ideas
 - R13.1. Mostrar contenido de la pantalla Batalla de ideas

- R13.2. Mostrar contenido de la pantalla Actos
- R13.3. Mostrar contenido de la pantalla Banderitas
- R13.4. Mostrar contenido de la pantalla Encuestas
- R13.5. Mostrar contenido de la pantalla Marchas
- R13.6. Mostrar contenido de la pantalla Misión Milagro
- R13.7. Mostrar contenido de la pantalla Opiniones

- R14. Mostrar contenido de la pantalla Residencia.
- R15. Mostrar contenido de la pantalla Eventos.
- R16. Mostrar submenú de Cultura
 - R16.1. Mostrar contenido de la pantalla Actividades
 - R16.2. Mostrar contenido de la pantalla Movimiento artístico
 - R16.3. Mostrar contenido de la pantalla Emulación

- R17. Mostrar submenú de Deporte
 - R17.1. Mostrar contenido de la pantalla Juegos deportivos
 - R17.2. Mostrar contenido de la pantalla Más destacados

- R18. Mostrar submenú de Vida universitaria
 - R18.1. Mostrar contenido de la pantalla Vida universitaria
 - R18.2. Mostrar contenido de la pantalla Elecciones
 - R18.3. Mostrar contenido de la pantalla Recreación
 - R18.4. Mostrar contenido de la pantalla Desde las facultades

- R19. Mostrar contenido de la pantalla Visitantes
- R20. Mostrar contenido de la pantalla Artículos.
 - R20.1. Mostrar contenido del artículo: La ciencia de todos los días
 - R20.2. Mostrar contenido del artículo: Una vez mas, la UCI

- R20.3 Mostrar contenido del artículo: Las banderitas ya están pegadas.
- R20.4 Mostrar contenido del artículo: Aires astrales para una gala.
- R20.5 Mostrar contenido del artículo: INFODANZ, la informática y la danza.
- R20.6 Mostrar contenido del artículo: I Activo de Residencia de la UCI.
- R20.7 Mostrar contenido del artículo: Visita partidista en la UCI.
- R20.8 Mostrar contenido del artículo: Por qué son importantes los jueves.
- R20.9 Mostrar contenido del artículo: Relevante actuación de los ajedrecistas de la UCI.
- R20.10 Mostrar contenido del artículo: Se cierran las cortinas de la fiesta deportiva.
- R20.11 Mostrar contenido del artículo: Rinden homenaje a Camilo la FEU y la UJC de la UCI.
- R20.12 Mostrar contenido del artículo: Ya se formó la rueda, ya se formó...
- R20.13 Mostrar contenido del artículo: Campeones en casa
- R20.14 Mostrar contenido del artículo: Celebrada reunión de inicio de curso en la UCI.
- R20.15 Mostrar contenido del artículo: Ya estamos en congreso.
- R20.16 Mostrar contenido del artículo: “Esta es la industria del futuro para todos”, afirmó el presidente de Malasia en la UCI.
- R20.17 Mostrar contenido del artículo: Nuestro primer oro
- R20.18 Mostrar contenido del artículo: Celebrado acto de la amistad entre los pueblos de China y Cuba.
- R20.19 Mostrar contenido del artículo: El antes y el después de nuestra FEU.
- R20.20 Mostrar contenido del artículo: Concluyó la visita integral de la UJC y la FEU.

2.3.2.2 Requerimientos no Funcionales a probar

Requerimiento no funcional de Apariencia o interfaz externa:

- RNF1. El diseño de la interfaz debe mantener el mismo fondo y opciones posibles en todas las pantallas de los módulos
- RNF2. El contenido de información que se presenta debe estar escrito correctamente
- RNF3. La multimedia debe cargarse a pantalla completa.
- RNF4. La multimedia será boteable desde la torre de CD

Requerimiento no Funcional de Rendimiento:

- RNF6.La aplicación debe garantizar que la velocidad de procesamiento sea lo mas alta posible, permitiendo así, un tiempo de respuesta corto ante las acciones del usuario

Requerimiento no Funcional de Software.

- RNF7.La aplicación debe correr sobre el Sistema Operativo Windows 98 o superior.

Requerimiento no Funcional Hardware

- RNF8.La aplicación debe ejecutarse en una Pentium II 300WHz o superior, con 64MB de RAM, Lector de CD ROM, Tarjeta de sonido, Tarjeta de video 16 bits de colores y unas bocinas o audífonos.

2.3.3 Definición de la Estrategia de Pruebas.

La estrategia de aplicación de las pruebas definida consta de dos niveles donde se aplicarán los tipos de pruebas que serán ejecutados:

- Pruebas a Nivel de Unidad. Se aplicarán pruebas al diseño donde se debe comprobar su consistencia, es decir, que el diseño sea realizable (Pruebas de Interfaz).
- Prueba a Nivel de Sistema. Las pruebas de Sistemas se ejecutaran en la aplicación completa. Estas pruebas tienen como objetivo verificar el funcionamiento de la multimedia a través de las interfaces. Para dar cumplimiento se aplicarán Pruebas Funcionales verificando los requerimientos funcionales de la multimedia. El método a utilizar para este tipo de prueba es el de Caja Negra a través de la técnica de partición equivalente. Los requerimientos no funcionales se verificarán a través de las Pruebas de Solidez, Pruebas de Resistencia (stress) y Pruebas de Soportabilidad.

2.3.3.1 Pruebas de Interfaz

Objetivos:	Verificar el cumplimiento de los requisitos no funcionales de Apariencia e Interfaz de usuario para comprobar el diseño gráfico especificado.
Técnica:	La forma de realizar los casos de pruebas no sigue ningún criterio específico, solo requiere de una interacción (tarea a realizar sobre el sistema) a través de la interfaz de la multimedia FEU.
Diseño de Casos de Pruebas:	Anexo 3
Criterios de Término:	Cada pantalla se ha verificado con éxito y es consistente con los estándares utilizados.
Consideraciones especiales:	Ninguna

2.3.3.2 Pruebas Funcionales

Objetivos:	Están enfocadas a asegurar que el software realiza correctamente todas las funciones que se han detallado en los requerimientos funcionales dados. Para el desarrollo de estas pruebas se hace uso del método de Caja Negra.
Técnica:	Para confeccionar los casos de prueba por el método de Caja Negra se utilizó la técnica de Particiones de Equivalencia. Las condiciones de entrada son de tipo lógica por lo que se identifica una clase válida y una no válida.
Diseño de Casos de Pruebas:	Anexo 2
Criterios de Término:	Todas las pruebas planificadas se han ejecutado.
Consideraciones especiales:	Ninguna

2.3.3.3 Pruebas de Solidez

Objetivos:	Verificar la estética de las pantallas, es decir que la información mostrada no presente errores gramáticos ni ortográficos
Técnica:	La forma de realizar los casos de pruebas no sigue ningún criterio específico, solo requiere de una interacción (tarea a realizar sobre el sistema) a través de la interfaz de la multimedia FEU.
Diseño de Casos de Pruebas:	Anexo 3
Criterios de Término:	Cada pantalla se ha verificado con éxito
Consideraciones especiales:	Ninguna

2.3.3.4 Pruebas de Soportabilidad

Objetivos:	Verificar que la multimedia funcione correctamente en los requerimientos de configuración de hardware y software mencionados.
Técnica:	La forma de realizar los casos de pruebas no sigue ningún criterio específico, solo se comprobará que la multimedia sea ejecutable en los elementos de hardware y en las versiones del Sistema Operativo Windows especificado en los requisitos no funcionales
Casos de Pruebas:	Anexo 3
Criterios de Término:	La aplicación reconoce los elementos configurados
Consideraciones especiales:	Ninguna

2.3.3.5 Pruebas de Resistencia (Stress)

Objetivos:	Evaluar el comportamiento de la aplicación bajo condiciones anormales
Técnica:	La forma de realizar los casos de pruebas no sigue ningún criterio específico, se ejecutara caso de prueba que comprueben el máximo de memoria, disponibilidad de servicios y tiempo de respuesta.
Casos de Pruebas:	Anexo 3
Criterios de Término:	La aplicación corre bajo condiciones anormales.
Consideraciones especiales:	Ninguna

2.3.4 Recursos Requeridos

En esta sección se describen los recursos necesarios para aplicar el proceso de prueba y los roles involucrado. Todos los roles y las responsabilidades para el proceso de prueba aplicado a la multimedia de la FEU serán realizadas por Yennia Rivero Águila, responsable de la entrega en fecha de la documentación del proceso.

2.3.4.1 Roles

Tabla 3. Roles y Responsabilidades asignados en el Proceso de prueba

Rol	Cantidad mínima de recursos recomendada	Responsabilidades
Administrador de pruebas	1 persona	Generar el Plan de Prueba. Adquiere los recursos apropiados.
Diseñador de los casos de pruebas	1 persona	Diseñar los casos de pruebas
Probador	1 persona	Ejecutar los casos de pruebas Registrar los resultados de las pruebas.

2.3.4.2 Recursos

Tabla 4. Recursos asignados para la ejecución de las pruebas

Recurso	Nombre/Tipo
PC para la ejecución de las pruebas	Ip 10.35.10.18

2.3.5 Actividades y tareas del proceso de prueba

Las actividades y tareas realizadas en el proceso de prueba son:

Tabla 5. Actividades y tareas del proceso de prueba

Actividades	Tareas
Planificación de las pruebas	<ol style="list-style-type: none"> 1. Identificar los requisitos para las pruebas. 1. Desarrollar la estrategia de pruebas. 2. Identificar los recursos necesarios para realizar las pruebas. 3. Generar el Plan de pruebas.
Diseño de las pruebas	<ol style="list-style-type: none"> 1. Identificar y describir los casos de prueba.
Ejecución de las pruebas	<ol style="list-style-type: none"> 1. Ejecutar los casos de prueba. 2. Verificar los resultados. 3. Investigar los resultados no esperados. 4. Registrar los defectos.
Análisis de los defectos	<ol style="list-style-type: none"> 1. Analizar los defectos. 2. Determinar si se han alcanzado los criterios de las pruebas. 3. Crear los informes de evaluación de las pruebas.

2.3.6 Definición de los Entregables

Tabla 6. Documentos del Proceso de prueba a entregar

Documentos	Descripción
Plan de Prueba General	Contiene la estrategia de prueba, los recursos, los requerimientos que serán probados.
Diseño de los casos de pruebas	Contiene los pasos a seguir, los requisitos a probar en el caso de prueba y los resultados que se esperan a aplicar las pruebas.
Registro de defectos y dificultades detectadas	Contiene los defectos encontrados en la multimedia
Resultados de las Pruebas	Contiene el análisis de los resultados reales que se obtuvieron al aplicar las pruebas.

2.3.7 Plazos y Calendarios

Las tareas del proceso de prueba deberán ser realizadas en las fechas siguientes.

Tabla 7. Tareas del proceso de prueba

Tareas	Responsable	Comienzo	Término	Duración
Plan de Prueba	Yennia Rivero A.	03/04/07	10/04/07	6 días
Diseño de los Casos de Pruebas	Yennia Rivero A.	20/04/07	01/05/07	12 días
Registro de defectos y dificultades detectadas	Yennia Rivero A.	07/05/07	18/05/07	11 días
Resultados de las Pruebas	Yennia Rivero A.	20/05/07	23/05/07	3 días

2.4 Procedimiento y Casos de Pruebas a Nivel de Unidad

2.4.1 Pruebas de Interfaz

Objetivos

Este tipo de prueba permite verificar si el diseño gráfico de la multimedia es igual en todas las pantallas del módulo. Para cumplir con los objetivos se ejecutaron los siguientes casos de pruebas.

