

**Universidad de las Ciencias Informáticas**

Facultad 8

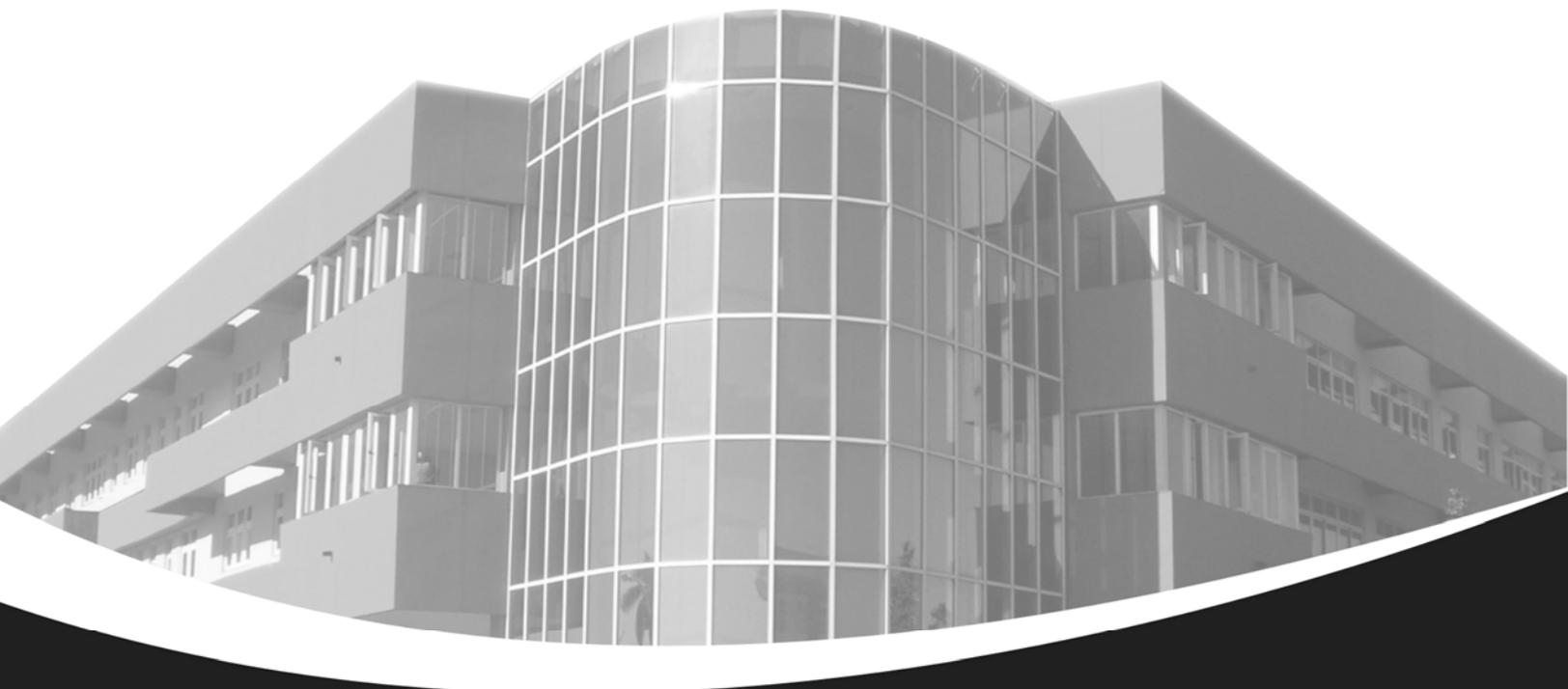


**Proceso de pruebas al producto multimedia  
“Organopónico”**

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

**Autora:** Lilian Alvarez Almanza

**Tutora:** Ing. Giselle Medina Martínez



**Declaración de autoría**

Yo, Lilian Alvarez Almanza, declaro ser autora de la presente tesis y reconozco 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 \_\_\_\_ días del mes de \_\_\_\_\_ del año \_\_\_\_\_.

Lilian Alvarez Almanza

Ing. Giselle Medina Martínez

---

---

## **Datos de contacto**

Tutora: Ing. Giselle Medina Martínez

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](mailto:gisellem@uci.cu)

## Agradecimientos

*A mis padres por su apoyo en todo momento...*

*A mis compañeros de aula por brindarme siempre su ayuda...*

*A mis amistades por confiar en mí y saber darme aliento para continuar...*

*A mi decano Alcides Cabrera por confiar en mí...*

*A mi tutora Giselle Medina por dedicarme parte de su tiempo y ayudarme...*

*A todas las personas que han hecho realidad el sueño de crear una Universidad como la que hoy  
tenemos...*

**Dedicatoria**

*A mi mamá Maritza Almanza por haberme apoyado siempre...*

*A mi papá Alfredo Alvarez por quererme tanto y saber comprenderme...*

*A mi tío Rafael Almanza por todos sus consejos...*

*A mi tía Blanca Escobar por haberme cuidado desde pequeña...*

*A mis amigas inseparables de estos cinco años de la Universidad: Yinimary Ortega e Isis Margarita Blanco por estar a mi lado en los momentos buenos y malos y haberme ayudado a soportar la distancia...*

### Resumen

En la actualidad, la informática ha alcanzado un gran auge en todo el mundo y como parte de esta la producción de software. Debido a este auge que va en ascenso se ha vuelto imprescindible la calidad de los productos software para su posterior comercialización y competencia en el mercado mundial. Un elemento crítico para definir la calidad de un software y de gran importancia a la hora de detectar sus defectos es: Las Pruebas de Software. Nuestro país y en específico la Universidad de las Ciencias Informáticas (UCI), no está ajena a esta realidad, pero aún así en la mayoría de los casos se crean productos sin haber tenido en cuenta un proceso de pruebas que permita detectar los defectos del software. Tal es el caso de la multimedia “Organopónico”, que fue creada en la UCI con el fin de ser exportada a Venezuela. Este trabajo se centra en establecer un proceso de pruebas al producto multimedia “Organopónico”, de forma tal que garantice detectar los defectos del mismo, para de esta forma ayudar a mejorar la calidad del producto. En aras de lograr este objetivo se planificaron y diseñaron pruebas de funcionalidad, de portabilidad y diseño, se crearon varios casos de prueba para cada una de ellas y luego de ejecutadas las pruebas se registraron los defectos encontrados, a partir de los cuales se hizo un análisis de los resultados.

**Palabras Claves:** Calidad de Software, Proceso de Pruebas, Plan de Prueba, Diseño de Prueba, Casos de Prueba, Registro de Defectos o Registro de No Conformidades.

## Índice

<b>AGRADECIMIENTOS .....</b>	<b>I</b>
<b>DEDICATORIA.....</b>	<b>II</b>
<b>RESUMEN.....</b>	<b>III</b>
<b>INTRODUCCIÓN.....</b>	<b>1</b>
<b>CAPÍTULO 1 - FUNDAMENTACIÓN TEÓRICA .....</b>	<b>4</b>
INTRODUCCIÓN.....	4
1.1 ¿QUÉ ES LA CALIDAD DE SOFTWARE? .....	4
1.2 PRUEBAS DE SOFTWARE EN EL CONTROL DE LA CALIDAD.....	6
1.2.1 CONTROL DE LA CALIDAD .....	6
1.2.2 PRUEBAS DE SOFTWARE .....	7
1.3 PRINCIPIOS DE LAS PRUEBAS.....	10
1.4 PROCESO DE PRUEBAS .....	12
1.4.1 ACTIVIDADES DEL PROCESO DE PRUEBAS.....	12
1.4.2 CICLO DEL PROCESO DE PRUEBAS .....	14
1.5 ESTRATEGIA DE PRUEBA .....	15
1.5.1 MÉTODOS DE PRUEBA .....	16
1.5.2 NIVELES DE PRUEBA .....	22
1.6 ANTECEDENTES DE PRUEBAS A MULTIMEDIA .....	28
1.7 CONCLUSIONES.....	30
<b>CAPÍTULO 2 – DESCRIPCIÓN DEL PROCESO DE PRUEBAS .....</b>	<b>31</b>
INTRODUCCIÓN.....	31
2.1 DESCRIPCIÓN DE LA MULTIMEDIA.....	31
2.2 ADAPTACIÓN DEL PROCESO DE PRUEBAS A LA MULTIMEDIA “ORGANOPÓNICO” .....	33
2.3 APLICACIÓN DEL PROCESO DE PRUEBAS .....	37
2.3.1 PLAN DE PRUEBA.....	37
2.3.1.1 <i>Objetivo del proceso de pruebas</i> .....	38
2.3.1.2 <i>Plan de tiempos de las actividades</i> .....	38
2.3.1.3 <i>Reparto de responsabilidades X rol</i> .....	38
2.3.1.4 <i>Recursos</i> .....	39

2.3.1.5 Estrategia de prueba .....	40
2.3.1.6 Entregables.....	41
2.3.1.7 Requerimientos a probar .....	41
2.3.2 DISEÑO DE PRUEBA .....	44
2.3.2.1 Pruebas del Nivel de Unidad .....	44
2.3.2.2 Pruebas del Nivel de Sistema.....	44
2.3.3 CASOS DE PRUEBA .....	46
2.3.3.1 Casos de Prueba Funcionales del Nivel de Unidad .....	46
2.3.3.2 Casos de Prueba Funcionales del Nivel de Sistema.....	83
2.3.3.3 Casos de Prueba de Portabilidad.....	88
2.3.3.4 Casos de Prueba de Diseño.....	89
2.4 CONCLUSIONES.....	91
<b>CAPÍTULO 3 – RESULTADOS DEL PROCESO DE PRUEBAS .....</b>	<b>92</b>
INTRODUCCIÓN.....	92
3.1 REGISTRO DE NO CONFORMIDADES .....	92
3.2 ANÁLISIS DE RESULTADOS .....	99
3.2.1 RESULTADOS DEL NIVEL DE UNIDAD .....	100
3.2.2 RESULTADOS DEL NIVEL DE SISTEMA.....	101
3.2.3 RESULTADOS GENERALES .....	101
3.3 CÁLCULO DE COSTOS .....	104
3.4 CONCLUSIONES.....	107
<b>CONCLUSIONES.....</b>	<b>108</b>
<b>RECOMENDACIONES .....</b>	<b>109</b>
<b>REFERENCIAS BIBLIOGRÁFICAS.....</b>	<b>110</b>
<b>BIBLIOGRAFÍA.....</b>	<b>112</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>114</b>