2.4.1.1 Casos de Pruebas de Interfaz

CPR1: Diseño de las pantallas

Aspectos Generales	
Requisito: RNF1	Aprobado: No
Responsable:	
Propósito:	
El caso de prueba permite verificar que cada pantalla cumpla con el diseño gráfico especificado (color, botones y menú).	
Prueba:	
Verificar que en todas las pantallas de la multimedia se muestra: <ol style="list-style-type: none"> 1. En la parte superior de la aplicación un menú con vínculos a otras pantallas. 2. En la parte inferior de la aplicación aparece los botones salida, sonido y siguiente. 3. El color de fondo de todas las pantallas es oscuro y debe ser el mismo. 4. En las pantallas de imágenes aparece solo las opciones siguiente y anterior. 5. En las pantallas de video aparece solo las opciones reproducción y pausa. 6. Los vínculos (imágenes, videos, botones y temas) presentes cambian la coloración al pasar el Mouse por encima de ellos 	
Resultados esperado:	
Todas las pantallas cumplen con el diseño grafico	

2.5 Procedimiento y Casos de Pruebas a Nivel de Sistema

2.5.1 Pruebas Funcionales

Objetivo:

Este tipo de prueba permite verificar si el producto desarrollado cumple con los requerimientos funcionales entregados. Para cumplir con los objetivos se ejecutaron los siguientes casos de pruebas.

2.5.1.1 Casos de Pruebas a los Requerimientos Generales

CPR2: Mostrar pantalla de inicio

Aspectos Generales	
Requisito: R1	Aprobado: No
Responsable: Yennia Rivero Aguila	
<p>Descripción General:</p> <p>El caso de prueba permite verificar que la presentación y la pantalla de inicio de la multimedia se muestren correctamente.</p>	
<p>Flujo Central:</p> <p>El caso de prueba se inicia cuando se ejecuta la multimedia:</p> <ol style="list-style-type: none"> 1. Se da clic sobre el ejecutable. 2. Se muestra la presentación de la multimedia. 3. Se muestra una pantalla de inicio con los botones salida, sonido y siguiente. Ver caso de prueba <i>Escuchar música de fondo</i> 4. Ver caso de prueba <i>Salir de la aplicación</i> 5. Ver caso de prueba <i>Navegar a la siguiente pantalla</i> 	
<p>Condiciones de ejecución:</p> <p>Debe aparecer el ejecutable de la multimedia</p>	

Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar presentación al abrir la aplicación.		1 Se muestra en pantalla la presentación de la multimedia.

CPR3: Permitir el acceso a los temas

Aspectos Generales	
Requisito: R6	Aprobado: Si
Responsable: Yennia Rivero Aguila	
<p>Descripción General:</p> <p>El caso de prueba permite verificar el acceso a las pantallas de la multimedia a través de los temas de contenidos que se muestran en el menú de la multimedia.</p>	
<p>Flujo Central:</p> <p>El caso de prueba se inicia al mostrarse en la pantalla el menú de la multimedia :</p> <ol style="list-style-type: none"> 1. Se muestra en la pantalla los temas siguiente <ul style="list-style-type: none"> ▪ Quienes somos ▪ Docencia ▪ Producción ▪ Batalla de ideas ▪ Residencia ▪ Eventos ▪ Cultura ▪ Deporte ▪ Vida universitaria ▪ Visitantes ▪ Artículos 2. Se selecciona un tema del menú 3. Ver caso de prueba <i>Mostrar submenú del tema Quienes somos</i> 4. Ver caso de prueba <i>Mostrar contenido del tema Docencia</i> 	

5. Ver caso de prueba *Mostrar contenido del tema Producción*
6. Ver caso de prueba *Mostrar submenú del tema Batalla de ideas*
7. Ver caso de prueba *Mostrar contenido del tema Residencia*
8. Ver caso de prueba *Mostrar contenido del tema Eventos*
9. Ver caso de prueba *Mostrar submenú del tema Cultura*
10. Ver caso de prueba *Mostrar submenú del tema Deporte*
11. Ver caso de prueba *Mostrar submenú del tema Vida universitaria*
12. Ver caso de prueba *Mostrar contenido del tema Visitantes*
13. Ver caso de prueba *Mostrar contenido del tema Artículos*

Condiciones de ejecución:

La multimedia se cargue correctamente.

Clases de Equivalencia

Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Permitir el acceso al tema al dar clic sobre Quienes somos	(2) No permitir el acceso sin seleccionar el tema Quienes somos	1 El sistema permite acceder al tema Quienes somos 2 El sistema no permite el acceso al tema
(3) Permitir el acceso al tema al dar clic sobre Docencia	(4) No permitir el acceso sin seleccionar el tema Docencia	3 El sistema permite acceder al tema Docencia 4 El sistema no permite el acceso al tema
(5) Permitir el acceso al tema al dar clic sobre Producción	(6) No permitir el acceso sin seleccionar el tema Producción	5 El sistema permite acceder al tema Producción 6 El sistema no permite el acceso al tema
(7) Permitir el acceso al tema al dar clic sobre Batalla de ideas	(8) No permitir el acceso sin seleccionar el tema Batalla de ideas	7 El sistema permite acceder al tema Batalla de ideas 8 El sistema no permite el acceso al tema

(9) Permitir el acceso al tema al dar clic sobre Residencia	(10) No permitir el acceso sin seleccionar el tema Residencia	9 El sistema permite acceder al tema Residencia 10 El sistema no permite el acceso al tema
(11) Permitir el acceso al tema al dar clic sobre Eventos	(12) No permitir el acceso sin seleccionar el tema Eventos	11 El sistema permite acceder al tema Eventos 12 El sistema no permite acceder al tema
(13) Permitir el acceso al tema al dar clic sobre Cultura	(14) No permitir el sin seleccionar el tema Cultura	13 El sistema permite acceder al tema Cultura 14 El sistema no permite el acceso al tema
(15) Permitir el acceso al tema al dar clic sobre Deporte	(16) No permitir el acceso sin seleccionar el tema Deporte	15 El sistema permite acceder al tema Deporte 16 El sistema no permite el acceso al tema
(17) Permitir el acceso al tema al dar clic sobre Vida universitaria	(18) No permitir el acceso sin seleccionar el tema Batalla de ideas	17 El sistema permite acceder al tema Vida universitaria 18 El sistema no permite acceder al tema
(19) Permitir el acceso al tema al dar clic sobre Visitantes	(20) No permitir el acceso sin seleccionar el tema Visitantes	19 El sistema permite acceder al tema Visitantes 20 El sistema no permite el acceso
(21) Permitir el acceso al tema al dar clic sobre Artículos	(22) No permitir el acceso sin seleccionar el tema Artículos	21 El sistema permite acceder al tema Artículos 22 El sistema no permite el acceso al tema

CPR4: Navegar a la siguiente pantalla.

Aspectos Generales		
Requisito: R3	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se puede navegar de una pantalla a la siguiente en la multimedia		
Flujo Central: El caso de prueba se inicia cuando se muestra el contenido de la multimedia: <ol style="list-style-type: none"> 1. Se muestra en la pantalla las opciones salida, sonido y siguiente. Se da clic en el botón sonido. Ver caso de prueba <i>Escuchar música de fondo</i>. 2. Se da un clic sobre el botón siguiente 3. El sistema muestra la pantalla siguiente de la multimedia. 4. Repetir desde paso 1. 		
Condiciones de ejecución: La multimedia se cargue correctamente.		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Permitir acceder a la siguiente pantalla al dar clic sobre el botón siguiente	(2) No acceder a la siguiente pantalla sin seleccionar el botón siguiente	1 Se muestra la pantalla siguiente 2 No se muestra la pantalla siguiente

CPR5: Escuchar música de fondo.

Aspectos Generales		
Requisito: R2	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se escucha música o no, al activar o desactivar el botón de sonido en cualquier pantalla de la aplicación.		
Flujo Central: El caso de prueba se inicia al mostrarse el contenido de la multimedia: <ol style="list-style-type: none"> 1. Se muestra en la pantalla un menú con los temas del módulo y las opciones salida, sonido y siguiente. 2. Se oprime el botón sonido para desactivar la música de fondo, si el botón esta activado. 3. Se oprime el botón sonido para activar la música de fondo, si el botón esta desactivado. 		
Condiciones de ejecución: La multimedia se cargue correctamente		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Escuchar música de fondo	(2) No se escucha la música de fondo	1 Se escucha una música de fondo. 2 El acceder al tema no se escucha la música de fondo
(3) Desactivar música de fondo al dar clic sobre el botón de sonido.	(4)No desactivar música sin seleccionar el botón de sonido	3 Queda la opción de sonido desactivada. 4 Se escucha la música de fondo
(5) Activar música de fondo al dar clic sobre el botón de sonido.	(4)No activar música sin seleccionar el botón de sonido	3 Queda la opción de sonido activada. 4 No se escucha la música de fondo

CPR6: Salir de la aplicación

Aspectos Generales		
Requisito: R4	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar la salida de la aplicación desde cualquier pantalla de la multimedia.		
Flujo Central: El caso de prueba se inicia cuando se desea salir de la multimedia. <ol style="list-style-type: none"> 1. Se encuentra en una de las pantallas de la multimedia 2. Se muestra en la pantalla las opciones salidas, sonido y siguiente 3. Se oprime el botón correspondiente a la salida de la aplicación. Ver caso de prueba <i>Visualizar créditos</i> 		
Condiciones de ejecución: La multimedia debe cargarse correctamente		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Salir de la aplicación al dar clic sobre en botón de salida	(2) No salir de la aplicación sin seleccionar el botón de salida	1 El sistema permite la salida de la multimedia 2 El sistema no permite la salida

CPR7: Visualizar créditos

Aspectos Generales		
Requisito: R5	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si la multimedia visualiza la pantalla de los créditos		
Flujo Central: El caso de prueba se inicia se oprime el botón de salida. <ol style="list-style-type: none"> 1. Se muestra una pantalla con el tema de despedida de la multimedia. 2. Se muestran una pantalla con los créditos de la multimedia. 3. La aplicación se cierra. 		
Condiciones de ejecución: La multimedia se cargue correctamente		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Visualizar créditos.		1 Se muestra una pantalla con los créditos.

CPR8: Permitir desplazar el scroll

Aspectos Generales		
Requisito: R7	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si el usuario puede desplazar el scroll cuando este presente la barra de desplazamiento.		
Flujo Central: El caso de prueba se inicia cuando se accede a una pantalla y se muestra una barra de		

desplazamiento:

1. Se muestra información en la pantalla
2. Se muestra en la pantalla una barra de desplazamiento
3. Se desplaza el scroll de la barra de desplazamiento.
4. La aplicación permite desplazar el scroll y ver el contenido que presenta.

Condiciones de ejecución:

La pantalla presenta una barra de desplazamiento

Clases de Equivalencia

Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Permitir desplazar el scroll al dar clic sobre la barra de desplazamiento		1. El sistema muestra la información contenida.

CPR9: Mostrar pantalla de imágenes

Aspectos Generales	
Requisito: R8	Aprobado: Si
Responsable: Yennia Rivero Aguila	
Descripción General: El caso de prueba permite verificar si se pueden visualizar las imágenes que aparecen en las pantallas de la multimedia.	
Flujo Central: El caso de prueba se inicia cuando se entra en una pantalla que contiene imágenes: <ol style="list-style-type: none"> 1. La pantalla muestra imágenes en la parte derecha-superior. 2. Se da clic sobre una de las imágenes. 3. Se muestra una pantalla con la imagen seleccionada. 4. La pantalla muestra los botones anterior y/o siguiente en dependencia de la imagen seleccionada. 5. Se selecciona: 	

- La opción siguiente si desea ver la imagen que le sigue de la lista
 - La opción anterior si desea ver la imagen anterior de la lista
6. Se oprime el botón de salida correspondiente a la pantalla de imágenes.
 7. El sistema muestra la pantalla en las que se encuentran la imagen que se seleccionó.