## Índice de Tablas

Tabla 1: Plantilla de Casos de Prueba mediante el Método de Caja Negra. ....	35
Tabla 2: Plantilla de Casos de Prueba que no utilizan el Método de Caja Negra ni el de Caja Blanca. ....	36
Tabla 3: Plantilla de Registro de Defectos. ....	37
Tabla 4: CP2: Mostrar el significado de: canteros, cosechas, abono y estiércoles. ....	46
Tabla 5: CP3: Mostrar el contenido que se aborda en: Materiales y Herramientas. ....	49
Tabla 6: CP4: Permitir relacionar el nombre de los materiales y herramientas con la imagen correspondiente. ....	50
Tabla 7: CP5: Mostrar el contenido que se aborda en: Preparación del Terreno. ....	51
Tabla 8: CP7: Mostrar las orientaciones de la actividad: Complete la definición. ....	52
Tabla 9: CP8: Permitir realizar la actividad: Complete la definición. ....	53
Tabla 10: CP10: Mostrar las orientaciones de la actividad: Pareo. ....	54
Tabla 11: CP11: Permitir realizar la actividad: Pareo. ....	56
Tabla 12: CP13: Mostrar el contenido que se aborda en: Compostero (Definición e importancia). ....	57
Tabla 13: CP14: Mostrar el significado de: abonera, estiércoles, abono orgánico, compost. ....	58
Tabla 14: CP16: Mostrar el significado de: demarca, material orgánico, capas. ....	61
Tabla 15: CP19: Mostrar el contenido que se aborda en: Canteros (Definición e Importancia). ....	63
Tabla 16: CP22: Mostrar el contenido que se aborda en: Semillero (Definición e Importancia). ....	64
Tabla 17: CP26: Mostrar las orientaciones de la actividad: Elabora virtualmente un compostero. ....	66
Tabla 18: CP27: Permitir realizar la actividad: Elabora virtualmente un compostero. ....	67
Tabla 19: CP29: Mostrar las orientaciones de la actividad: Sopa de letras. ....	69
Tabla 20: CP30: Permitir realizar la actividad: Sopa de letras. ....	70
Tabla 21: CP32: Mostrar el contenido que se aborda en: Transplante del semillero al cantero. ....	71
Tabla 22: CP33: Mostrar el contenido que se aborda en: Sistema de riego. ....	73
Tabla 23: CP35: Mostrar el contenido que se aborda en: Tipos de plagas. ....	74
Tabla 24: CP36: Mostrar el contenido que se aborda en: Control de plagas. ....	75
Tabla 25: CP37: Mostrar la actividad: Auto Evaluación. ....	77
Tabla 26: CP38: Permitir realizar la actividad: Auto Evaluación. ....	79
Tabla 27: CP39: Mostrar el contenido que se aborda en: Recolección y clasificación de la cosecha. ....	80
Tabla 28: CP40: Mostrar el contenido que se aborda en: Embalaje y clasificación. ....	81
Tabla 29: CP41: Mostrar presentación particular de la aplicación. ....	83
Tabla 30: CP43: Permitir activar o desactivar el sonido desde cualquier pantalla de la aplicación. ....	84
Tabla 31: CP44: Permitir salir de la aplicación desde cualquier pantalla. ....	85

## Índice de Tablas

---

Tabla 32: CP46: Permitir navegar dentro de cada módulo mediante los botones: siguiente y anterior.....	87
Tabla 33: CP48: Permitir que la multimedia sea multiplataforma.....	88
Tabla 34: CP49: Mostrar una interfaz amigable.....	89
Tabla 35: No Conformidades del Nivel de Unidad.....	92
Tabla 36: No Conformidades del Nivel de Sistema.....	98
Tabla 37: Resultados del Nivel de Unidad.....	100
Tabla 38: Resultados del Nivel de Sistema.....	101
Tabla 39: Defectos más comunes encontrados en la multimedia “Organopónico”.....	103
Tabla 40: Gastos de Recursos Materiales.....	105
Tabla 41: Cálculo del total de horas trabajadas.....	105

## Índice de Figuras

Figure 1: Actividades de Control de Calidad .....	7
Figura 2: Actividades y salidas estándar del proceso de pruebas .....	13
Figure 3: Ciclo del proceso de pruebas [LÓPEZ 2005].....	15
Figure 4: Plan de tiempo de las actividades del proceso de pruebas.....	38
Figure 5: Defectos encontrados en la multimedia “Organopónico” por niveles. ....	102
Figure 6: Defectos encontrados en la multimedia “Organopónico” por pruebas. ....	102

### **Introducción**

A nivel mundial, la informática y las comunicaciones han logrado alcanzar una gran relevancia, debido al desarrollo de nuevas tecnologías que caracterizó las últimas décadas del siglo XX y que mantienen un ritmo en ascenso durante este inicio del siglo XXI. La informática se ha convertido en una base imprescindible para el trabajo del resto de las ciencias y de cualquier actividad del mundo actual, abarcando todas las esferas de trabajo, como por ejemplo: económicas, financieras, científicas, culturales y deportivas, lo que ha requerido el desarrollo de software que den respuestas a este constante movimiento de información.

En la elaboración de cualquier producto software, un aspecto imprescindible lo constituye la calidad del mismo, es decir, que este sea capaz de satisfacer los requerimientos especificados por el cliente, además de cumplir con los requisitos propios del producto. Un elemento crítico para definir la calidad de un software y de gran importancia a la hora de detectar sus defectos es: Las Pruebas de Software.

Actualmente, en la Universidad de las Ciencias Informáticas (UCI), la inexistencia de un proceso de pruebas provoca, en la mayoría de los casos, que el cliente no se sienta satisfecho con el producto final ya que el mismo no cumple con las características funcionales y requerimientos que él solicita. Esto puede traer consigo que el cliente rechace el producto, causando así que el mismo no se ajuste al tiempo y costo, estimados para su realización, además afecta en gran medida el prestigio y la confianza depositada en la Universidad y puede provocar la pérdida del cliente.

En el caso específico de la multimedia "Organopónico", que fue creada en la UCI con el fin de ser exportada a Venezuela para apoyar la idea del gobierno venezolano de incentivar el desarrollo endógeno, con la finalidad de que las comunidades rurales y urbanas a través de cultivos organopónicos sean capaces de autoabastecerse y colocar productos agrícolas en el mercado nacional, no se le aplicó un proceso de pruebas que garantizara su correcta funcionalidad y adecuación a los requerimientos especificados por el cliente, es decir, su calidad. Por lo que se declara como problema científico de esta investigación: ¿Cómo se podría establecer un proceso de pruebas al producto multimedia "Organopónico"?

Para lograr resolver este problema científico, se plantea como objeto de estudio el proceso de pruebas a productos de software, y como campo de acción, el proceso de pruebas al producto multimedia “Organopónico”.

El objetivo general que se persigue con la realización de este trabajo es: Establecer un proceso de pruebas al producto multimedia “Organopónico”, de forma tal que garantice detectar los defectos del mismo, para de esta forma ayudar a mejorar la calidad del producto.

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

- Desarrollar un proceso de pruebas al producto multimedia “Organopónico”.
- Obtener, como resultado de la aplicación del proceso de pruebas, los defectos del producto multimedia “Organopónico”.

Para dar cumplimiento a estos objetivos se realizaron las siguientes tareas de la investigación:

- Investigar, sobre los antecedentes de la aplicación de pruebas a productos multimedia y los procesos de pruebas existentes, mediante búsquedas bibliográficas en internet y libros de Ingeniería de Software, y mediante la comunicación con los encargados de la calidad en centros de trabajos que producen multimedia.
- Adecuar un proceso de pruebas al producto multimedia “Organopónico”.
- Aplicar el proceso de pruebas al producto multimedia “Organopónico”.
- Documentar los defectos encontrados al producto multimedia “Organopónico”.
- Analizar los defectos encontrados.

El presente documento consta de tres capítulos. En el capítulo 1 se trata la fundamentación teórica del tema, obtenida como resultado de las investigaciones realizadas. En el capítulo 2 queda establecido el proceso de pruebas a la multimedia “Organopónico” y en el capítulo 3 se documentan los resultados obtenidos al aplicar el proceso de pruebas y se realiza un análisis de los mismos.

# Fundamentación Teórica

## Introducción

El proceso de concepción, elaboración y utilización de un producto software puede ser especialmente complejo, no solamente por las características del mismo sino además porque surge la necesidad de definir cómo obtener un software de calidad y cómo evaluarlo. Estos últimos son dos aspectos muy polémicos y que han sido motivo de investigación por parte de especialistas, ingenieros, investigadores y comercializadores de software, principalmente por el hecho de que el término “Calidad” es relativo y multidimensional.

A lo largo de este capítulo se pretende definir algunos conceptos necesarios para el desarrollo del trabajo, entre los más importantes se encuentran: qué es calidad de software, qué es una prueba, así como qué es un proceso de pruebas. También se abordará la definición de la estrategia de prueba, los métodos para realizarlas y los niveles de las mismas.

Se conocerán algunos antecedentes de lo que se ha realizado en materia de pruebas a multimedia, tanto a nivel internacional como nacional.

### 1.1 ¿Qué es la Calidad de Software?

Como un aspecto esencial para entender el alcance de este término es necesario diferenciar entre la calidad del *producto* software y la calidad del *proceso* de elaboración del mismo.

No obstante, las metas que se establezcan para la calidad del producto van a determinar las metas a establecer para la calidad del proceso de desarrollo, ya que la calidad del producto va a estar en función de la calidad del proceso de desarrollo.

Centrándose en lo que respecta a la calidad de un producto software se plantea la siguiente interrogante ¿Es realmente posible encontrar un conjunto de propiedades en un producto software que de una indicación de su calidad? Por lo que es necesario definir un único concepto de calidad, de la misma forma surge la necesidad de definir cómo medir el grado de calidad de un producto software, lo que se tratará a continuación.

Diferentes personalidades e instituciones han dado su opinión con respecto a la definición de calidad:

“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” [PRESSMAN 1993]

“El conjunto de características de una entidad que le confieren su aptitud para satisfacer las necesidades expresadas y las implícitas” [ISO 1995]

“Grado con el cual el cliente o usuario percibe que el software satisface sus expectativas” [IEEE 1990]

Luego de realizar un análisis a las definiciones de calidad antes expuestas, se puede definir como calidad de software: el conjunto de cualidades que lo caracterizan y responden a los requisitos implícitos y explícitos del producto.

Como criterios que puedan utilizarse para medir la calidad de un software se pueden proponer los siguientes:

- *Eficiencia*: Que exista concordancia entre la cantidad de código o tiempo de respuesta con los recursos que precisa un programa para desempeñar su función.
- *Flexibilidad*: El coste de modificación del producto cuando cambian sus especificaciones.

- *Corrección:* Hasta qué punto un programa cumple sus especificaciones y satisface los objetivos del usuario.
- *Mantenibilidad:* El coste de localizar y corregir defectos en un programa que aparecen durante su funcionamiento.
- *Portabilidad:* El coste de transportar o migrar un producto de una configuración hardware o entorno operativo a otro.
- *Usabilidad:* El coste y esfuerzo de aprender a manejar un producto, preparar la entrada de datos e interpretar la salida del mismo.
- *Integridad:* Hasta qué punto se controlan los accesos ilegales a programas o datos. Un programa que permite el acceso de personas no autorizadas a ciertos datos es poco íntegro.

## **1.2 Pruebas de Software en el Control de la Calidad**

Una evaluación o medición de la calidad sólo después de elaborado el producto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, u otra etapa de desarrollo del mismo, por lo que es imprescindible tener en cuenta tanto la obtención de la calidad como su control durante todas las etapas del ciclo de vida del software.

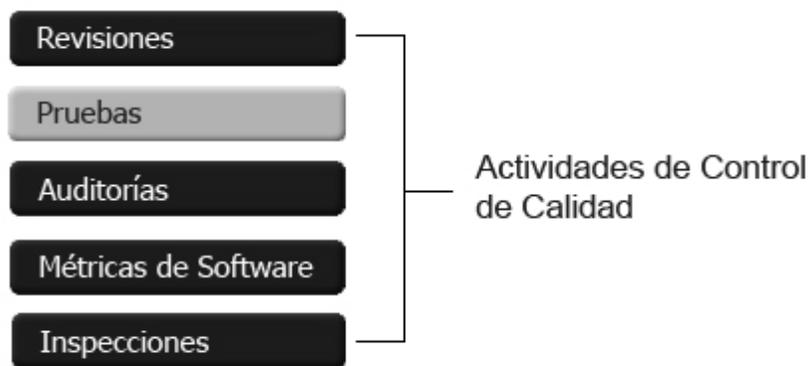
### **1.2.1 Control de la Calidad**

Son las técnicas y actividades de carácter operativo, utilizadas para satisfacer los requisitos relativos a la calidad, centradas en dos objetivos fundamentales: [CUEVA 1999]

- Mantener bajo control un proceso.
- Eliminar las causas de los defectos en las diferentes fases del ciclo de vida.

El objetivo de las actividades de control de calidad es evaluar la calidad de los productos desarrollados, es decir, comprobar si un producto posee o no una determinada característica de calidad en el grado requerido. Cuando un producto no posee una determinada característica de calidad se dice que tiene un defecto. Por lo tanto, se puede decir también que el objetivo del control de calidad es identificar defectos en el producto y corregirlos. [DE ANTONIO 1999]

Se pueden contemplar los siguientes tipos de actividades de control de calidad:



**Figure 1: Actividades de Control de Calidad**

### 1.2.2 Pruebas de Software

Hasta la fecha no se ha desarrollado ninguna teoría universalmente aceptada acerca de la prueba de software. Pero se puede decir que es un conjunto de aproximaciones metodológicas que facilitan y hacen más eficiente el proceso de pruebas.

Una de las características típicas del desarrollo de software basado en el ciclo de vida es la realización de controles periódicos, normalmente coincidiendo con los hitos del proyecto o la terminación de documentos. Estos controles pretenden una evaluación de la calidad de los productos generados (especificación de requisitos, documentos de diseño, etc.) para poder detectar posibles defectos cuanto antes. Sin embargo, todo sistema o aplicación, independientemente de estas revisiones, debe ser probado mediante su ejecución controlada antes de ser entregado al cliente.

Estas ejecuciones o ensayos de funcionamiento, posteriores a la terminación del código del software, se denominan habitualmente pruebas.

Los objetivos de las pruebas según Glen Myers [MYERS 1979] son los siguientes:

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

El coste de detección de los defectos suele ser mucho mayor que el coste de corrección de los mismos, y este es un punto en contra de las pruebas como técnica de control de calidad, ya que siempre es necesario un paso de diagnóstico hasta que se localiza la causa de los fallos. En otras actividades de control de calidad, por el contrario, como pueden ser las revisiones, se localizan directamente los defectos, no sus síntomas, por lo que se ahorra el proceso de diagnóstico. [DE ANTONIO 1999]

Algunas definiciones en relación a las pruebas:

- Pruebas:

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

“Proceso de ejecutar un programa con el fin de encontrar errores”. [MYERS 1979]

“Un conjunto de casos y procedimientos de prueba”. [IEEE 1990]

- Caso de prueba:

“Un conjunto de entradas, condiciones de ejecución y resultados esperados desarrollados para un objetivo particular como, por ejemplo, ejercitar un camino concreto de un programa o verificar el cumplimiento de un determinado requisito”. [IEEE 1990]

También se puede referir a la documentación en la que se describen las entradas, condiciones y salidas de un caso de prueba.

- Defecto:

“Un defecto en el software como por ejemplo, un proceso, una definición de datos o un paso de procesamiento incorrecto en un programa”. [IEEE 1990]

Se entiende por defecto la consecuencia de un error, o una desviación en el valor esperado para una cierta característica. Los defectos no tienen por qué afectar al funcionamiento del objeto defectuoso.

- Error:

Una acción humana que conduce a un resultado incorrecto. [FERNÁNDEZ-MEDINA 2006]

- Fallo:

La incapacidad de un sistema o de alguno de sus componentes para realizar las funciones requeridas dentro de los requisitos de rendimiento especificados. Este puede ser la manifestación de un defecto en el software. [FERNÁNDEZ-MEDINA 2006]

- Relación Error – Defecto – Fallo

El *Error* es una acción incorrecta cometida por el programador, se ve evidenciado en el software como un *Defecto* del mismo, dando lugar a que se produzca un *Fallo* en el sistema, ya que el mismo tiene un comportamiento no deseado.

- Fallas:

Las fallas son los defectos que aún no han sido detectados y eliminados cuando comienzan las pruebas. Algunas de estas fallas se convertirán en fallos si se detectan durante las pruebas o el uso del sistema. [DE ANTONIO 1999]

- Depuración:

Proceso en el que se localiza el defecto que es la causa de un fallo, se determina la forma de corregirlo, se evalúa el efecto de la corrección y se lleva a cabo la corrección. [DE ANTONIO 1999]

### **1.3 Principios de las Pruebas**

Antes de la aplicación de métodos para el diseño de casos de pruebas o para planificar un proceso o estrategia de prueba, un ingeniero del software deberá entender los principios básicos que guían las pruebas del software: [PRESSMAN 2005]

- *A todas las pruebas se les debería poder hacer un seguimiento hasta los requisitos del cliente.* El objetivo de las pruebas del software es descubrir errores. Se entiende que los defectos más graves (desde el punto de vista del cliente) son aquellos que impiden al programa cumplir sus requisitos.

- *Las pruebas deberían planificarse mucho antes de que empiecen.* La planificación de las pruebas puede empezar tan pronto como esté completo el modelo de requisitos. La definición detallada de los casos de prueba puede empezar tan pronto como el modelo de diseño se haya consolidado. Por tanto, se pueden planificar y diseñar todas las pruebas antes de generar ningún código.
- *El principio de Parapeto es aplicable a la prueba del software.* Dicho de manera sencilla, el principio de Parapeto implica que al 80% de todos los errores descubiertos durante las pruebas se les puede hacer un seguimiento hasta un 20% de todos los módulos del programa. El problema es aislar estos módulos sospechosos y probarlos concienzudamente.
- *Las pruebas deberían empezar por lo pequeño y progresar hacia lo grande.* Las primeras pruebas planeadas y ejecutadas se centran generalmente en módulos individuales del programa. A medida que avanzan las pruebas, desplazan su punto de mira en un intento de encontrar errores en grupos integrados de módulos y finalmente en el sistema entero.
- *No son posibles las pruebas exhaustivas.* El número de permutaciones de caminos, para incluso un programa de tamaño moderado, es excepcionalmente grande. Por este motivo, es imposible ejecutar todas las combinaciones de caminos durante las pruebas. Es posible, sin embargo, cubrir adecuadamente la lógica del programa y asegurarse de que se han aplicado todas las condiciones en el diseño a nivel de componente.
- *Para ser más eficaces, las pruebas deberían ser realizadas por un equipo independiente.* Por “más eficaces” se refiere a pruebas con la más alta probabilidad de encontrar errores (el objetivo principal de las pruebas). El ingeniero del software que creó el sistema no es el más adecuado para llevar a cabo las pruebas para el software.

## **1.4 Proceso de Pruebas**

El proceso de pruebas comienza con la generación de un plan de pruebas en base a la documentación sobre el proyecto y la documentación sobre el software a probar. A partir de dicho plan, se entra en detalles, diseñando pruebas específicas basándose en la documentación del software a probar. Una vez detalladas las pruebas (especificaciones de casos de prueba), se toma el software que se va a probar y se ejecutan los casos de prueba. En algunas situaciones, se puede tratar de reejecuciones de pruebas, por lo que es conveniente tener constancia de los defectos ya detectados aunque aún no corregidos. A partir de los resultados de salida, se pasa a su evaluación, mediante una comparación con la salida esperada. A partir de ésta, se pueden realizar dos actividades:

- La depuración (localización y corrección de defectos).
- El análisis de la estadística de errores.

El objetivo del proceso de pruebas no es, como pudiera parecer, demostrar que el software está libre de defectos, sino precisamente descubrir defectos. [DE ANTONIO 1999]

Por otra parte pueden existir productos con un número de defectos alto que sin embargo apenas fallen, y que serán percibidos por el usuario como de alta calidad, y en este caso se puede indicar que el proceso de pruebas ha sido poco exhaustivo y que la mayor parte de los defectos permanecen ocultos. Por lo que es muy importante seleccionar bien las pruebas que se van a realizar, teniendo en cuenta que sólo las pruebas que revelan defectos son las que realmente dan cumplimiento a su objetivo. [DE ANTONIO 1999]

### **1.4.1 Actividades del Proceso de Pruebas**

Para lograr conformar un proceso de pruebas hay que realizar un conjunto de actividades estándar, así como la producción de un conjunto de salidas estándar: [DE ANTONIO 1999]

Actividades Estándar de Prueba	Salidas Estándar de Prueba
Planificación de la prueba	Plan de prueba
Diseño de la prueba	Documento de diseño de la prueba
Determinación de los casos de prueba	Especificación de los casos de prueba
Planificación del procedimiento de prueba	Especificación del procedimiento de prueba
Ejecución de la prueba	Informe de los casos de prueba
Análisis y evaluación de la prueba	Informe de la prueba

**Figura 2: Actividades y salidas estándar del proceso de pruebas**

A continuación se dan a conocer cada una de estas actividades en detalle. [DE ANTONIO 1999]

1. Planificación de las pruebas: Esta actividad consiste en la creación de un plan de pruebas en el que se registra:

- El objetivo del proceso de pruebas.
- Los objetos que hay que probar.
- Las características que se van a probar y las que no.
- El método de prueba a utilizar.
- Los recursos que se van a emplear.
- El plan de tiempos.
- Los productos a generar durante las pruebas
- El reparto de las responsabilidades.