Condiciones de ejecución:

El caso de prueba se ejecuta si la pantalla seleccionada contiene imágenes

Clases de Equivalencia

Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Visualizar pantalla de imágenes al dar clic sobre una imagen.	(2) No visualizar la pantalla de imágenes sin seleccionar una imagen	1. El sistema muestra una pantalla con la imagen seleccionada. 2 El sistema no permite visualizar la pantalla de imágenes
(3) Mostrar imagen siguiente al dar clic sobre el botón siguiente	(4) No mostrar la imagen sin seleccionar el botón siguiente	3 Se muestra la imagen que sigue en la pantalla. 4 No se muestra en la pantalla la imagen siguiente
(5) Mostrar la imagen anterior al dar clic sobre el botón anterior	(6) No mostrar la imagen anterior sin seleccionar el botón anterior	5 Se muestra la imagen anterior en la pantalla. 6 No se muestra en la pantalla la imagen anterior
(7) Salir de la pantalla al dar clic sobre el botón salida.	(8) No permitir la salida de la pantalla sin seleccionar el botón de salida	7 El sistema permite la salida de la pantalla. 8 El sistema no permite la salida de la pantalla

CPR10: Mostar pantalla de video

Aspectos Generales		
Requisito: R9	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra la pantalla de video.		
Flujo Central: El caso de prueba se inicia cuando se entra en una pantalla que contiene video : <ol style="list-style-type: none"> 1. Se muestran uno o varios videos en la parte derecha -inferior de la pantalla. 2. Se da clic sobre un video. 3. Se muestra una pantalla con el video seleccionado 4. La pantalla muestra las opciones reproducción y pausa. 5. El sistema desactiva la música de fondo. 6. El sistema muestra el contenido del video. 7. Se selecciona pausa al video, si el video esta en reproducción 8. Se selecciona reproducción al video, si el video esta en pausa. 9. Se oprimir el botón de salida correspondiente a la pantalla de video. 10. Se muestra la pantalla en la que se encuentra el video. 11. El sistema activa la música. 		
Condiciones de ejecución: El caso de prueba se ejecuta si la pantalla seleccionada contiene videos.		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Visualizar pantalla de video al dar clic sobre un video.	(2) No visualizar pantalla de video sin seleccionar un video	1. El sistema muestra la pantalla de video 2 El sistema no muestra la pantalla de video
(3) Desactivar la música de	(4) No desactivar la música	3. El sistema desactiva

fondo al visualizar la pantalla	de fondo sino se visualiza la pantalla	automáticamente la música de fondo del tema. 4 El sistema no desactiva la música de fondo
(5) Mostrar contenido del video.	(6) No se muestra el contenido del video	5. Se muestra la información que contiene el video 6 No se muestra la información que contiene el video
(7) Activar la pausa del video al dar clic sobre el botón pausa.	(8) No activar la pausa del video sin seleccionar el botón	7 Se detiene la reproducción del video. 8 El sistema no activa la pausa del video.
(9) Activar la reproducción del video al dar clic sobre el botón reproducción.	(10) No activar la reproducción del video sin seleccionar el botón	9 El sistema activa la reproducción del video 10 El sistema no activa la reproducción
(11) Salir de la pantalla de video al dar clic sobre el botón de salida.	(12) No salir de la pantalla video sin seleccionar el botón de salida	11. El sistema permite la salida de la pantalla. 12 El sistema no permite la salida de la pantalla.

2.5.1.2 Casos de Pruebas por tema de contenido

CPR11: Mostrar submenú del tema ¿Quiénes somos?

Aspectos Generales	
Requisito: R10	Aprobado: Si
Responsable: Yennia Rivero Aguila	
Descripción General: El caso de prueba permite verificar si se puede acceder a los contenidos que presenta el tema “¿Quiénes somos? “.	

<p>Flujo Central:</p> <p>El caso de prueba se inicia se accede al tema ¿Quienes somos?</p> <ol style="list-style-type: none"> 1. Se da clic sobre ¿Quienes somos? 2. Se muestra un submenú con los siguientes temas: <ol style="list-style-type: none"> a. Inicio b. Estructura c. Funcionamiento 3. Si se selecciona Inicios. Se muestra en pantalla el contenido e imágenes. Ver casos de pruebas <i>Mostrar pantalla de imágenes y Permitir desplazar el scroll</i> 4. Se da clic sobre Estructura. Se muestra en pantalla el contenido. 5. Se da clic sobre Funcionamiento. Se muestra en pantalla el contenido 		
<p>Condiciones de ejecución:</p> <p>El usuario navegue en la multimedia</p>		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar submenú del tema al dar clic sobre ¿Quienes somos?	(2) No mostrar el submenú del tema sin seleccionar ¿Quienes somos?	1 Se muestra el submenú del tema 2 No se muestra el submenú del tema
(3) Mostrar contenido al dar clic sobre Inicios	(4) No mostrar el contenido sin seleccionar Inicios	3 Se muestra en la pantalla el contenido de Inicios. 4 No se muestra el contenido del tema Inicios
(5) Mostrar contenido al dar clic sobre Estructura	(6) No mostrar el contenido sin seleccionar Estructura	5. Se muestra en la pantalla el contenido de Estructura. 6 No se muestra el contenido del tema Estructura
(7) Mostrar contenido al dar clic sobre Funcionamiento	(8) No mostrar el contenido sin seleccionar Funcionamiento	7 Se muestra en la pantalla el contenido de Funcionamiento 8 No se muestra el contenido del tema Funcionamiento

CPR12: Mostrar contenido del tema Docencia

Aspectos Generales		
Requisito: R11	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Docencia.		
Flujo Central: El caso de prueba se inicia cuando al dar clic sobre el tema Docencia se puede acceder a la pantalla <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Docencia 2. Se muestran imágenes en pantalla. Ver caso de prueba <i>Mostrar pantalla de imágenes.</i> 		
Condiciones de ejecución: El usuario pudo acceder a la pantalla		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar contenido al dar clic sobre el tema Docencia		1. Se muestra en la pantalla el contenido de Docencia.

CPR13: Mostrar contenido del tema Producción

Aspectos Generales		
Requisito: R11	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Producción.		
Flujo Central: El caso de prueba se inicia cuando al dar clic sobre el tema Producción se puede acceder a la		

pantalla <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Producción 2. Se muestran imágenes en pantalla. Ver caso de prueba <i>Mostrar pantalla de imágenes</i>. 		
Condiciones de ejecución: El usuario pudo acceder a la pantalla		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar contenido al dar clic sobre el tema Producción		1 Se muestra en la pantalla el contenido de Producción

CPR14: Mostrar submenú del tema Batalla de ideas

Aspectos Generales	
Requisito: R13	Aprobado: No
Responsable: Yennia Rivero Aguila	
Descripción General: El caso de prueba permite verificar si se puede acceder a los contenidos que presenta el tema “Batalla de ideas “.	
Flujo Central: El caso de prueba se inicia si se accede al tema Batalla de ideas: <ol style="list-style-type: none"> 1. Se da clic sobre Batalla de ideas 2. Se muestra un submenú con los siguientes temas: <ul style="list-style-type: none"> ▪ Actos ▪ Banderitas ▪ Encuestas ▪ Marchas ▪ Misión Milagro ▪ Opiniones 3. Se selecciona Batalla de ideas. Se muestra en pantalla imágenes y dos videos. Ver casos de pruebas <i>Mostrar pantalla de imágenes</i> y <i>Mostrar pantalla de video</i>. 	

4. Si se selecciona uno de los temas Actos, Banderitas, Encuestas, Marchas, Misión Milagro u Opiniones. Se muestra en pantalla el contenido e imágenes. Ver caso de prueba *Mostrar pantalla de imágenes*

Condiciones de ejecución:

El usuario navegue en la multimedia

Clases de Equivalencia

Clases Válidas	Clases Inválidas	Resultado Esperado
1 Mostrar submenú del tema al dar clic sobre Batalla de ideas	(2) No mostrar el submenú sin seleccionar el tema Batalla de ideas	1.1. Se muestra el submenú del tema 2 No se muestra el submenú
(3) Mostrar contenido de la pantalla Batalla de ideas	(4) No mostrar el contenido sin seleccionar Batalla de ideas	3. Se muestra en la pantalla el contenido de Batalla de ideas. 4 No se muestra el contenido de Actos
(5) Mostrar contenido de Actos.	(6) No mostrar el contenido sin seleccionar Actos	5. Se muestra en la pantalla el contenido de Actos. 6 No se muestra el contenido del tema
(7) Mostrar contenido al dar clic sobre Banderitas.	(8) No mostrar el contenido sin seleccionar Banderitas	7 Se muestra en la pantalla el contenido de Banderitas. 8 No se muestra el contenido del tema
(9). Mostrar contenido al dar clic sobre Encuestas.	(10) No mostrar el contenido sin seleccionar Encuestas	9 Se muestra en la pantalla el contenido de Encuestas 10 No se muestra el contenido del tema
(11) Mostrar contenido al dar clic sobre Marchas.	(12) No mostrar el contenido sin seleccionar Marchas	11 Se muestra en la pantalla el contenido de Marchas 12 No mostrar el contenido del tema
(13) Mostrar contenido al dar clic sobre Misión Milagro.	(14) No mostrar el contenido sin seleccionar Misión Milagro	13. Se muestra en la pantalla el contenido de Misión Milagro. 14 No se muestra el contenido del tema
(15). Mostrar contenido de	(16) No mostrar el	15. Se muestra en la pantalla el

Opiniones.	contenido sin seleccionar Opciones	contenido de Opiniones. 16 No se muestra el contenido del tema
------------	---------------------------------------	---

CPR15: Mostrar contenido del tema Residencia

Aspectos Generales		
	Requisito: R14	Aprobado: Si
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Residencia		
Flujo Central: El caso de prueba se inicia cuando al dar clic sobre el tema Residencia se accede a la pantalla: <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Residencia 2. Se muestran imágenes en pantalla. Ver caso de prueba <i>Mostrar pantalla de imágenes.</i> 		
Condiciones de ejecución: El usuario navegue en la multimedia		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar contenido al dar clic sobre el tema Residencia		1. Se muestra en la pantalla el contenido de Residencia

CPR16: Mostrar contenido del tema Eventos

Aspectos Generales		
	Requisito: R15	Aprobado: Si
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Eventos.		

Flujo Central:		
<p>El caso de prueba se inicia si se accede a la pantalla al dar clic sobre el tema Eventos.</p> <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Eventos. 2. Se muestran imágenes en pantalla. Ver casos de pruebas <i>Mostrar pantalla de imágenes y Permitir desplazar el scroll.</i> 		
Condiciones de ejecución:		
El usuario pudo acceder a la pantalla		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar contenido al dar clic sobre Eventos.		1. Se muestra en la pantalla el contenido de Eventos.

CPR17: Mostrar submenú del tema Cultura

Aspectos Generales	
Requisito: R16	Aprobado: Si
Responsable: Yennia Rivero Aguila	
Descripción General:	
El caso de prueba permite verificar si se puede acceder a los contenidos que presenta el tema “Cultura “.	
Flujo Central:	
El caso de prueba se inicia se accede al tema Cultura:	
<ol style="list-style-type: none"> 1. Se da un clic sobre Cultura 2. Se muestra un submenú con los siguientes temas: <ol style="list-style-type: none"> a. Actividades b. Movimiento artístico c. Emulación 3. Si se selecciona una de las pantallas Actividades o Emulación. Se muestra en pantalla el contenido e imágenes. Ver caso de prueba <i>Mostrar pantalla de imágenes</i> 	

4. Si se selecciona Movimiento artístico. Se muestra en pantalla el contenido, imágenes y video.
Ver casos de pruebas *Mostrar pantalla de imágenes y Mostrar pantalla de video.*

Condiciones de ejecución:

El usuario navegue en la multimedia

Clases de Equivalencia

Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar submenú del tema al dar clic sobre Cultura	(2) No mostrar el submenú del tema sin seleccionar Cultura	1 Se muestra el submenú del tema 2 No se muestra el submenú
(3) Mostrar contenido al dar clic sobre Actividades	(4) No mostrar el contenido sin seleccionar Actividades	3 Se muestra en la pantalla el contenido de Actividades 4 No se muestra el contenido del tema
(5) Mostrar contenido al dar clic sobre Movimiento artístico	(6) No mostrar el contenido sin seleccionar Movimiento artístico	5 Se muestra en la pantalla el contenido de Movimiento artístico. 6 No se muestra el contenido del tema
(7) Mostrar contenido al dar clic sobre Emulación.	(8) No mostrar el contenido sin seleccionar Emulación	7 Se muestra en la pantalla el contenido de Emulación. 8 No se muestra el contenido del tema

CPR18: Mostrar submenú del tema Deporte

Aspectos Generales	
Requisito: R17	Aprobado: No
Responsable: Yennia Rivero Aguila	
<p>Descripción General:</p> <p>El caso de prueba permite verificar si se puede acceder a los contenidos que presenta el tema "Deporte".</p>	