2. Diseño de las pruebas: Esta actividad consiste en dar instrucciones detalladas acerca de:

- ¿Cómo llevar a cabo la prueba para alcanzar los objetivos deseados?
- ¿De qué forma se van a utilizar los métodos de prueba?
- ¿Qué objetos se van a probar en cada una de las pruebas?

- ¿Qué criterios se van a utilizar para determinar si el objeto pasa o no pasa la prueba?

3. Determinación de los casos de prueba: Esta actividad consiste en especificar el conjunto de casos de prueba a utilizar en función del diseño realizado para la prueba.

Para cada caso de prueba habrá que especificar:

- ¿Qué objetos se van a probar?
- ¿Qué entradas se les van a dar?
- ¿Cuáles son las salidas esperadas?

4. Planificación del procedimiento de prueba: Esta actividad consiste en fijar un conjunto de pasos para la ejecución de la prueba. Se especifica detalladamente:

- La secuencia exacta de ejecución de los distintos casos de prueba.
- Los requisitos que hay que cumplir para la ejecución de cada caso de prueba.
- Las condiciones de terminación de cada uno de ellos.

5. Ejecución de las pruebas: Esta actividad consiste en ejecutar cada caso de prueba, según el procedimiento especificado en el paso anterior, y registrar los incidentes o problemas encontrados durante la misma.

6. Análisis y evaluación de las pruebas: Se examinan los resultados de la prueba y se decide si se han alcanzado los objetivos propuestos o se deben repetir las pruebas.

### **1.4.2 Ciclo del Proceso de Pruebas**

Para realizar el proceso de pruebas, se tiene en cuenta el siguiente esquema, en el cual se representa el ciclo de las actividades a realizar, con sus respectivas salidas asociadas.

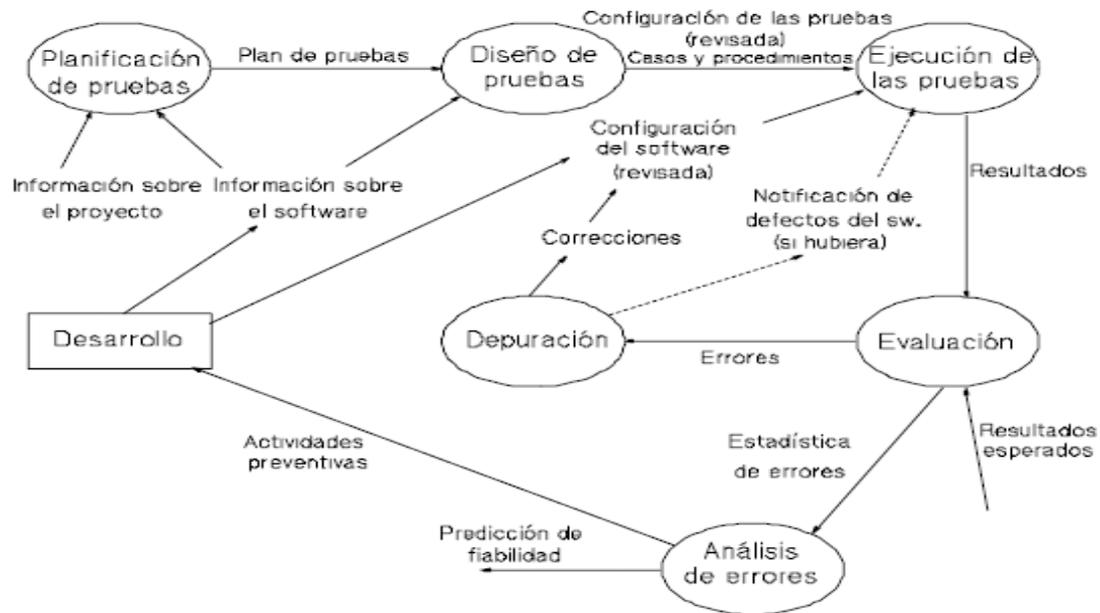


Figure 3: Ciclo del proceso de pruebas [LÓPEZ 2005]

## 1.5 Estrategia de Prueba

Es muy importante a la hora de crear el plan de pruebas, tener bien definida la estrategia de prueba que se va a seguir durante el proceso de pruebas. Son múltiples las estrategias de pruebas que se han propuesto, pero todas tienen las siguientes características: [PRESSMAN 2005]

- Las pruebas comienzan a nivel de módulo y trabajan hacia fuera, hacia la integración de todo el sistema.
- Según el momento, son apropiadas diferentes técnicas de pruebas.
- Las pruebas la lleva a cabo el responsable del desarrollo del software y en caso de que el proyecto sea grande un grupo independiente de pruebas.
- Las pruebas y la depuración son actividades diferentes, pero aún así la depuración se debe incluir en cualquier estrategia de prueba.

Una estrategia de prueba del software debe incluir 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 validen las principales funciones del sistema frente a los requisitos del cliente. Para la realización de estas pruebas existen dos métodos de prueba.

### **1.5.1 Métodos de Prueba**

Para la realización de las pruebas se utilizan métodos o procedimientos básicos que permiten diseñar los casos de prueba, según el tipo de prueba que se vaya a aplicar. Estos son:

- Métodos de Caja Negra
- Métodos de Caja Blanca

#### **Métodos de Caja Negra**

En este tipo de método, el objeto que se desea probar se ve como una caja negra. Esto quiere decir que la elección de los casos de prueba no se va a basar en el conocimiento que se tenga acerca de la estructura del objeto, sino en el conocimiento acerca de la funcionalidad deseada (descripción funcional). Por esto a la prueba de caja negra también se le llama prueba funcional o prueba orientada al diseño. [DE ANTONIO 1999]

La prueba de Caja Negra se centra principalmente en los requisitos funcionales del software. Estas pruebas permiten obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa. En ellas se ignora la estructura de control, concentrándose en los requisitos funcionales del sistema y ejercitándolos. [NAPAL 2003]

La prueba de Caja Negra no es una alternativa a las pruebas de Caja Blanca. Más bien se trata de un enfoque complementario, que intenta descubrir diferentes tipos de errores a los descubiertos mediante métodos de Caja Blanca. [PRESSMAN 2005]

La prueba de Caja Negra intenta encontrar errores de las diferentes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a las bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y terminación.

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

- *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.
- *Técnica del Análisis de Valores Límites:* Esta técnica prueba la habilidad del programa para manejar datos que se encuentran en los límites aceptables.
- *Técnica de Grafos de Causa-Efecto:* Es una técnica que permite al encargado de la prueba validar complejos conjuntos de acciones y condiciones.

### ***Técnica de la Partición Equivalente***

Una partición equivalente es una técnica de prueba de Caja Negra que divide el dominio de entrada de un programa en clases de datos de los que se pueden derivar casos de prueba.

El diseño de casos de prueba para la partición equivalente se basa en una evaluación de las clases de equivalencia para una condición de entrada. Una clase de equivalencia representa un conjunto de estados válidos o inválidos para condiciones de entrada. Regularmente, una condición de entrada es un valor numérico específico, un rango de valores, un conjunto de valores relacionados o una condición lógica. [NAPAL 2003]

Para aplicar esta técnica de prueba primeramente se deben identificar las clases de equivalencia. Una clase de equivalencia representa un conjunto de estados válidos o no válidos para condiciones de entrada. [PRESSMAN 2005]

Para definir las clases de equivalencia hace falta tener en cuenta un conjunto de reglas: [NAPAL 2003]

- Si una condición de entrada especifica un rango, entonces se confecciona una clase de equivalencia válida y 2 inválidas.
- Si una condición de entrada especifica la cantidad de valores, se debe identificar una clase de equivalencia válida y dos inválidas.
- Si una condición de entrada especifica un conjunto de valores de entrada y existen razones para creer que el programa trata en forma diferente a cada uno de ellos, se debe identificar una clase válida para cada uno de ellos y una clase inválida.
- Si una condición de entrada especifica una situación de tipo “debe ser”, se debe identificar una clase válida y una inválida.
- Si existe una razón para creer que el programa no trata de forma idéntica ciertos elementos pertenecientes a una clase, dividirla en clases de equivalencia menores.

### ***Técnica del Análisis de Valores Límites (AVL)***

El análisis de valores límites es una técnica de diseño de casos de prueba que complementa a la partición equivalente. En lugar de seleccionar cualquier elemento de una clase de equivalencia, el AVL lleva a la elección de los casos de prueba en los extremos de la clase. Además no se concentra solamente en las condiciones de entrada, AVL obtiene casos de prueba también para el campo de salida. [PRESSMAN 2005]

Las directrices AVL son similares en muchos aspectos a las que proporciona la partición equivalente:

- Si una condición de entrada especifica un rango delimitado por los valores a y b, se deben diseñar casos de prueba para los valores a y b, y para los valores justo por debajo y justo por encima de a y b, respectivamente.
- Si una condición de entrada especifica un número de valores, se deben desarrollar casos de prueba que ejerciten los valores máximo y mínimo. También se deben probar los valores justo por encima y justo por debajo del máximo y el mínimo.
- Aplicar las directrices 1 y 2 a las condiciones de salida.
- Si las estructuras de datos internas tienen límites preestablecidos, hay que asegurarse de diseñar un caso de prueba que ejercite la estructura de datos en sus límites.

### ***Técnica de Grafos de Causa-Efecto***

Consiste en crear un grafo causa-efecto a partir de las especificaciones, y seleccionar suficientes casos de prueba como para asegurar la cobertura del grafo. Se llama “causas” a las características de los datos de entrada y “efectos” a las clases de salida que puede proporcionar el programa. A partir del grafo causa-efecto se construye una tabla de decisión que refleje las dependencias entre causas y efectos. Lo que se hace entonces es reducir la tabla de decisión y seleccionar sólo un caso de prueba para todas las causas que producen el mismo efecto, o para cada columna de la tabla de decisión. [DE ANTONIO 1999]

### **Métodos de Caja Blanca**

En este tipo de métodos, el objeto que se desea probar se ve como una Caja Blanca. Esto quiere decir, que la elección de los casos de prueba se va a basar en el conocimiento que se tenga acerca de la estructura del objeto (diseño detallado, diagramas de flujo de datos y de control, código fuente). A la prueba de Caja Blanca también se le llama Prueba Estructural. [DE ANTONIO 1999]

Mediante el método de Caja Blanca el ingeniero del *software* puede obtener casos de prueba que: [PRESSMAN 2005]

- Garanticen que se ejerciten por lo menos una vez todos los caminos independientes de cada módulo, programa o método.
- Ejerciten todas las decisiones lógicas en las vertientes: verdadera y falsa.
- Ejecuten todos los bucles en sus límites operacionales.
- Ejerciten las estructuras internas de datos para asegurar su validez.

Valdría la pena preguntarnos: ¿Por qué emplear tiempo y energía probando minuciosidades lógicas, cuando se podía emplear mejor el esfuerzo, asegurando que se hayan alcanzado los requisitos del programa? La respuesta se encuentra en la naturaleza misma de los defectos del software: [PRESSMAN 2005]

- Los errores lógicos y las suposiciones incorrectas son inversamente proporcionales a la probabilidad de que se ejecute un camino del programa. Los errores tienden a introducirse en nuestro trabajo cuando se diseñan e implementan funciones, condiciones o controles que se encuentran fuera de lo normal. El procedimiento habitual tiende a hacerse más comprensible, mientras que el procesamiento de casos especiales tiende a caer en el caos.
- A menudo se cree que un camino lógico tiene pocas posibilidades de ejecutarse, cuando de hecho, se puede ejecutar de forma normal. El flujo lógico de un programa a veces no es nada intuitivo, lo que significa que nuestras suposiciones intuitivas sobre el flujo de control y los datos pueden llevar a tener errores de diseño que sólo se descubren cuando comienza la prueba del camino.
- Los errores tipográficos son aleatorios. Cuando se traduce un programa a código fuente en un lenguaje de programación, es muy probable que se den algunos errores de escritura. Muchos serán descubiertos por los mecanismos de comprobación de sintaxis, pero otros permanecerán sin detectar hasta que comiencen las pruebas. Es igual de probable que haya un error tipográfico en un oscuro camino lógico que en un camino principal.

Las Técnicas de Caja Blanca se pueden clasificar en dos grupos: [DE ANTONIO 1999]

- Basadas en Métricas de Cobertura
- Basadas en Métricas de Complejidad

### ***Técnicas Basadas en Métricas de Cobertura***

Todo programa se puede representar mediante un grafo de flujo de control, donde cada nodo es una sentencia o una secuencia de sentencias. Los arcos dirigidos en el grafo representan el flujo de control.

Para cada conjunto de datos de entrada el programa se ejecutará a través de un camino concreto dentro de este grafo. Cuando el programa incluye estructuras iterativas, el número de posibles caminos en el grafo puede ser infinito.

Una prueba de Caja Blanca exhaustiva requeriría la generación de un caso de prueba por cada posible camino. Como esto no es posible, por lo general, se utilizan métricas que dan una indicación de la calidad de un determinado conjunto de casos de prueba en función del grado de cobertura.

Las métricas más utilizadas son: [FERNÁNDEZ-MEDINA 2006]

- *Cobertura de sentencias*: Se trata de generar los casos de prueba necesarios para que cada sentencia o instrucción del programa se ejecute al menos una vez.
- *Cobertura de decisiones*: Consiste en escribir casos de prueba suficientes para que cada decisión tenga, por lo menos una vez, un resultado verdadero y al menos una vez, uno falso.
- *Cobertura de condiciones*: Se trata de diseñar tantos casos de prueba como sea necesario para que cada condición, de cada decisión, adopte el valor verdadero al menos una vez y el falso al menos una vez.

- *Criterio de decisión-condición:* Consiste en exigir el criterio de cobertura de condiciones obligando a que se cumpla también el criterio de decisiones.
- *Criterio de condición múltiple:* En el caso de que se considere que la evaluación de las condiciones de cada decisión no se realiza de forma simultánea, se puede considerar que cada decisión multicondicional se descompone en varias condiciones unicondicionales.
- *Criterio de cobertura de caminos:* Se recorren todos los caminos.

### ***Técnicas Basadas en Métricas de Complejidad***

Las métricas de complejidad más utilizadas en la generación de casos de prueba son las de MacCabe. El criterio de prueba de McCabe es: Elegir tantos casos de prueba como caminos independientes. [FERNÁNDEZ-MEDINA 2006]

Métricas de McCabe: [DE ANTONIO 1999]

- Complejidad ciclomática (arcos - nodos + 2 \* número de componentes conexos).
- Complejidad esencial (complejidad ciclomática - número de subgrafos propios de entrada y salida única).
- Complejidad real (número de caminos ejecutados).

### **1.5.2 Niveles de Prueba**

De acuerdo con el estándar IEEE 1012-1986 el conjunto mínimo de pruebas o niveles de prueba como también se le suele llamar, que se deben realizar son: [DE ANTONIO 1999]

- Nivel Modular, Nivel de Unidad o Nivel de Componentes
- Nivel de Integración

- Nivel de Sistema
- Nivel de Aceptación

Aunque contempla la posible realización de otro tipo de nivel llamado nivel de regresión.

### **Nivel de Unidad**

Consiste en la prueba de cada módulo aislado del resto del sistema. Las pruebas que se dan como parte de las pruebas de unidad son las siguientes: [PRESSMAN 2005]

- *Prueba de Interfaz:* Se prueba la interfaz del módulo para asegurar que la información fluye de forma adecuada hacia y desde la unidad del programa que está siendo probada.
- *Prueba de Estructura de Datos Locales:* Se examinan las estructuras de datos locales para asegurar que los datos que se mantienen temporalmente conservan su integridad durante todos los pasos de ejecución del algoritmo.
- *Prueba de Condiciones Límite:* Se prueban las condiciones límite para asegurar que el módulo funciona correctamente en los límites establecidos como restricciones de procesamiento.
- *Prueba de Caminos Independientes y de Manejo de Errores:* Se ejercitan todos los caminos independientes (caminos básicos) de la estructura de control con el fin de asegurar que todas las sentencias del módulo se ejecutan por lo menos una vez. Finalmente se prueban los caminos de manejo de errores.

### **Nivel de Integración**

Se realiza a medida que los diferentes módulos del sistema se integran en el mismo. Esto sucede una vez que se ha realizado la prueba modular, por lo que se parte de que todos los módulos son correctos. El objetivo fundamental de esta prueba es comprobar que las interfaces entre los distintos módulos son correctas. [DE ANTONIO 1999]

Algunas de las comprobaciones que es necesario realizar son:

- Corrección en la sintaxis, en la invocación de procedimientos y funciones.
- Compatibilidad de tipos entre los argumentos del procedimiento o función y los parámetros de llamada.
- Corrección y completitud de las especificaciones de los módulos.

Para este tipo de prueba se pueden utilizar cuatro posibles estrategias de integración: [DE ANTONIO 1999]

- *De arriba a abajo (top-down)*: Consiste en empezar la integración y las pruebas por los módulos que están en los niveles superiores de abstracción, e integrar incrementalmente los niveles inferiores.
- *De abajo a arriba (bottom-up)*: Consiste en empezar la integración y las pruebas por los módulos que están en los niveles inferiores de abstracción, e integrar incrementalmente los niveles superiores.
- *De big-bang*: Consiste en integrar y probar todo al mismo tiempo.
- *De regresión*: Consiste en volver a ejecutar un subconjunto de pruebas que se hayan llevado a cabo anteriormente, para asegurarse de que los cambios no han propagado efectos colaterales no deseados. [PRESSMAN 2005]

También existe un método de prueba de integración llamado *Prueba de Humo*, la cual es comúnmente utilizada cuando se ha desarrollado un producto software empaquetado. Es diseñado como un mecanismo para proyectos críticos por tiempo, permitiendo que el equipo de software valore su proyecto sobre una base sólida. En esencia, comprende las siguientes actividades: [PRESSMAN 2005]

- Los componentes software que han sido traducidos a código se integran en una construcción. Una construcción incluye ficheros de datos, librerías, módulos reutilizables y componentes de ingeniería que se requieren para implementar una o más funciones del producto.
- Se diseña una serie de pruebas para descubrir errores que impiden a la construcción realizar su función adecuadamente. El objetivo será descubrir errores bloqueantes que tengan la mayor probabilidad de impedir al proyecto de software el cumplimiento de su planificación.
- Es habitual que la construcción se integre con otras construcciones y que se aplica una prueba de humo al producto completo (en su forma actual). La integración puede hacerse bien de forma descendente (top-down) o ascendente (bottom-up).

### **Nivel de Sistema**

Se realiza cuando se han integrado todos los módulos, y su objetivo es comprobar que el sistema satisface los requisitos del usuario, tanto los funcionales como los no funcionales. [DE ANTONIO 1999]

La prueba del sistema realmente está constituida por una serie de pruebas diferentes cuyo propósito primordial es ejercitar profundamente el sistema basado en computadora. Aunque cada prueba tiene un propósito diferente, todas trabajan para verificar que se hayan integrado adecuadamente todos los elementos del sistema y que realizan las funciones apropiadas. [PRESSMAN 2005]

Los tipos de prueba de sistema son los siguientes:

- *Prueba de Recuperación:* Es una prueba del sistema que fuerza al fallo del software de muchas formas y verifica que la recuperación se lleva a cabo apropiadamente. Si la recuperación es automática (llevada a cabo por el propio sistema) hay que evaluar la corrección de la inicialización, de los mecanismos de recuperación del estado del sistema, de la recuperación de datos y del proceso de re arranque. Si la recuperación requiere la intervención humana, hay que evaluar los tiempos medios de reparación para determinar si están dentro de unos límites aceptables.

- *Prueba de Seguridad:* Intenta verificar que los mecanismos de protección incorporados en el sistema lo protegerán de accesos impropios. Para citar a Beizer [Beizer, 1984]: “Por supuesto, la seguridad del sistema debe ser probada en su invulnerabilidad frente a un ataque frontal, pero también debe probarse en su invulnerabilidad a ataques por los flancos o por la retaguardia.” El responsable de la prueba desempeña el papel de un individuo que desea entrar en el sistema. Debe intentar conseguir las claves de acceso por cualquier medio, puede atacar al sistema con algún software, diseñado para romper cualquier defensa que se haya construido, debe bloquear el sistema, negando así el servicio a otras personas, debe producir a propósito errores del sistema, intentando acceder durante la recuperación o debe curiosear en los datos sin protección, intentando encontrar la clave de acceso al sistema.
- *Prueba de Resistencia (Stress):* Están diseñadas para enfrentar a los programas con situaciones anormales. En esencia el sujeto que realiza la prueba de resistencia se pregunta: ¿A qué potencia se puede poner a funcionar antes de que falle? La prueba de resistencia ejecuta un sistema de forma que demande recursos en cantidad, frecuencia o volúmenes anormales. El objetivo del responsable de la prueba es intentar romper el programa de alguna manera. Una variante de la prueba de resistencia es la *prueba de sensibilidad*, la cual intenta descubrir combinaciones dentro de una clase de entrada válida que pueda producir inestabilidad o un proceso incorrecto.
- *Prueba de Rendimiento:* Está diseñada para probar el rendimiento del software en tiempo de ejecución, dentro del contexto de un sistema integrado. Las pruebas de rendimiento a menudo van emparejadas con las pruebas de resistencia y frecuentemente requieren instrumentación tanto de software como de hardware.