Flujo Central:		
El caso de prueba se inicia si se accede al tema Deporte:		
<ol style="list-style-type: none"> 1. Se da un clic sobre Deporte 2. Se muestra un submenú con los siguientes temas: <ol style="list-style-type: none"> a. Juegos deportivos b. Más destacados 3. Se da clic sobre Juegos deportivos. Se muestra en pantalla contenido, imágenes y dos videos. Ver casos de pruebas <i>Mostrar pantalla de imágenes</i> y <i>Mostrar pantalla de video</i>. 4. Se da clic sobre Más destacados. Se muestra en pantalla el contenido e imágenes. Ver caso de prueba <i>Mostrar pantalla de imágenes</i> 		
Condiciones de ejecución:		
El usuario navegue en la multimedia		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar submenú del tema al dar clic sobre Deporte	(2) No mostrar submenú sin seleccionar Deporte	1 Se muestra el submenú del tema 2 No se muestra el submenú
(3) Mostrar contenido al dar clic sobre Juegos deportivos	(4) No mostrar el contenido sin seleccionar Juegos deportivos	3 Se muestra en la pantalla el contenido de Juegos deportivos. 4 No se muestra el contenido del tema
(5) Mostrar contenido al dar clic sobre Más destacados.	(6) No mostrar el contenido sin seleccionar Más destacados	5. Se muestra en la pantalla el contenido de Más destacados. 6 No se muestra el contenido del tema
CPR19: Mostrar submenú del tema Vida universitaria		
Aspectos Generales		
Requisito: R18	Aprobado: No	
Responsable: Yennia Rivero Aguila		
Descripción General:		
El caso de prueba permite verificar si se puede acceder a los contenidos que presenta el tema “Vida		

universitaria “		
Flujo Central:		
El caso de prueba se inicia si se accede al tema Vida universitaria:		
<ol style="list-style-type: none"> 1. Se da un clic sobre Vida universitaria 2. Se muestra un submenú con los siguientes temas: <ol style="list-style-type: none"> a. Elecciones b. Recreación c. Desde las facultades 3. De nuevo se da clic sobre Vida universitaria. Se muestra en pantalla el contenido. 4. Si se selecciona una de las pantallas Elecciones, Recreación o Desde las facultades. Se muestra en pantalla el contenido e imágenes. Ver caso de prueba <i>Mostrar pantalla de imágenes</i> 		
Condiciones de ejecución:		
El usuario navegue en la multimedia		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
1 Mostrar submenú del tema al dar clic sobre Vida universitaria	(2) No mostrar el submenú sin seleccionar el tema Vida universitario	1 Se muestra el submenú del tema
(3) Mostrar contenido al dar clic sobre Vida universitaria	(4) No mostrar el contenido sin seleccionar la pantalla Vida universitaria	3 Se muestra en la pantalla el contenido de Vida universitaria 4 No se muestra el contenido del tema
(5)Mostrar contenido al dar clic sobre Elecciones.	(6) No mostrar el contenido sin seleccionar la pantalla Elecciones	5 Se muestra en la pantalla el contenido de Elecciones. 6 No se muestra el contenido del tema
(7) Mostrar contenido al dar clic sobre Recreación.	(8) No mostrar el contenido sin seleccionar Recreación	7. Se muestra en la pantalla el contenido de Recreación. 8 No se muestra el contenido del tema

(9) Mostrar contenido al dar clic sobre Desde las facultades.	(10) No mostrar el contenido sin seleccionar Desde las facultades	9 Se muestra en la pantalla el contenido de Desde las facultades. 10 No se muestra el contenido del tema
---	---	---

CPR20: Mostrar contenido del tema Visitantes

Aspectos Generales		
Requisito: R19	Aprobado: Si	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Visitantes		
Flujo Central: El caso de prueba se inicia al dar clic sobre el tema Visitantes se pudo acceder a la pantalla <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Visitantes 2. Se muestran imágenes en pantalla. Ver caso de prueba <i>Mostrar pantalla de imágenes.</i> 		
Condiciones de ejecución: El usuario pudo acceder a la pantalla		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado
(1) Mostrar contenido al dar clic sobre el tema Visitantes		1. Se muestra en la pantalla el contenido de Visitantes

CPR21: Mostrar contenido del tema Artículos

Aspectos Generales		
Requisito: R20	Aprobado: No	
Responsable: Yennia Rivero Aguila		
Descripción General: El caso de prueba permite verificar si se muestra el contenido del tema Artículos		

<p>Flujo Central:</p> <p>El caso de prueba se inicia cuando al dar clic sobre el tema “Artículos” se pudo acceder a la pantalla:</p> <ol style="list-style-type: none"> 1. Se muestra en pantalla el contenido de Artículos 2. Se muestran imágenes en pantalla. Ver caso de prueba <i>Mostrar pantalla de imágenes.</i> 3. Se muestra en la parte izquierda superior de la pantalla los títulos de los artículos publicado. Aparece una barra de desplazamiento Ver caso de prueba <i>Permitir desplazar el scroll</i> 4. Se muestra en la parte izquierda inferior de la pantalla el contenido de los artículos publicado. Aparece una barra de desplazamiento Ver caso de prueba <i>Permitir desplazar el scroll</i> 5. Si se selecciona un título se muestra en la parte izquierda inferior el contenido del título seleccionado 		
<p>Condiciones de ejecución:</p> <p>El usuario navegue en la multimedia</p>		
<p>Clases de Equivalencia</p>		
<p>Clases Válidas</p>	<p>Clases Inválidas</p>	<p>Resultado Esperado</p>
<p>(1) Mostrar contenido al dar clic sobre Artículos.</p>	<p>(2) No mostrar el contenido sin seleccionar el tema Artículos</p>	<p>1. Se muestra en la pantalla el contenido de Artículos. 2 No se muestra el contenido del tema</p>
<p>(3)Mostrar contenido al dar clic sobre el título “La ciencia de todos los días”</p>	<p>(4) No mostrar el contenido sin seleccionar “La ciencia de todos los días”</p>	<p>3. Se muestra en la parte inferior de la pantalla el contenido de artículo. 4 No se muestra el contenido del artículo</p>
<p>(5) Mostrar contenido al dar clic sobre el título “Una vez mas, la UCI”</p>	<p>(6) No mostrar el contenido sin seleccionar “Una vez mas, la UCI”</p>	<p>5. Se muestra en la parte inferior de la pantalla el contenido de artículo. 6 No se muestra el contenido del artículo</p>
<p>(7) Mostrar contenido al dar clic sobre el título” Las banderitas ya están</p>	<p>(8) No mostrar el contenido sin seleccionar” Las banderitas ya están</p>	<p>7. Se muestra en la parte inferior de la pantalla el contenido de artículo. 8 No se muestra el contenido del</p>

pegadas.”	pegadas.”	artículo
(9) Mostrar contenido al dar clic sobre el título” Aires astrales para una gala.”	(10) No mostrar el contenido sin seleccionar”Aires astrales para una gala.”	9 Se muestra en la parte inferior de la pantalla el contenido de artículo.
(11) Mostrar contenido al dar clic sobre el título “INFODANZ, la informática y la danza.”	(12) No mostrar el contenido sin seleccionar “INFODANZ, la informática y la danza.”	11 Se muestra en la parte inferior de la pantalla el contenido de artículo. 12 No se muestra el contenido del artículo
(13) Mostrar contenido al dar clic sobre el título “I Activo de Residencia de la UCI.”	(14) No mostrar el contenido al dar clic fuera del título “I Activo de Residencia de la UCI.”	13 Se muestra en la parte inferior de la pantalla el contenido de artículo. 14 No se muestra el contenido del artículo
(15) Mostrar contenido del artículo al dar clic sobre el título “Visita partidista en la UCI.”	(16) No mostrar el contenido al dar clic fuera del título “Visita partidista en la UCI.”	15 Se muestra en la parte inferior de la pantalla el contenido de artículo. 16 No se muestra el contenido del artículo
(17). Mostrar contenido del artículo al dar clic sobre el título “Por qué son importantes los jueves.”	(18) No mostrar el contenido al dar clic fuera del título “Por qué son importantes los jueves.”	17. Se muestra en la parte inferior de la pantalla el contenido de artículo. 18 No se muestra el contenido del artículo
(19) Mostrar contenido del artículo al dar clic sobre el título “Relevante actuación de los ajedrecistas de la UCI.”	(20) No mostrar el contenido al dar clic fuera del título “Relevante actuación de los ajedrecistas de la UCI.”	19 Se muestra en la parte inferior de la pantalla el contenido de artículo. 20 No se muestra el contenido del artículo
(21) Mostrar contenido del artículo al dar clic sobre el título “Se cierran las cortinas de la fiesta deportiva.”	(22) No mostrar el contenido al dar clic fuera del título “Se cierran las cortinas de la fiesta deportiva.”	21 Se muestra en la parte inferior de la pantalla el contenido de artículo. 22 No se muestra el contenido del artículo
(23). Mostrar contenido del	(24) No mostrar el contenido	23 Se muestra en la parte inferior de

artículo al dar clic sobre el título “Rinden homenaje a Camilo la FEU y la UJC de la UCI.”	al dar clic fuera del título “Rinden homenaje a Camilo la FEU y la UJC de la UCI.”	la pantalla el contenido de artículo. 24 No se muestra el contenido del artículo
(25). Mostrar contenido del artículo al dar clic sobre el título “Ya se formó la rueda, ya se formó...”	(26) No mostrar el contenido al dar clic fuera del título “Ya se formó la rueda, ya se formó...”	25 Se muestra en la parte inferior de la pantalla el contenido de artículo. 26 No se muestra el contenido del artículo
(27) Mostrar contenido del artículo al dar clic sobre el título “Campeones en casa”	(28) No mostrar el contenido al dar clic fuera del título “Campeones en casa”	27 Se muestra en la parte inferior de la pantalla el contenido de artículo. 28 No se muestra el contenido del artículo
(29) Mostrar contenido del artículo al dar clic sobre el título “Celebrada reunión de inicio de curso en la UCI.”	(30) No mostrar el contenido al dar clic fuera del título “Celebrada reunión de inicio de curso en la UCI.”	29 Se muestra en la parte inferior de la pantalla el contenido de artículo. 30 No se muestra el contenido del artículo
(31). Mostrar contenido del artículo al dar clic sobre el título “Ya estamos en congreso.”	(32) No mostrar el contenido al dar clic fuera del título “Ya estamos en congreso.”	31 Se muestra en la parte inferior de la pantalla el contenido de artículo. 32 No se muestra el contenido del artículo
(33). Mostrar contenido del artículo al dar clic sobre el título “Esta es la industria del futuro para todos”, afirmó el presidente de Malasia en la UCI.”	(34) No mostrar el contenido al dar clic fuera del título “Esta es la industria del futuro para todos”, afirmó el presidente de Malasia en la UCI.”	33. Se muestra en la parte inferior de la pantalla el contenido de artículo. 34 No se muestra el contenido del artículo
(35). Mostrar contenido del artículo al dar clic sobre el título “Nuestro primer oro”	(36) No mostrar el contenido al dar clic fuera del título “Nuestro primer oro”	35. Se muestra en la parte inferior de la pantalla el contenido de artículo. 36 No se muestra el contenido del artículo

(37). Mostrar contenido del artículo al dar clic sobre el título “Celebrado acto de la amistad entre los pueblos de China y Cuba.”	(38) No mostrar el contenido al dar clic fuera del título “Celebrado acto de la amistad entre los pueblos de China y Cuba.”	37. Se muestra en la parte inferior de la pantalla el contenido de artículo. 38 No se muestra el contenido del artículo
(39). Mostrar contenido del artículo al dar clic sobre el título “El antes y el después de nuestra FEU.”	(40) No mostrar el contenido al dar clic fuera del título “El antes y el después de nuestra FEU.”	39 Se muestra en la parte inferior de la pantalla el contenido de artículo. 40 No se muestra el contenido del artículo
(41) Mostrar contenido del artículo al dar clic sobre el título “Concluyó la visita integral de la UJC y la FEU.”	(42) No mostrar el contenido al dar clic fuera del título “Concluyó la visita integral de la UJC y la FEU.”	41. Se muestra en la parte inferior de la pantalla el contenido de artículo. 42 No se muestra el contenido del artículo

2.5.3 Pruebas de Solidez

Objetivos

Este tipo de prueba permite verificar si el contenido de información que muestra la multimedia está redactado correctamente. Para dar cumplimiento al objetivo se diseñó el siguiente caso de prueba.