### **Nivel de Aceptación**

Se realiza una vez que el sistema se ha implantado en su entorno real de funcionamiento, y su objetivo es demostrar al usuario que el sistema satisface sus necesidades. [DE ANTONIO 1999]

Estas pruebas las realiza el usuario final en lugar del responsable del desarrollo del sistema, una prueba de aceptación puede ir desde un informal paso de prueba, hasta la ejecución sistemática de una serie de

pruebas bien planificadas. Si el software se desarrolla como un producto que va a ser usado por muchos clientes, no es práctico realizar pruebas de aceptación formales para cada uno de ellos. La mayoría de los desarrolladores de productos de software llevan a cabo un proceso denominado prueba alfa y beta para descubrir errores que parezca que solo el usuario final puede descubrir. [PRESSMAN 2005]

- *Prueba Alfa:* Se lleva a cabo por un cliente en el lugar de desarrollo. Se usa el software de forma natural con el desarrollador como observador del usuario y registrando los errores y los problemas de uso. Las pruebas alfa se llevan a cabo en un entorno controlado.
- *Prueba Beta:* Se lleva a cabo por los usuarios finales del software en los lugares de trabajo de los clientes. A diferencia de la prueba alfa, el desarrollador no está presente normalmente. Así, la prueba beta es una aplicación en vivo del software en un entorno que no puede ser controlado por el desarrollador. El cliente registra todos los problemas (reales o imaginarios) que encuentra durante la prueba beta e informa a intervalos regulares al desarrollador. Como resultado de los problemas informados durante la prueba beta, el desarrollador del software lleva a cabo modificaciones y así prepara una versión del producto software para toda clase de clientes.

### **Nivel de Regresión**

Tiene como objetivo comprobar que toda nueva versión de un producto software es de no menos calidad que la versión anterior, es decir, que al introducir cambios no se ha reducido la valoración de ninguna de las características de calidad que tenía el producto. [DE ANTONIO 1999]

El conjunto de pruebas de regresión contiene tres clases diferentes de casos de prueba: [PRESSMAN 2005]

- Una muestra representativa de pruebas que ejercite todas las funciones del software.
- Pruebas adicionales que se centran en las funciones del software que se van a ver probablemente afectadas por el cambio.
- Pruebas que se centran en los componentes del software que se han cambiado.

El conjunto de pruebas de regresión debería diseñarse para incluir solo aquellas pruebas que traten una o más clases de errores en cada una de las funciones principales del programa. No es práctico ni eficiente volver a ejecutar cada prueba de cada función del programa después de un cambio.

## 1.6 Antecedentes de Pruebas a Multimedia

Para realizar un buen análisis de los antecedentes de pruebas a multimedia, se van a tener en cuenta los antecedentes de las mismas a nivel internacional, nacional y específicamente en la Universidad de las Ciencias Informáticas (UCI).

### A Nivel Internacional

ParqueSoft (Parque Tecnológico de Software), se inició en 1999, en Colombia y su misión es servir como espacio de innovación y respaldo para emprendedores en el área de desarrollo de software.

En ParqueSoft se realizan dos programas para realizar pruebas a multimedia: [GÓMEZ 2006]

- *Programa Pruebas de Usabilidad Multimedia:* Consiste en conformar un equipo multidisciplinario para definir herramientas de trabajo, entre ellas se destaca el estándar de usabilidad por tipo de multimedia (educativas, informativas) que posteriormente servirá para hacer las pruebas manuales y semiautomáticas. En el equipo participan: ingenieros de sistema que trabajan concretamente en un generador estándar de framework para multimedia, diseñadores gráficos que son responsables del look and feel e ingenieros de pruebas, que a partir del trabajo realizado en el laboratorio tiene la responsabilidad de abstraer una metodología para las pruebas de usabilidad.
- *Programa Pruebas de Usabilidad Multimedia para niños:* Consiste en evaluar mediante la observación en uso, para lo cual se cuenta con un laboratorio y un grupo de niños entre los 6 y 14 años, que utilizan las multimedia como usuarios finales, lo cual permite que el ingeniero de pruebas líder evalué un conjunto de criterios previamente definidos, logrando encontrar aspectos a mejorar en el software asociados a la calidad en uso de los productos.

## **A Nivel Nacional**

La Empresa de Tecnologías de la Información y Servicios Telemáticos Avanzados, más conocida como CITMATEL, cuenta con un grupo de calidad, el cual realiza pruebas de funcionalidad una vez culminadas las multimedia sin aplicar ningún método de prueba, pues se basan para realizar esta actividad en el conocimiento y experiencia del personal que integra este equipo, el cual se está capacitando continuamente. Para la elaboración de las multimedia utilizan una metodología basada en RUP, la cual fue creada en la propia empresa. [GUERRERO 2007]

El Centro de Información y Gestión Tecnológica de Camagüey (CIGET) no realizan un trabajo sistémico a la hora de realizarle pruebas a las multimedia que ellos producen, pues muchas veces carecen de tiempo hasta para documentar. Trabajan guiados por la experiencia muchas veces, aunque se basan en una metodología para el desarrollo de multimedia llamada Multimed. A las multimedia le realizan pruebas funcionales a las cuales ellos mismos le ponen nombres, como es el caso de: Prueba de Navegación (el objetivo de esta prueba es comprobar que la navegación sea factible y didáctica), Interactividad (el objetivo de esta prueba es comprobar el grados de interactividad en todas las pantallas), Medios (el objetivo de esta prueba es comprobar que se cargan correctamente los videos, imágenes, sonidos y animaciones) y Resolución de Pantallas (el objetivo de esta prueba es comprobar que la multimedia funcione correctamente en distintas resoluciones de pantalla), entre otras que van a estar en dependencia de las características de la multimedia y sus requerimientos. [ESTRADA 2007]

## **En la Universidad de las Ciencias Informáticas**

En la Universidad de las Ciencias Informáticas, en la facultad 8, se realizan productos multimedia a los cuales se le realizan pruebas funcionales y por aspectos (a través de un guión de revisión), para las cuales no utilizan ningún método de prueba y se hacen una vez que se ha completado la elaboración de la misma. Luego de aplicadas estas pruebas, se emite el resultado en un documento de no conformidades que se le entrega al líder del proyecto y luego de corregidas el equipo de calidad repite las mismas pruebas. Si finalmente la multimedia está libre de errores el mismo equipo de calidad hace entrega de un certificado de calidad del producto. [MEDINA 2007]

## 1.7 Conclusiones

A lo largo de este capítulo se ha podido observar, que a pesar de las diferentes definiciones obtenidas de *Prueba* para evaluar la calidad de un software, se considera más prudente sustituir el término de prueba, por uno más complejo y abarcador que es *Proceso de Pruebas*.

Se puede decir que la estrategia de prueba es muy importante a la hora de realizar un proceso de pruebas que garantice la calidad de un software, ya que en ella se van a plantear los métodos de prueba que se van a tener en cuenta, los niveles y dentro de estos los tipos de prueba.

Es necesario destacar que las prácticas de prueba a multimedia que desarrollan diferentes entidades, tanto nacionales como internacionales, no responden al proceso que se ha planteado y se realizan muchas veces a última hora sin haberse planificado con tiempo, y en la mayoría de los casos se basan en la práctica obtenida en cada una de estas entidades, lo cual está mal hecho y se debe de trabajar en erradicar estos problemas.

## Descripción del Proceso de Pruebas

### Introducción

En este capítulo se van a realizar las principales actividades que se tienen en cuenta para la aplicación del proceso de pruebas a la multimedia “Organopónico”. Antes de comenzar la aplicación del proceso de pruebas, se va a adecuar el proceso de pruebas estudiado en el capítulo anterior a las características de la multimedia “Organopónico”. Una vez culminada esta actividad se va a describir el plan de pruebas y se van a diseñar las pruebas de la multimedia, a partir de los datos que se generen de estas dos actividades, se van a realizar tantos casos de prueba como se necesiten.

También se va a realizar una breve descripción de la multimedia “Organopónico” para conocer su estructura e importancia.

### 2.1 Descripción de la Multimedia

La multimedia “Organopónico” fue creada en coordinación con el Núcleo de Desarrollo Endógeno Fabricio Ojeda en apoyo a la iniciativa del gobierno de incentivar el desarrollo endógeno y la economía social, sustentado en la participación de las comunidades rurales y urbanas, organizadas como entes productivos; capaces de generar sus propios alimentos a través del desarrollo de cultivos organopónicos y huertos familiares, para autoabastecerse y colocar productos para la venta en el mercado nacional a precios solidarios. Partiendo de esta premisa el Núcleo Fabricio Ojeda se ha propuesto elaborar un software informativo de cultivos organopónicos como parte del programa agrícola, a fin de documentar y transmitir a otras comunidades estos conocimientos y experiencias, a través de las nuevas tecnologías de la información y la comunicación.

### **Ventajas que aporta el software**

- Permite socializar los conocimientos, con este software se puede llevar la información a todas las personas que quieran aprender cómo desarrollar sus propios cultivos organopónicos.
- Incentiva a las comunidades a participar y ser protagonistas de su propio desarrollo, a través de su participación activa en la economía del país.
- Promueve la importancia de la participación de la población y en especial a las personas de la tercera edad, en actividades agrícolas que beneficien la economía social del país.

Este software tiene como propósito dar a conocer a la población adulta venezolana qué son cultivos organopónicos y cómo se producen y comercializan. Se caracteriza por presentar la información en forma detallada y completa, pero al mismo tiempo sencillo e ilustrativo, con imágenes y animaciones que facilitan la comprensión del tema. Este software informativo está estructurado en cuatro módulos de contenido (ver en **Anexo 1** el diagrama de navegación):

- Módulo 1: Elementos Básicos de los Cultivos Organopónicos.
- Módulo 2: Estructuras utilizadas en los Cultivos Organopónicos.
- Módulo 3: Proceso de Cultivo.
- Módulo 4: Recolección, Clasificación y Comercialización de la Cosecha.