2.5.3.1 Casos de Pruebas de Solidez

CPR22: Verificar ortografía

Aspectos Generales	
Requisito: RNF2	Aprobado: No
Responsable: Yennia Rivero Aguila	
Propósito: El caso de prueba permite verificar que el contenido de la multimedia no presente errores ortográficos.	

<p>Prueba: Verificar que: Se muestra en la pantalla un contenido de información sin errores ortográficos.</p>
<p>Resultado esperado: La multimedia no presentó errores.</p>

2.5.4 Pruebas de Soportabilidad

Objetivos:

Este tipo de prueba permite verificar si la multimedia reconoce los elementos de hardware y software mencionado en la descripción de los requisitos no funcionales (RNF3, RNF4, FNF7, RNF8). Para este tipo de prueba se diseñaron los siguientes casos de pruebas.

2.5.4.1 Caso de prueba de Soportabilidad

CPR23: Verificar elementos

Aspectos Generales	
Requisito: RNF7, RNF8	Aprobado: Si
Responsable: Yennia Rivero Aguila	
<p>Propósito: Verificar los elementos de software y hardware.</p>	
<p>Prueba: Verificar si la multimedia corre en los Sistemas Operativos Windows 98, Windows 2000 y Windows NT y Windows XP. Para estas pruebas debe estar almacenada la multimedia en un CD. La multimedia debe ejecutarse en una PC con un microprocesador Pentium II 300MHz o superior. La memoria es de 64MB de RAM, lector de CD ROM, tarjeta de sonido, tarjeta de video 16 bits de colores y un Mouse. Para estas pruebas debe estar almacenada la multimedia en un CD.</p>	
<p>Resultado esperado: La multimedia se ejecutó correctamente</p>	

CPR24: Cargar a pantalla completa

Aspectos Generales	
Requisito: RNF3	Aprobado: No
Propósito: El caso de prueba permite verificar que la multimedia se muestre a pantalla completa	
Prueba: El caso de prueba inicia cuando: <ol style="list-style-type: none"> 1. Se carga el ejecutable de la multimedia 2. La multimedia se abre a pantalla completa 	
Resultado esperado: Se muestra a pantalla completa	

CPR25: Butear desde torre CD

Aspectos Generales	
Requisito: RNF4	Aprobado: No
Responsable: Yennia Rivero Aguila	
Descripción General: El caso de prueba permite verificar que la multimedia sea buteable	
Flujo Central: El caso de prueba inicia cuando se inserta el CD de la multimedia en la torre CD La multimedia butea desde la torre de CD	
Resultado esperado: Es buteable	

2.5.5 Pruebas de Resistencia (Stress)

Objetivo:

Este tipo de prueba permite verificar que la aplicación garantice una velocidad de procesamiento lo más alta posible (RNF6)

2.5.5.1 Caso de prueba de Resistencia

CPR26: Verificar multimedia ante situaciones anormales

Aspectos Generales	
Requisito: RNF6	Aprobado: Si
Responsable: Yennia Rivero Aguila	
Propósito: Evaluar el comportamiento del sistema bajo condiciones anormales.	
Prueba: Evaluar si la multimedia trabaja con memoria insuficiente, el tiempo de respuesta de un elemento seleccionado.	
Resultado esperado: La multimedia trabaja bajo condiciones anormales	

2.6 Conclusiones

En el capítulo se expuso una breve descripción del producto a probar y de los 11 temas en los que se divide el módulo de la multimedia FEU.

Para la aplicación de las pruebas se expuso un proceso de prueba que consta de cuatro etapas en las que se organizan de manera sistemática las pruebas. En la actividad de planificación se elaboró un plan de prueba especificando la estrategia de prueba a utilizar y los tipos de pruebas que se aplicarían en cada nivel establecido.

Para el diseño de las pruebas se tuvieron presente los siguientes puntos claves:

- Un buen caso de prueba es aquél que tiene una alta probabilidad de encontrar un defecto no descubierto, no aquél en el que el programa funciona correctamente.
- Cada caso de prueba debe definir el resultado de salida esperado.
- Al generar casos de prueba, se deben incluir tanto datos de entrada válidos y esperados como no válidos e inesperados.

CAPÍTULO 3: RESULTADOS DE LAS PRUEBAS

En un proceso de prueba se debe documentar además de la planificación y el diseño de los casos de pruebas los resultados reales de las pruebas aplicadas al software. Esto da la posibilidad al programador de corregir los errores encontrados en cada parte de la aplicación. En este capítulo se documentan los defectos detectados y los resultados que se obtuvieron al aplicar los casos de pruebas

3.1 Estructura de la Bitácora de prueba

En este caso se describen los defectos encontrados en cada nivel de prueba establecido donde se creó una plantilla con la cantidad de errores detectados en cada tipo de prueba aplicada a la multimedia.

Para la construcción de la bitácora se siguieron los siguientes pasos: [TERUEL, 2001]

- Configuración exacta del software a probar.
- Responsable de la ejecución
- Una tabla con el siguiente formato:

Tabla 8. Reporte de defectos y dificultades detectadas

Elemento	No	No conformidad	Localización	Importancia	Recomendaciones
<Nombre de Elemento>	< 1>	<Descripción de la No Conformidad>	<Descripción de Aspecto correspondiente>	<X>	<X>

- La descripción de la no conformidad y su localización deben ser lo más precisa y completa posible.

El nivel de importancia de los defectos se puede catalogar como:

- Crítico - denota una función que causa un término anormal o una falla general, o cuando un cambio en un área de la aplicación causa un problema en otra parte.
 - Severo - una función no actúa como fue requerido o diseñado, o un objeto de interfaz no trabaja como se muestra.
 - Advertencia - la función trabaja, pero no como esperaba, o no se ajusta a las normas y convenciones.
 - Cosmético - no crítico para el funcionamiento de sistema: palabras con mala ortografía, formateo incorrecto, mensajes de error vagos o confusos o advertencias
- Una tabla resumen con la cantidad de errores detectados en cada prueba

3.1.1 Registro de defectos y dificultades detectado a Nivel de Unidad

Las pruebas de interfaz se ejecutaron en una PC con Sistema Operativo Windows XP, con 256 Mb de RAM y microprocesador Pentium 4. Para la ejecución de los casos de pruebas de este tipo se contó con un probador.

Tabla 9. Registro de defectos y dificultades detectado a Nivel de Unidad

Elemento	No	No conformidad	Localización	Importancia	Recomendaciones
Botón de sonido	1	Aparece un botón de sonido en la pantalla de imágenes que no debe aparecer.	-El usuario da clic sobre el tema Deporte - Se muestra un submenú - El usuario da clic sobre Juegos ínter facultades - Aparecen imágenes en la parte derecha-superior de la pantalla El usuario da clic sobre una de las imágenes.	advertencia	<i>El diseño de la pantalla de imágenes debe mantenerse al seleccionar una imagen en todas las pantallas.</i>

			- Se visualiza una pantalla con la imagen seleccionada. Ver anexo 7		
Botón de sonido	2	Aparece un botón de sonido en la pantalla de imágenes que no debe aparecer.	-El usuario da clic sobre el tema Deporte - Se muestra un submenú. - El usuario da clic sobre Mas destacados - Aparecen imágenes en la parte derecha-superior de la pantalla El usuario da clic sobre una de las imágenes. - Se visualiza una pantalla con la imagen seleccionada. Ver anexo 9	advertencia	<i>El diseño de la pantalla de imágenes debe mantenerse al seleccionar una imagen en todas las pantallas.</i>

3.1.2 Registro de defectos y dificultades detectado a Nivel de Sistema

Las pruebas funcionales se ejecutaron en una PC con Sistema Operativo Windows XP, con 256Mb de RAM y microprocesador Pentium 4. Para la ejecución de los casos de pruebas de este tipo se contó con un probador.

Tabla 10. Registro de defectos y dificultades detectado en las pruebas funcionales

Elemento	No	No conformidad	Localización	Importancia	Recomendaciones
Botón de Sonido	1	No se desactiva la música de fondo al oprimir el botón correspondiente a	-Abrir multimedia -Aparece un la presentación de la multimedia. -Se muestra una pantalla	severo	<i>Permitir que el usuario decida si desea oír la música o no en cualquier</i>

		sonido.	con los temas de la multimedia y las opciones salida, sonido y siguiente. -El usuario selecciona el botón sonido para desactivar la música de fondo.		<i>pantallas de la multimedia</i>
Acceder a la pantalla Batalla de ideas	2	No se puede acceder a la pantalla Batalla de ideas.	-El usuario da un clic sobre el tema Batalla de ideas - Se muestra un submenú -El usuario vuelve a dar clic sobre el tema Batalla de ideas. Ver anexo 5	<i>advertencia</i>	<i>Permitir que el usuario no solo acceda a las pantallas a través del botón siguiente, sino también por el menú.</i>
Contenido de video	3	No se mostrar el contenido del video	- El usuario da un clic sobre el tema Batalla de ideas - Se muestra un submenú -El usuario vuelve a dar clic sobre el tema Batalla de ideas. - Se muestra en la parte derecha-inferior de la pantalla un video -El usuario da clic sobre el video -Se muestra una pantalla con el video seleccionado. Ver anexo 6	<i>severo</i>	
Música de fondo	4	Se desactiva la música de fondo	-El usuario da clic sobre el tema Deporte		<i>El sistema no debe desactivar la música</i>

		del tema Deporte cuando se cierra la pantalla de imágenes.	<ul style="list-style-type: none"> - Se muestra un submenú - El usuario da clic sobre el tema Juegos Inter facultades - Se muestran imágenes en la parte derecha- superior de la pantalla. -El usuario da clic sobre una de las imágenes - Se muestra una pantalla con la imagen seleccionada. -El usuario cierra la pantalla. 	severo	<i>cuando se muestra la pantalla de imágenes, el diseño debe mantenerse en todas las pantallas de la multimedia.</i>
Acceder a la pantalla Vida universitaria	5	En ocasiones no se puede acceder al contenido de la pantalla Vida universitaria.	<ul style="list-style-type: none"> -El usuario da clic sobre Vida universitaria -Aparece un submenú -El usuario da clic sobre Vida universitaria. Ver anexo 8 	advertencia	<i>Permitir que el usuario no solo acceda a las pantallas a través del botón siguiente, sino también por el menú del módulo</i>
Artículo “La ciencia de todos los días “	6	Se muestra el contenido del artículo al dar clic fuera del título “La ciencia de todos los días “.	<ul style="list-style-type: none"> -El usuario da clic sobre el tema Artículos. - Se muestra en la parte izquierda –superior los títulos de los artículos -El usuario da clic fuera del título “La ciencia de todos los días”. - Se muestra en la parte inferior de la pantalla el contenido del artículo 	severo	<i>Para acceder a la información que contiene los artículos, hay que dar clic sobre el título</i>

Artículo "Una vez mas, la UCI"	7	Se muestra el contenido del artículo al dar clic fuera del título "Una vez mas, la UCI ".	-El usuario da clic fuera del título "Una vez mas, la UCI " - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo "Las banderitas ya están pegadas"	8	Se muestra el contenido del artículo al dar clic fuera del título "Las banderitas ya están pegadas. "	-El usuario da clic fuera del título "Las banderitas ya están pegadas. " - Se muestra en la parte inferior de la pantalla el contenido del artículo.	severo	
Artículo "Aires astrales para una gala"	9	Se muestra el contenido del artículo al dar clic fuera del título "Aires astrales para una gala"	-El usuario da clic fuera del título "Aires astrales para una gala" - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo "INFODANZ, la informática y la danza"	10	Se muestra el contenido del artículo al dar clic fuera del título "INFODANZ, la informática y la danza ".	-El usuario da clic fuera del título "INFODANZ, la informática y la danza ". - Se muestra en la parte inferior de la pantalla el contenido del artículo.	severo	