A continuación se explica la información con que cuenta cada uno de los módulos de la multimedia:

#### **Módulo 1: Elementos Básicos de los Cultivos Organopónicos**

Este módulo es de vital importancia, pues explica lo que se entiende por *cultivos organopónicos*, la importancia que tiene su producción para la población y en general la economía del país. También se muestran los *materiales* y *herramientas* que se necesitan y cómo debe estar preparado el terreno para poder iniciar el cultivo.

### **Módulo 2: Estructuras utilizadas en los Cultivos Organopónicos**

Este módulo presenta información acerca del significado de *compostero*, su importancia, los materiales orgánicos que se colocan en el mismo, el proceso que se debe llevar a cabo para su elaboración en el suelo y el procedimiento para la remoción de la mezcla del material orgánico en el mismo. También presenta información acerca del significado de los *canteros*, su importancia y el proceso que se debe seguir para su construcción. Por último explica lo que se entiende por *semillero*, la importancia del mismo y el proceso para la elaboración de un semillero temporal.

### **Módulo 3: Proceso de Cultivo**

Este módulo explica cómo es el proceso de transplante de las plántulas del semillero al cantero donde van a crecer, los tipos de sistemas de riego de plantas que existen y lo que se entiende por *fertilizantes* y *control de plagas*. También brinda información acerca de los tipos de plagas que afectan los cultivos organopónicos y los productos naturales más comunes que se utilizan para el control de plagas y enfermedades en los cultivos sin dañar el medio ambiente.

### **Módulo 4: Recolección, Clasificación y Comercialización de la Cosecha**

Este módulo brinda información acerca de cómo llevar a cabo la recolección de la cosecha, el proceso de clasificación de la misma y su embalaje y comercialización.

## **2.2 Adaptación del Proceso de Pruebas a la multimedia “Organopónico”**

Para lograr establecer un proceso de pruebas al producto multimedia “Organopónico”, primeramente se tuvo que adecuar el proceso de pruebas estudiando en el Capítulo 1 de este trabajo, a las características particulares de la multimedia “Organopónico”, el cual quedó estructurado de esta manera:

1. *Planificación de las pruebas*: Consiste en la creación de un plan de pruebas en el que se registra:

- Objetivo del proceso de pruebas.
- Plan de tiempos de las actividades.
- Reparto de responsabilidades X rol.
- Recursos.
- Estrategia de prueba.
- Entregables.
- Requerimientos a probar.

2. *Diseño de las pruebas*: En el diseño de prueba se muestran instrucciones detalladas acerca de la forma en que se van a realizar cada una de las pruebas especificada en la Estrategia de Prueba, teniendo en cuenta los siguientes aspectos:

- **Objetivos**: Los objetivos que se desean alcanzar con la realización de la prueba.
- **Método**: El método que se va a utilizar para realizar la prueba y la forma en que se va a utilizar.
- **Requisitos**: Los requisitos que van a ser probados al aplicar la prueba.
- **Criterio**: El criterio que se va a tener en cuenta para determinar si la prueba se pasa o no.

3. *Determinación de los Casos de Prueba*: Para la realización de los casos de prueba se crearon dos plantillas, que van a ser utilizadas en dependencia del método especificado para cada prueba en el diseño de las pruebas. En cada plantilla de casos de prueba está contenida la Planificación del Procedimiento de Prueba, es decir, el conjunto de pasos para la ejecución de la prueba, por lo cual esta actividad no se realiza de forma independiente.

**Tabla 1:** Plantilla de Casos de Prueba mediante el Método de Caja Negra.

<b>Requisito</b>	<i>&lt;Los requisitos que se desean probar al realizar el caso de prueba&gt;</i>		
<b>Objetivos</b>	<i>&lt;Se describen los objetivos que se persiguen con la realización del caso de prueba&gt;</i>		
<b>Pasos</b>	<i>&lt;Los pasos que se deben dar, para llegar a el lugar exacto de la multimedia donde se va a aplicar el caso de prueba&gt;</i>		
<b>Condiciones de Ejecución</b>	<i>&lt;Las condiciones que se deben cumplir para poder realizar el caso de prueba&gt;</i>		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<i>&lt;Es una condición lógica de entrada a la aplicación, es decir, una acción que realiza el usuario esperando que la aplicación le de una respuesta, en el caso de una multimedia habría una clase válida cada vez que interactué con algún vínculo o algún botón&gt;</i>	<i>&lt;Es la condición lógica de entrada va a ser booleana, es decir, puede que funcione correctamente el vínculo o el botón, pero también puede que no, por cada clase válida va a existir una clase inválida&gt;</i>	<i>&lt;Las funcionalidades que se esperan que realice la aplicación en caso de que se cumpla la clase válida&gt;</i>	<i>&lt;El resultado de la prueba va a ser Satisfactorio, en caso de que se realice correctamente la funcionalidad o la acción que se está probando, e Inválida en caso contrario&gt;</i>

**Tabla 2:** Plantilla de Casos de Prueba que no utilizan el Método de Caja Negra ni el de Caja Blanca.

<b>Objetivos</b>	<Se describen los objetivos que se persiguen con la realización del caso de prueba>		
<b>Pasos</b>	<Los pasos que se deben dar, para llegar a el lugar exacto de la multimedia donde se va a aplicar el caso de prueba>		
<b>Condiciones de Ejecución</b>	<Las condiciones que se deben cumplir para poder realizar el caso de prueba>		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
	<b>Requisito</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
	<Los requisitos que se desean probar en el caso de prueba>	<Las forma en que se espera que responda la aplicación, al comprobar que se le de cumplimiento el requisito correspondiente>	<El resultado de la prueba va a ser Satisfactorio, en caso de que se realice correctamente el resultado esperado, e Insatisfactorio en caso contrario>

4. *Ejecución de las pruebas:* Consiste en ejecutar cada caso de prueba, según el procedimiento especificado en cada uno de ellos y registrar los defectos encontrados, mediante la siguiente tabla:

**Tabla 3:** Plantilla de Registro de Defectos.

No.	No Conformidad	Lugar	CP	Importancia
	<i>&lt;Los defectos que se han encontrado al ejecutar las pruebas&gt;</i>	<i>&lt;Lugar de la multimedia en el cual se localiza el defecto encontrado&gt;</i>	<i>&lt;Caso de Prueba en el cual se detectó el defecto correspondiente&gt;</i>	<i>&lt;Alta o Baja en dependencia del cumplimiento de los objetivos con que se creó la multimedia&gt;</i>

5. *Análisis de los resultados:* Se examinan los resultados de la ejecución de las pruebas y se decide si se ha realizado eficientemente el proceso de pruebas y si la multimedia posee la calidad que requiere para su exportación.

### 2.3 Aplicación del Proceso de Pruebas

A continuación se van a aplicar cada una de las actividades antes establecidas a la multimedia “Organopónico”.

#### 2.3.1 Plan de Prueba

Para la planificación de las pruebas se van a tener en cuenta un conjunto de aspectos que ayudan a realizar esta actividad, a continuación se describen cada uno de ellos.

### 2.3.1.1 Objetivo del proceso de pruebas

El principal objetivo que se persigue con la aplicación del proceso de pruebas a la multimedia “Organopónico” es: detectar el mayor número posible de defectos que existan en este producto y luego de un análisis de los mismos, decidir si la multimedia tiene la calidad requerida para su exportación o no.

### 2.3.1.2 Plan de tiempos de las actividades

La planificación del tiempo destinado para cada actividad, se realizó en el Microsoft Office Project. En la tabla se encuentran las tareas enumeradas en el orden en que se van a realizar, la duración en días, la fecha de comienzo y de fin (no se tuvo en cuenta los sábados ni los domingos) y por último la dependencia (consiste en definir las tareas que dependen del cumplimiento de otras, en la tabla se muestra este aspecto a través de los números de predecessors).

		Task Name	Duration	Start	Finish	Predecessors
1		Planificar las pruebas	15 days	Mon 1/15/07	Fri 2/2/07	
2		Diseñar las pruebas	15 days	Mon 2/5/07	Fri 2/23/07	1
3		Realizar los casos de prueba	30 days	Mon 2/26/07	Fri 4/6/07	2
4		Ejecutar las pruebas	5 days	Mon 4/9/07	Fri 4/13/07	3
5		Registrar las no conformidades	5 days	Mon 4/16/07	Fri 4/20/07	4
6		Analizar los resultados	7 days	Mon 4/23/07	Tue 5/1/07	5

**Figure 4: Plan de tiempo de las actividades del proceso de pruebas**

### 2.3.1.3 Reparto de responsabilidades X rol

Para la realización del proceso de pruebas a la multimedia se recomienda contar con los roles que se muestran a continuación:

- Diseñador de prueba: Es el encargado de realizar el plan de prueba, el diseño de prueba y los casos de prueba. También realiza el análisis de los resultados.

- Probador: Es el encargado de ejecutar las pruebas, basándose en el plan de prueba, el diseño y los casos de prueba que realizó el diseñador de prueba. También registra los resultados de las pruebas.

En el caso específico de este trabajo, se cuenta con una sola persona para realizar el proceso de pruebas, por tanto la misma desempeña los dos roles.

#### **2.3.1.4 Recursos**

Para la realización del proceso de pruebas a la multimedia “Organopónico” es necesario contar como mínimo con una PC con los siguientes requerimientos:

Requerimientos de software:

- Flash MX 2004 instalado.
- Sistema Operativo Windows 98 o superior, Mac o Linux.

Requerimientos de hardware:

- Procesador PENTIUM a 200 MHz de velocidad de procesamiento.
- 32 MB de RAM.
- Tarjeta de Video.
- Tarjeta de Sonido.
- 700 Mb al menos de espacio libre en el disco duro.
- Lector de CD.
- Mouse.

### 2.3.1.5 Estrategia de prueba

Al producto multimedia “Organopónico” se le aplicarán pruebas en los *Niveles de Unidad y Sistema*. En el nivel de unidad se le harán pruebas a cada uno de los cuatro módulos de la multimedia, separándolos del resto de la aplicación. En el nivel de sistema se tendrá en cuenta la integración de los cuatro módulos y se le harán pruebas a toda la aplicación para comprobar que funciona como un sistema.

No se realizarán pruebas en el nivel de integración, porque no tiene lógica ir integrando poco a poco cada uno de los módulos y probándolos, cuando ya la multimedia está terminada, es más factible hacer estas pruebas en el sistema. Tampoco se harán pruebas en el nivel de aceptación, porque los usuarios de la multimedia se encuentran en Venezuela y se dificulta el contanto con ellos, ni de regresión, porque el objetivo del trabajo es solo detectar y reportar los defectos que se encuentren.