Artículo "I Activo de Residencia en la UCI"	11	Se muestra el contenido del artículo al dar clic fuera del título "I Activo de Residencia en la UCI"	-El usuario da clic fuera del título "I Activo de Residencia en la UCI" - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo "Visita partidista en la UCI"	12	Se muestra el contenido del artículo al dar clic fuera del título "Visita partidista en la UCI"	-El usuario da clic fuera del título "Visita partidista en la UCI" - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo "Por qué son importantes los jueves"	13	Se muestra el contenido del artículo al dar clic fuera del título "Por que son importantes los jueves".	-El usuario da clic fuera del título "Por qué son importantes los jueves". - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo "Relevante actuación de los ajedrecistas de la UCI. "	14	Se muestra el contenido del artículo al dar clic fuera del título "Relevante actuación de los ajedrecistas de la UCI. "	-El usuario da clic fuera del título "Relevante actuación de los ajedrecistas de la UCI. " - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	

Artículo “Se cierran las cortinas de la fiesta deportiva	15	Se muestra el contenido del artículo al dar clic fuera del título “Se cierran las cortinas de la fiesta deportiva “.	-El usuario da clic fuera del título “Se cierran las cortinas de la fiesta deportiva“ - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo “Rinden homenaje a Camilo la FEU y la UJC de la UCI. “	16	Se muestra el contenido del artículo al dar clic fuera del título “Rinden homenaje a Camilo la FEU y la UJC de la UCI. “	-El usuario da clic fuera del título “Rinden homenaje a Camilo la FEU y la UJC de la UCI. “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo “Ya se formó la rueda, ya se formó...”	17	Se muestra el contenido del artículo al dar clic fuera del título “Ya se formó la rueda, ya se formó...”	-El usuario da clic fuera del título “Ya se formó la rueda, ya se formó...” - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo “Campeones en casa “.	18	Se muestra el contenido del artículo al dar clic fuera del título “Campeones en casa “.	-El usuario da clic fuera del título “Campeones en casa “. - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo “Celebrada reunión de	19	Se muestra el contenido del artículo al dar clic	-El usuario da clic fuera del título “Celebrada reunión de inicio de curso en la UCI. “	severo	

inicio de curso en la UCI. “		fuera del título “Celebrada reunión de inicio de curso en la UCI. “	- Se muestra en la parte inferior de la pantalla el contenido del artículo		
Artículo “Ya estamos en congreso “.	20	Se muestra el contenido del artículo al dar clic fuera del título “Ya estamos en congreso “.	-El usuario da clic fuera del título “Ya estamos en congreso “. - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo “Esta es la industria del futuro para todos”, afirmó el presidente de Malasia en la UCI. “	21	Se muestra el contenido del artículo al dar clic fuera del título “Esta es la industria del futuro para todos”, afirmo el presidente de Malasia en la UCI. “	-El usuario da clic fuera del título “Esta es la industria del futuro para todos”, afirmó el presidente de Malasia en la UCI. “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo “Nuestro primer oro. “	22	Se muestra el contenido del artículo al dar clic fuera del título “Nuestro primer oro. “	-El usuario da clic fuera del titulo “Nuestro primer oro. “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	<i>severo</i>	
Artículo “Celebrado acto de la amistad entre los pueblos	23	Se muestra el contenido del artículo al dar clic fuera del título “Celebrado acto de	-El usuario da clic fuera del titulo “Celebrado acto de la amistad entre los pueblos de China y Cuba. “ - Se muestra en la parte	<i>severo</i>	

de China y Cuba. “		la amistad entre los pueblos de China y Cuba. “	inferior de la pantalla el contenido del artículo		
Artículo “El antes y el después de nuestra FEU. “	24	Se muestra el contenido del artículo al dar clic fuera del título “El antes y el después de nuestra FEU. “	-El usuario da clic fuera del título “El antes y el después de nuestra FEU. “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	
Artículo “Concluyó la visita integral de la UJC y la FEU“	25	Se muestra el contenido del artículo al dar clic fuera del título “Concluyó la visita integral de la UJC y la FEU “	-El usuario da clic fuera del título “Concluyó la visita integral de la UJC y la FEU “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	severo	

Las pruebas de solidez se ejecutaron en una PC con Sistema Operativo Windows XP, con 256 Mb de RAM y microprocesador Pentium 4. Para la ejecución de los casos de pruebas de este tipo se contó con un probador.

Tabla 11. Registro de defectos y dificultades detectado en las pruebas de solidez

Elemento	No	No conformidad	Localización	Importancia	Recomendaciones
Ortografía	1	Existen errores ortográficos en el artículo “ Aires astrales para una gala. “	El usuario da clic sobre el título “ Aires astrales para una gala. “ - Se muestra en la parte inferior de la pantalla el contenido del artículo	cosmético	<i>Tener mas cuidado a la hora de redactar la información</i>

			- Aparece en el ultimo párrafo la palabra " aritmico "; debería ser la palabra arritmico.		
Ortografía	2	Existen errores ortográficos en el artículo " Ya estamos en congreso "	El usuario da clic sobre el título " Ya estamos en congreso - Se muestra en la parte inferior de la pantalla el contenido del artículo - Aparece en el tercer párrafo la palabra " Sancti Spíritus "; sin tilde.	cosméti co	

Para la ejecución de los casos de pruebas de soportabilidad se contó con un probador.

Tabla 10. Registro de defectos y dificultades detectado en las pruebas soportabilidad

Elemento	No	No conformidad	Localización	Importancia	Recomendaciones
Cargar multimedia	1	La multimedia tiene dificultad al butear por la torre CD.	Insertar CD.	<i>Crítico</i>	
Cargar pantalla completa	2	La multimedia no se ejecuta en a pantalla completa.	Abrir el ejecutable de la multimedia.	<i>severo</i>	<i>Permitir que se muestre la multimedia a pantalla completa</i>

3.2 Resultados Generales de las pruebas

Dentro de las dificultades detectadas, la mayoría de los errores que encontrados fueron de tipo severo debido a que presentaban problemas de interfaz y funcionamiento. Para los casos de pruebas diseñados a los temas Quiénes somos, Docencia, Visitantes, Producción no se detectaron errores, arrojando resultados satisfactorios.

De la prueba aplicada a nivel de unidad solo se detectaron 2 errores mientras que las aplicadas a nivel de sistemas arrojaron la cantidad de 29 errores, por lo que el 6% de los errores detectados fueron a nivel de unidad y el 94% a nivel de sistema

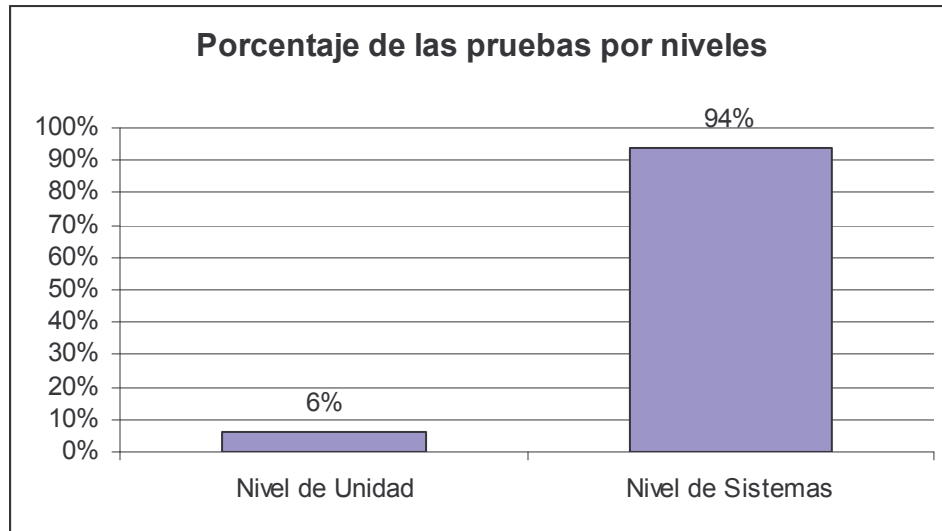


Figura 1. Porcentaje de las pruebas por niveles

Los errores encontrados por cada tipo de prueba se muestran en la tabla 11

Tabla 11. Informe del proceso de prueba

Producto: Multimedia FEU		Fecha: 30/05 /07	
Versión: 1.0			
Revisado por: Yennia Rivero Águila			
TIPOS DE PRUEBA	Nº ERRORES	Nº CASOS DE PRUEBAS	Nº SUGERENCIAS
Pruebas de Interfaz	2	1	2
Pruebas de Solidez	2	1	1
Pruebas Funcionales	25	20	6
Pruebas de Soportabilidad	2	3	1
Pruebas de Resistencia	0	1	0
Totales	31	26	10

Se diseñaron un total de 26 casos de pruebas para verificar los requerimientos funcionales y no funcionales de la multimedia. Del total de casos de pruebas ejecutados solo en 17 no se encontraron errores, esto indica que sólo un 65 % de las pruebas fueron satisfactorias y un 35% insatisfactorias.

Se registraron un total de 31 fallas de las pruebas ejecutadas, de las cuales 25 corresponden a la categoría “severo”, 2 a la categoría “cosmético” y 4 a la categoría de “advertencia”. En cuanto a la categoría “crítico” la aplicación presentó una sola falla al sistema en general. Por lo tanto, se concluye, que se detectaron un 3% de tipo “crítico”, 78 % de tipo “severo”, 13 % de tipo “advertencia” y 6% de tipo “cosmético” aproximadamente.

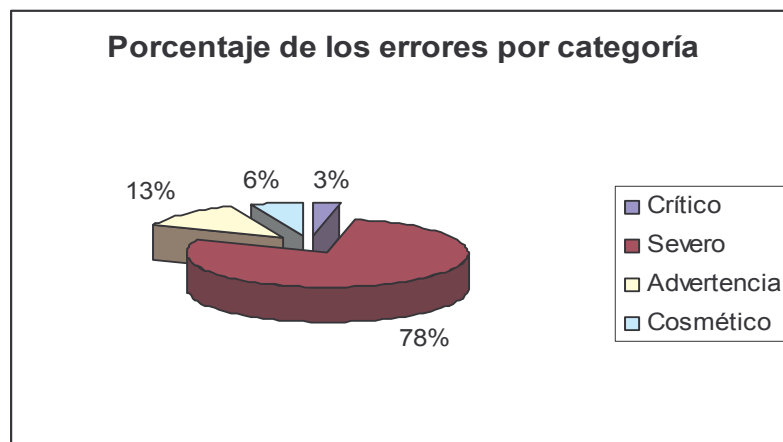


Figura 2. Porcentajes de las pruebas por categoría

A la hora de hacer una comparación entre los resultados esperados y los que se obtuvieron se puede decir que la multimedia realiza la mayor parte de las funciones.

Hay algunas observaciones que hacer con respecto a la multimedia, como por ejemplo: la multimedia antes de cerrarse debería mostrar una ventana que le permita al usuario decidir si desea o no salir de la aplicación y la pantalla de video debería ocultar las opciones salida, sonido y siguiente como sucede con el diseño de la pantalla de imágenes.

3.3 Costos de las pruebas

Para la realización de las pruebas se contó con un trabajador en un plazo de 5 meses de trabajo el cual es el responsable de todas las actividades que comprende el proceso de prueba.

Total de Horas trabajadas

En un mes se trabajo todas las semanas de lunes a viernes 8 horas laborables, para un total de 20 días laborables.

CHM= CHD * CDM CHM: Cantidad de horas trabajadas por mes CHD: Cantidad de horas por día

CHM= 8 * 20 CDM: Cantidad de días trabajados en el mes

CHM= 160 horas-mes

Para un total de 800 horas laborables en los 5 meses de trabajo

Gastos de salario

Para la realización de las pruebas se contó con un trabajador desempeñando todos los roles del proceso de prueba, en un plazo de 5 meses de trabajo con salario de \$100 MN.

ST= SM * CM ST: Salario total ST: Salario al mes CM: Cantidad de meses trabajados

ST= 100 * 5

ST= \$ 500

Gasto de recursos materiales

Para el trabajo realizado se contó con los siguientes recursos materiales (el precio es en moneda nacional MN)

Recurso	Cantidad	Precio
CD-R	1	\$10
Libreta	1	\$5
Lapicero	1	\$7
Hojas de impresión	110	\$100

Para un total de \$ 122 en gastos materiales

Gasto de electricidad

Para la ejecución de las pruebas se utilizó una PC en tiempo completo, teniendo en cuenta que una PC gasta 150 watts/hora y 1 watts/hora equivale a 0.0016 (centavos) se tiene que:

En un día laborar de 8 horas se gasta 1200 watts que equivale a \$1.92 MN y al mes se gasta 18000 que equivale a \$ 28.8

GE= GM * 5 GE: Gasto de electricidad en los meses trabajados GM: Gasta de electricidad al mes

GE= \$ 28.8 * 5

GE= \$ 144

Total de gastos: \$766

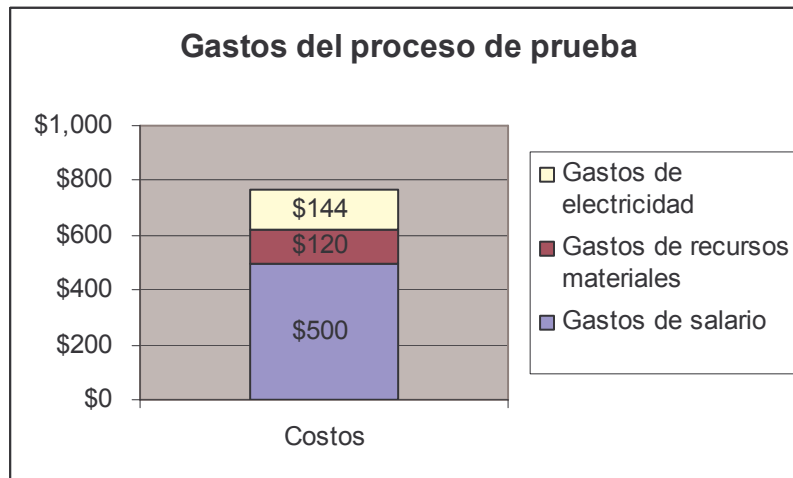


Figura 3. Costos de las pruebas

Teniendo en cuenta que la multimedia no es un producto desarrollado para la comercialización, no es válido mencionar entonces beneficios económicos. Más, reiterando que esta propuesta es resultado de la necesidad de lograr la calidad en estos productos se hizo una estimación de los gastos monetarios asociados para determinar si es factibles aplicar el proceso de prueba y como la Universidad asume los costos directos e indirectos en los que se incurra, se puede decir que es factible el proceso de prueba.

3.4 Conclusiones

Este Capítulo cumple con la condición de ser concluyente para el proceso de pruebas que se ha desarrollado. Al ejecutar los casos de pruebas se detectó que gran parte de la multimedia cumplía con las funciones especificadas y que solo el 35% presentan fallas.

Se puede plantear que la multimedia FEU no está lista debido a los errores detectados durante este proceso, pero se prevee que puedan solucionarse antes de ser entregado el producto. Para los costos de las pruebas se han detallado los recursos materiales necesarios y los recursos humanos implicados.

CONCLUSIONES

Las pruebas de software están íntimamente ligadas con la calidad del producto software desarrollado. Con esto, aquellas personas que no tiene una cultura de calidad causan retrasos en las entregas de los proyectos, porque si los incluyen, el costo del proyecto se elevará y seguramente estos individuos saldrán más baratos, pero los errores cometidos lo son mucho más.

Al finalizar el proceso de prueba propuesto con el objetivo de aplicarlo en la multimedia FEU no podemos demostrar que el producto esta al 100% libre de errores pero sí plantear que la mayor parte de las dificultades fueran detectas antes de ser entregado.

Como resultado de la planificación y aplicación de las pruebas se puede concluir que:

- Con el estudio realizado referente a la aplicación de las pruebas del software se obtuvieron conocimientos en cuanto a estrategias y tipos de pruebas existentes.
- Al elaborar de un Plan de Prueba y definir de una estrategia se logró organizar y documentar las pruebas aplicadas a la multimedia FEU.
- Se obtuvieron casos de pruebas a través del método de Caja Negra cubriendo la mayor parte de los caminos que puede recorrer la aplicación en su ejecución
- Con los errores detectados en el proceso de prueba y el análisis de los resultados obtenidos se llegó a la conclusión que el 65% de los requerimientos de la multimedia cumplen la funcionalidad requerida.

RECOMENDACIONES

Como resumen de todo lo expuesto hasta el momento y luego de aplicar el proceso de prueba propuesto para la detección de errores a la multimedia FEU, se listan a continuación una serie de recomendaciones tanto en el ámbito teórico como el práctico para mejorar la calidad en productos multimedia:

- Aplicar pruebas de regresión al producto con vistas a erradicar totalmente los errores.
- Aplicar pruebas por el método de Caja Blanca para verificar el funcionamiento interno del producto debido a que las pruebas de la Caja Negra, sin tener en cuenta cómo sea de completa, puede pasar por alto otros tipos de errores.
- Aplicar el proceso propuesto a productos de este tipo, pero agregando el nivel de integración a aquellos que presenten más de un módulo.
- Registrar la documentación obtenida a la asignatura de Ingeniería de Software.



REFERENCIAS BIBLIOGRAFICAS

ARENAS, L. G. "ESTRATEGIA DE CALIDAD - PARQUESOFT." 2006. Disponible en: <http://ParqueSoft.com>.

BOEHM, B. *Software Engineering Economics*. 1981. 37 p.

CROSBY, P. *Quality is Free: the Art of Making Quality Certain*. USA: 1979.

EQUIZABAL, D, FLORES, E, RODAS, I. *Metodología de Prueba en Ingeniería de Software*. Universidad Centroamericana José Simeón Cañas, 2006.

ESPINOZA, D. R. V. E. Y. E. G. V. *Estándares de calidad para pruebas de software*. UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS, 2005.

GONZALEZ, C. *Plan de Pruebas Exitoso*. Disponible en: <http://www.americaxxi.cl/modules.php?name=News&file=article&sid=20> .

GUTIÉRREZ, J. J. *Generación de pruebas de sistema a partir de la especificación funcional*. Escuela Técnica Superior de Ingeniería Informática, Universidad de Sevilla, 2005.

GUTIERREZ, Y. G. *Temática Ingeniería de Software Chile*: Universidad Católica del Maule, 2003. Disponible en: http://www.eici.ucm.cl/Academicos/ygomez/descargas/In_Sw_TUC/apuntes/ingenieriasoftwaretematica.doc .

IEEE, Standard 610, *Computer Dictionary*. Nueva York: 1990

MENDEZ, C. (2004). "*Metodo de Pruebas Orientada a Objetos para el Ciclo de Vida Completo*." Disponible en: <http://www.ambyssoft.com/essays/flootSpanish.html>.

MYERS, G. *The art of Software Tests*, Wiley, 1979

PRESSMAN, R. S. *Ingeniería del Software: Un enfoque Práctico*. 2002.

PRESSMAN, R. S. *Software Engineering: A Practitioner's Approach*. European Adaptation: 2000.



QUESADA, J. A. L. *Tema 8. Gestión de Calidad*. Universidad de Murcia, Disponible en: http://dis.um.es/~lopezquesada/documentos/FIS_0405/Tema8Calidad.ppt#258.4, Definición de Calidad

SACHA, , Universidad de las Ciencias Informáticas, 2006

SANTOS, F. Ó. G. R. Y. C. B. *Pruebas del Software*. Escuela Superior de Informática de Ciudad Real, 2005. Disponible en: http://alarcos.inf-cr.uclm.es/per/fgarcia/isoftware/doc/tema7_2xh.pdf.

SORIANO, A. *Pruebas del Software*, 2005 Disponible en: http://carolina.terna.net/ingsw3/datos/Tipos_Prueba.pdf .

VÁZQUEZ, P.E, M. N. M. G., GARCÍA F.P. "*Verificación con XM*"L. Universidad de Salamanca, 2001.

Varios autores. "Pruebas de sistemas a entornos multimedias.", 2007 Disponible en: <http://www.telefonica.es/sociedaddelainformacion/pdf/publicaciones/imagenio/capitulos/imageniocap8.pdf>.

RIOJA, A. M. *PRUEBA DE APLICACIONES*. Madrid: Ciclo Formativo de Grado Superior en Desarrollo de Aplicaciones Informáticas, 2005. Disponible en: <http://boj.cnice.mecd.es/~amora2/adai/ADAI09.pdf>.

TERUEL., A. *Prueba repetibles y mantenibles*. Venezuela: Universidad Simón Bolívar, Disponible en: <http://www ldc.usb.ve/~teruel/ci4713/clases2001/pruebasRep.html#bitacora> .

BIBLIOGRAFIA

ANNA. C GRIMÁN, M. P., LUIS. E MENDOZA. *Estrategia de Pruebas para Software OO que garantiza Requerimientos No Funcionales*. Universidad Simón Bolívar, 2006.

IRINA, B REYES, IRINA N TORRES. *Las pruebas de software, su aplicación al Config. CASE*. INSTITUTO SUPERIOR POLITÉCNICO JOSÉ ANTONIO ECHEVERRÍA, 2003

FERNÁNDEZ, L. *Generación de casos de prueba a partir de especificaciones UML* [página Web]. Madrid. Disponible en: <http://www.esi.uem.es/~plaraber/Investigacion/JICS2003.pdf>.

FERNANDO ADRIAN GOMEZ, FERNANDO JAVIER LAGE. "La revalorización del hipermedia en las agendas didácticas universitarias", 2002. Disponible en: <http://ism.dei.uc.pt/ribie/docfiles/txt2003729138La%20revalorizaci%C3%B3n%20del%20hipermedia.pdf>

JAVIER J. GUTIÉRREZ, M. J. E., ARTURO H. TORRES, MANUEL MEJÍAS Y JESÚS TORRES. *Hacia una propuesta de pruebas tempranas del Sistema*. Escuela Técnica Superior de Ingeniería Informática, Universidad de Sevilla, 2006.

LAUREYRO, V. V."Testing en WEB". Programa Consultoria Internacional. Disponiblen en: http://download.microsoft.com/download/7/3/3/73359d0c-ac21-4609-8d15-4e627634b40/Testing_en_web.pdf

LUIS E. MENDOZA, MARIA A. PEREZ y ANNA C. GRIMAN. "Prototipo de Modelo Sistémico de Calidad (MOSCA) del Software ". Universidad Simón Bolívar, 2004. Disponible en: http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_26.pdf

MORENO, G. A. "Plan de Pruebas (ejemplo)". Granada: Universidad de Granada, 2004
Disponible en: <http://lsi.ugr.es/~arroyo/inndoc/inicio.php>

PABLO, F. M. J. Y. G. J. *Sistemas de Programas*. Venezuela: [Consultado el: 25 de abril de 2007]. Disponible en: <http://www ldc.usb.ve/~teruel/ci3711/test1/index.html>

SALAZAR, R. CH. "Verificación y Validación del Software". Tecnológico de Monterey, 2005
Disponible en: <http://cb.mty.itesm.mx/cursos/TC3008/Presentaciones/Modulo%203%20-%20Pruebas%20de%20SW.pdf>

SALVADOR, O. G. "Diseño y ejecución de pruebas de software". Universidad de Guadalajara. Disponible en: http://148.202.148.5/cursos/cc321/fundamentos/unidad6/tema6_1.html

RAMÍREZ, J. *Métodos de pruebas del software* [pagina Web]. Madrid: Universidad Politécnica de Madrid. Disponible en: <http://lml.ls.fi.upm.es/~jramirez/ed2/>

RIVERA, O. S. M. *INVESTIGACIÓN SOBRE MODELOS DE PRUEBAS (TESTING) INGENIERÍA DE SOFTWARE* Monterrey: Instituto Tecnológico y de estudios Superiores de Monterrey. Disponible en: http://homepages.mty.itesm.mx/al1031749/Prueba_Testing.doc.

ROMERO, S. A. "Implementación de una arquitectura de tres modelos para crear multimedias educativas". Cuba: Universidad de Cienfuegos, 2005. Disponible en: <http://www.nonio.uminho.pt/challenges/05comunicacoes/Tema11/09VicenteZaldivar.pdf>

TAPIA, R. G. S. *CALIDAD EN EL DESARROLLO DE SOFTWARE EN EL GOBIERNO DEL ESTADO DE TAMAULIPAS Y ALTERNATIVAS DE MEJORA*. Universidad Autónoma de Tamaulipas, 2003.

USAOLA, D. M. P. *Pruebas del Software*. Ciudad Real: Universidad de Castilla-La Mancha, Disponible en: <http://alarcos.inf-cr.uclm.es/>.

V., M. J. R. *Planeación de Pruebas Funcionales: Aspectos relevantes* Cali: GreenSQA - ParqueSoft, [Consultado el: 11 de febrero de 2007]. Disponible en: <http://ParqueSoft.com>.

VALVERDE, J. C. "Acerca de las estrategias de prueba del software." Venezuela: Universidad Nacional de Trujillo, 2003 Disponible en: <http://www.seccperu.org/files/EstrategiasPruebaSw.pdf>.

VEGA, J.A, F. "Pruebas de Caja Blanca.", Tecnológico de Monterey, 1999. Disponible en: http://www.rogeliodavila.com/tcs/TCS%20Notes%20JAVega/Parte_16_TestWhite.ppt

GOÑI'S, A. "Tema 3.Evaluación / Pruebas del Software". Universidad del País Vasco, 1999. Disponible en: <http://siul02.si.ehu.es/~alfredo/iso/Tema3.pdf>

GLOSARIO

Aseguramiento de Calidad: (*Quality Assurance*, ISO 8402, 1994) Todas las actividades planificadas y sistemáticas necesarias para aportar la confianza suficiente en que un producto o servicio cumplirá con unos requisitos dados de calidad.

Defecto: Cualquier requerimiento, elemento de diseño o de implementación que si no es cambiado, causará un diseño, implementación, prueba, uso, o mantenimiento inapropiado del producto

Facilidad de prueba (*Testability*, ISO 9126) subcaracterística de mantenimiento, que indica la capacidad del software para permitir que sea validado tras ser modificado

Fallo: Un fallo es la incapacidad de un sistema o de alguno de sus componentes para realizar las funciones requeridas dentro de los requisitos de rendimiento especificad”

Fase: Son los pasos en que se descomponen las metodologías. Cada fase puede o no estar subordinada a otra fase, pudiendo existir entre ellas relaciones de dependencia de inicio fin y paralelismo.

HW: Hardware

Inspección del código: Actividad de garantía de la calidad del software que es llevada a cabo por los ingenieros del software.

Interfaz: Un conjunto de operaciones que posee un nombre y que caracteriza el comportamiento de un elemento

Proceso: Una secuencia de pasos realizados con un propósito determinado (IEEE Std. 610). El conjunto de actividades, métodos y prácticas usadas en la producción y evolución del software (SEI CMM).

Requisito o Requerimiento: Una característica, propiedad o comportamiento que se desea para el sistema.

Sistema: Colección de unidades conectadas que se organiza para lograr un propósito. El sistema es el “modelo completo”.

SW: Software

Usuario: Persona que utiliza normalmente el software.

ANEXOS

Anexo 1: FORMATO DE VERIFICACIÓN DE DISPOSITIVOS PARA PRUEBAS DE SOPORTABILIDAD

LISTA DE VERIFICACIÓN DE DISPOSITIVOS

I. INFORMACIÓN DEL DISPOSITIVO:

Dispositivo: _____	
Descripción: _____	
Procesador: _____	Memoria: _____
Espacio en Disco: _____	S.O: _____

II. RESULTADOS DE LA VERIFICACIÓN:

	SI	NO
El dispositivo cumple con los requisitos:.....	<input type="checkbox"/>	<input type="checkbox"/>
El Hardware esta conforme:.....	<input type="checkbox"/>	<input type="checkbox"/>
El Sistema Operativo esta conforme:.....	<input type="checkbox"/>	<input type="checkbox"/>
El espacio en disco disponible es suficiente:.....	<input type="checkbox"/>	<input type="checkbox"/>
Tarjeta de red conforme:.....	<input type="checkbox"/>	<input type="checkbox"/>
Tarjeta de sonido conforme:.....	<input type="checkbox"/>	<input type="checkbox"/>

III. OBSERVACIONES:

Verificador : _____

Firma: _____

Fecha : _____

Anexo 2: Diseño de Casos de pruebas para pruebas funcionales:

Para el diseño de todos los casos de prueba se debe tener lo siguiente (presentado en la tabla)

CPR<Nº>: Nombre descriptivo del caso de prueba.

- Requisito: Código del requerimiento
- Aprobado: Si el caso de prueba presentó o no errores al ser ejecutada
- Responsable: Persona que realiza la prueba
- Descripción General: objetivo del caso de prueba.
- Flujo Central: Pasos a ejecutar la prueba
- Condiciones de ejecución: Prerrequisitos imprescindibles para que se pueda ejecutar correctamente el Caso de Prueba
- Clases de Equivalencia: Estados válidos y no válidos para las condiciones de entrada.

Ejemplos de clases de equivalencia:

Botones

- seleccionado
- sin seleccionar

-Resultado Esperado: Resultado Esperado de la Prueba

Tabla 4: Diseño de Caso de prueba para los requerimientos funcionales

Aspectos Generales		
Requisito: R1	Aprobado:	
Responsable:		
Descripción General:		
Flujo Central:		
Condiciones de ejecución:		
Clases de Equivalencia		
Clases Válidas	Clases Inválidas	Resultado Esperado

Anexo 3: Diseño de Casos de pruebas para pruebas no funcionales

Para el diseño de todos los casos de prueba se debe tener lo siguiente (presentado en la tabla)

CPR<Nº>: Nombre descriptivo del caso de prueba.

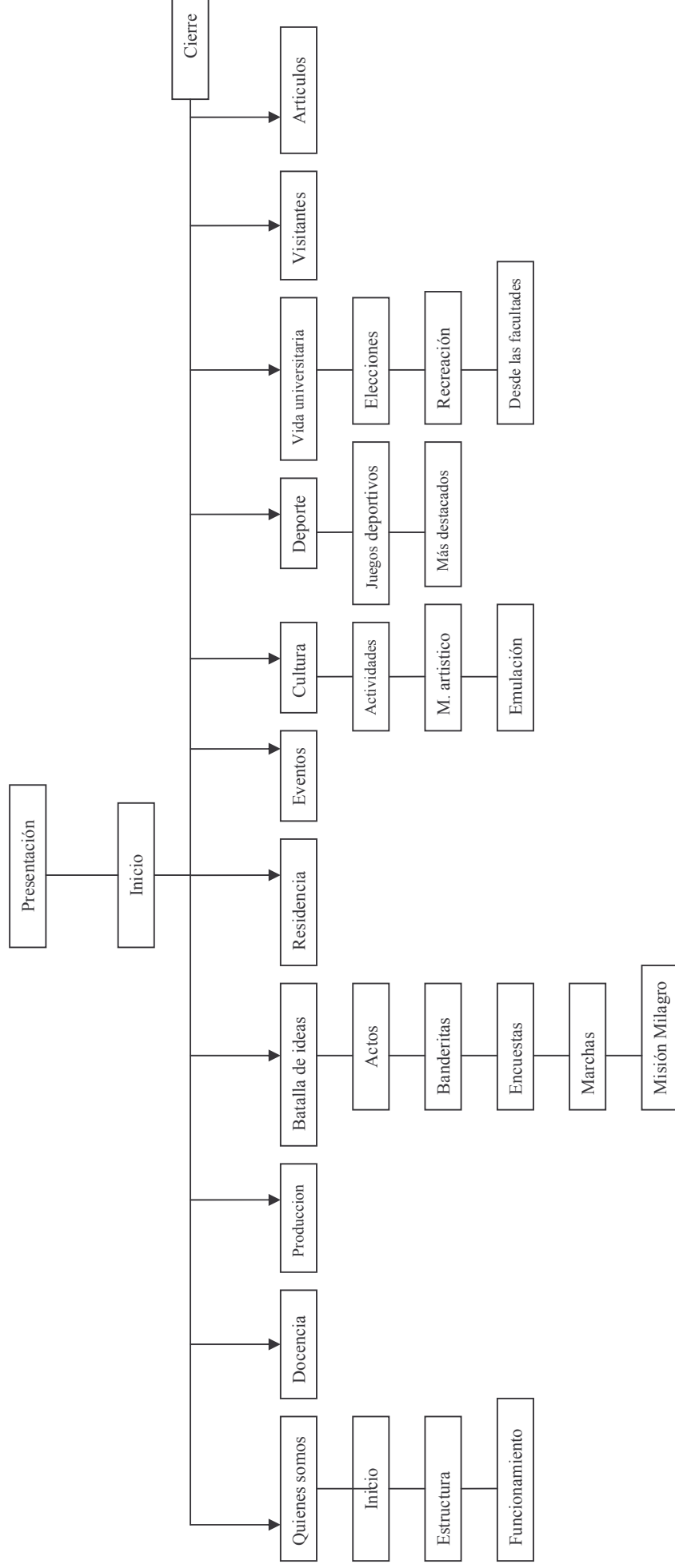
- Requisito: Código del requerimiento
- Aprobado: Si el caso de prueba presentó o no errores al ser ejecutada
- Responsable: Persona que realiza la prueba
- Propósito: objetivo del caso de prueba.
- Prueba: Descripción a probar
- Resultado Esperado: Este deberá ser que se cumpla la prueba

Tabla 3: Diseño de Caso de prueba para los requerimientos no funcionales

Aspectos Generales	
Requisito: RNF1	Aprobado:
Responsable:	
Propósito:	
Prueba:	
Resultado esperado:	



Anexos 4: Mapa de Navegación



Anexo 5: Pantalla Batalla de ideas



Anexo 6: Video de la pantalla Batalla de ideas



Anexo 7: Diseño de la pantalla de imágenes



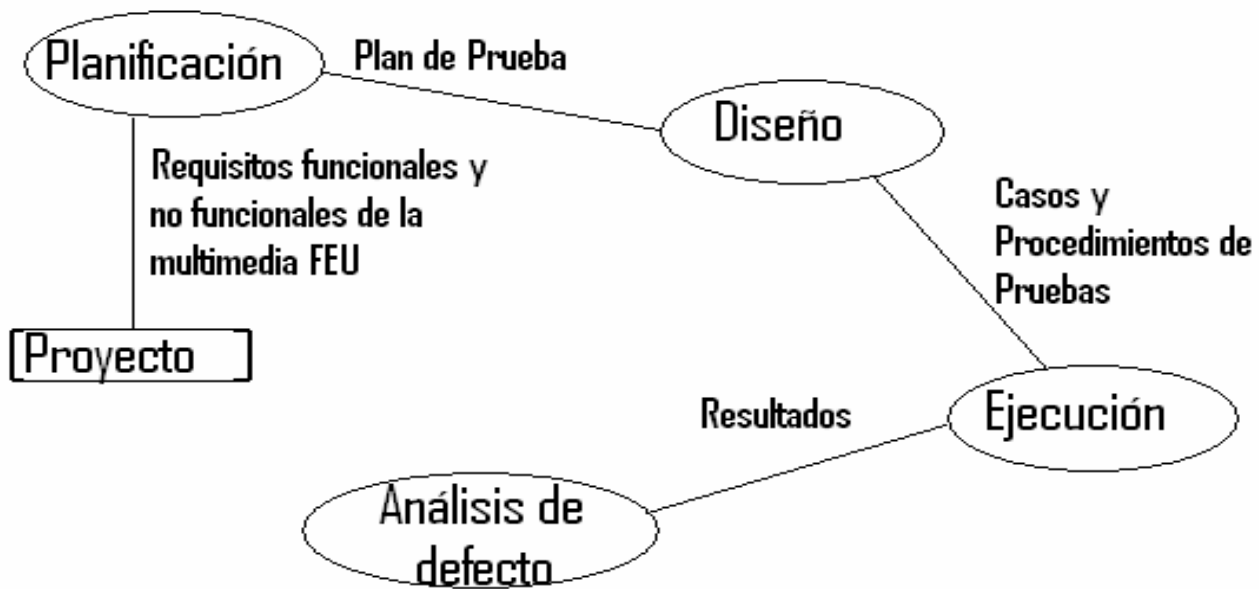
Anexo 8: Pantalla Vida universitaria



Anexo 9: Diseño de la Pantalla de imágenes



Anexo 10: Etapas del Proceso de Prueba



Anexo 11. Planificación del Proceso de Prueba de la Multimedia FEU