De los dos métodos existentes para la realización de pruebas, solo se utilizará el de *Caja Negra*, porque no se cuenta con el código de la multimedia, por tanto no se puede utilizar el método de *Caja Blanca*.

Para la realización de la *Caja Negra* se va a utilizar de las tres técnicas existentes, la *Técnica de Partición de Equivalencia* porque es una de las más efectivas, ya que permite examinar los valores válidos e inválidos de las entradas existentes en el software y descubre de forma inmediata las clases de errores que de otro modo, requerirían la ejecución de muchos casos de prueba antes de detectar el error genérico.

Se le realizarán además las siguientes pruebas según los niveles y métodos acordados:

- *Prueba de Funcionalidad*: Se aplicará en los niveles de unidad y sistema. Se utilizará el método de *Caja Negra*
- *Prueba de Portabilidad*: Se aplicará solamente en el nivel de sistema. No se utilizará ninguno de los dos métodos.
- *Prueba de Diseño*: Se aplicará solamente en el nivel de sistema. No se utilizará ninguno de los dos métodos.

No se va a realizar ninguna de las pruebas que se definieron en el capítulo anterior para cada uno de los niveles, porque no se creyó conveniente, debido a las características propias que presenta la multimedia “Organopónico”.

### **2.3.1.6 Entregables**

Durante el proceso de pruebas que se está planificando hacer al producto multimedia “Organopónico”, se van a generar los siguientes entregables:

- *Plan de Pruebas:* Contiene la estrategia a seguir para aplicar el proceso de pruebas, la planificación de cada una de las actividades que se van a realizar, los requisitos que serán probados y los recursos y roles que se necesitan para cumplir exitosamente con el objetivo planteado.
- *Diseño de Prueba:* Contiene la explicación de las pruebas que se tuvieron en cuenta en la estrategia y la forma en que se van a aplicar.
- *Casos de Prueba:* Se van a generar tantos casos de prueba como se requieran para poder ejecutar todas las pruebas que fueron primeramente planificadas y luego diseñadas. Cada caso de prueba va a contener la planificación del procedimiento de prueba.
- *Registro de No Conformidades:* Contiene los defectos que se detectaron luego de ejecutarse todos los casos de prueba.

### **2.3.1.7 Requerimientos a probar**

Los requerimientos a probar se van a dividir teniendo en cuenta cada uno de los niveles que se definieron en la estrategia de prueba.

### **Requerimientos del Nivel de Unidad:**

Se van a probar los cuatro módulos de la multimedia “Organopónico”, las funcionalidades implicadas en este nivel se han distribuido por cada uno de ellos como se muestra a continuación.

#### Módulo 1: Elementos Básicos de los Cultivos Organopónicos

**RF1:** Mostrar el contenido que se aborda en: Definición e Importancia.

**RF2:** Mostrar el contenido que se aborda en: Materiales y Herramientas.

**RF3:** Mostrar el contenido que se aborda en: Preparación del Terreno.

**RF4:** Permitir realizar la actividad: Completa la definición.

**RF5:** Permitir realizar la actividad: Pareo.

#### Módulo 2: Estructuras utilizadas en los Cultivos Organopónicos

**RF6:** Mostrar el contenido que se aborda en: Compostero (Definición e importancia).

**RF7:** Mostrar el contenido que se aborda en: Compostero (Proceso de elaboración).

**RF8:** Mostrar el contenido que se aborda en: Compostero (Remoción de la mezcla del material orgánico).

**RF9:** Mostrar el contenido que se aborda en: Canteros (Definición e importancia).

**RF10:** Mostrar el contenido que se aborda en: Canteros (Proceso de elaboración).

**RF11:** Mostrar el contenido que se aborda en: Semillero (Definición e importancia).

**RF12:** Mostrar el contenido que se aborda en: Semilleros (Proceso de elaboración).

**RF13:** Permitir realizar la actividad: Elabora virtualmente un compostero.

**RF14:** Permitir realizar la actividad: Sopa de letras.

#### Módulo 3: Proceso de Cultivo

**RF15:** Mostrar el contenido que se aborda en: Transplante del semillero al cantero.

**RF16:** Mostrar el contenido que se aborda en: Sistemas de riego.

**RF17:** Mostrar el contenido que se aborda en: Fertilizantes.

**RF18:** Mostrar el contenido que se aborda en: Tipos de plagas.

**RF19:** Mostrar el contenido que se aborda en: Control de plagas.

**RF20:** Permitir realizar la actividad: Auto evaluación.

Módulo 4: Recolección, Clasificación y Comercialización de la Cosecha

**RF21:** Mostrar el contenido que se aborda en: Recolección y Clasificación de la cosecha.

**RF22:** Mostrar el contenido que se aborda en: Embalaje y Clasificación.

**Requerimientos del Nivel de Sistema:**

A continuación se muestran los requisitos funcionales y no funcionales del sistema que van a ser probados, teniendo en cuenta la integración de todos los módulos.

**Requisitos Funcionales**

**RF23:** Mostrar la presentación particular de la aplicación.

**RF24:** Permitir desde la pantalla principal acceder a cada uno de los módulos.

**RF25:** Permitir activar o desactivar el sonido desde cualquier pantalla de la aplicación.

**RF26:** Permitir salir de la aplicación desde cualquier pantalla.

**RF27:** Permitir desde cualquier pantalla acceder a la pantalla principal.

**RF28:** Permitir navegar dentro de cada módulo mediante los botones: siguiente y anterior.

**RF29:** Permitir desde cualquier módulo acceder a cualquiera de los módulos restantes de la aplicación.

**Requisitos No Funcionales**

***De Portabilidad***

**RNF1:** Permitir que la aplicación funcione en el sistema operativo: Windows 98 o superior.

**RNF2:** Permitir que la aplicación funcione en el sistema operativo: Mac.

**RNF3:** Permitir que la aplicación funcione en el sistema operativo: Linux.

***De Diseño***

**RNF4:** Lograr uniformidad en todas las pantallas de la multimedia.

**RNF5:** Lograr orientar al usuario mediante la navegación por la multimedia.

## 2.3.2 Diseño de Prueba

Para la realización del diseño de las pruebas, se tuvieron en cuenta las pruebas a realizar por cada uno de los niveles.

### 2.3.2.1 Pruebas del Nivel de Unidad

#### 1. Prueba de Funcionalidad

*Objetivos:* El objetivo de las pruebas funcionales en el nivel de unidad va a estar basado en que se pruebe la correcta interacción con la interfaz de los módulos, además de que se le den cumplimiento a los requisitos funcionales que se especifiquen que se desean probar.

*Método:* Se va a aplicar el Método de Caja Negra, utilizando la Técnica de la Partición de Equivalencia, la cual divide el campo de entrada en clases de datos (válidas e inválidas) que tienden a ejercitar determinadas funciones del software.

*Requisitos:* RF1, RF2, RF3, RF4, RF5, RF6, RF7, RF8, RF9, RF10, RF11, RF12, RF13, RF14, RF15, RF16, RF17, RF18, RF19, RF20, RF21, RF22.

*Criterio:* Si el resultado de la prueba es *Satisfactorio* entonces se puede decir que el software pasó la prueba. En caso de que sea *Insatisfactorio*, significa que el software no pasó la prueba.

### 2.3.2.2 Pruebas del Nivel de Sistema

#### 1. Prueba de Funcionalidad

*Objetivo:* El objetivo de las pruebas funcionales en el nivel de sistema va a estar basado en comprobar que se cumplan los requisitos funcionales que se especifiquen que se desean probar.

*Método:* Se va a aplicar el Método de Caja Negra, utilizando la Técnica de la Partición de Equivalencia, la cual divide el campo de entrada en clases de datos (válidas e inválidas) que tienden a ejercitar determinadas funciones del software.

*Requisitos:* RF23, RF24, RF25, RF26, RF27, RF28, RF29.

*Criterio:* Si el resultado de la prueba es *Satisfactorio* entonces se puede decir que el software pasó la prueba. En caso de que sea *Insatisfactorio*, significa que el software no pasó la prueba.

## **2. Prueba de Portabilidad**

*Objetivo:* El objetivo de las pruebas de portabilidad se basa en comprobar que la multimedia funciona sin ninguna dificultad en los Sistemas Operativos: Windows 98 o superior, Mac y Linux. Es decir, que sea multiplataforma.

*Método:* Para aplicar esta prueba no se va a utilizar el método de Caja Negra ni el de Caja Blanca. Se creará un caso de prueba donde se especifiquen los requisitos que se desean probar, el resultado que se espera observar al probar en la aplicación cada requisito y el resultado de esa prueba.

*Requisitos:* RNF1, RNF2, RNF3.

*Criterio:* Si el resultado de la prueba es *Satisfactorio* entonces se puede decir que el software pasó la prueba. En caso de que sea *Insatisfactorio*, significa que el software no pasó la prueba.

## **3. Prueba de Diseño**

*Objetivo:* El objetivo de las pruebas de diseño se basa en comprobar que la multimedia posea una interfaz amigable para el usuario, fácil de usar, clara, sencilla y mantenga el mismo formato en todas sus partes.

*Método:* Para aplicar esta prueba no se va a utilizar el método de Caja Negra ni el de Caja Blanca. Se creará un caso de prueba donde se especifiquen los requisitos que se desean probar, el resultado que se espera observar al probar en la aplicación cada requisito y el resultado de esa prueba.

*Requisitos:* RNF4, RNF5.

*Criterio:* Si el resultado de la prueba es *Satisfactorio* entonces se puede decir que el software pasó la prueba. En caso de que sea *Insatisfactorio*, significa que el software no pasó la prueba.

### 2.3.3 Casos de Prueba

Para la realización de los casos de prueba, se dividieron los mismos en dependencia del nivel y la prueba que se vaya a realizar.

En este capítulo se dejaron solo los casos de prueba que emitieron alguna no conformidad de la multimedia, los restantes se puede ver en el **Anexo 2** del documento.

#### 2.3.3.1 Casos de Prueba Funcionales del Nivel de Unidad

##### Casos de Prueba Funcionales del Módulo 1

**Tabla 4:** CP2: Mostrar el significado de: canteros, cosechas, abono y estiércoles.

<b>Requisito</b>	RF1
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que al posicionar el mouse encima de las palabras: canteros, cosechas, abono y estiércoles aparezca una ventanita con su significado, el cual debe mostrarse sin errores gramaticales u ortográficos.
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Cultivos Organopónicos Definición e Importancia”, luego posicionar el mouse encima de las palabras: canteros, cosechas, abono o estiércoles.
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1 y al vínculo: “Cultivos Organopónicos Definición e

Importancia".				
Iteraciones realizadas al aplicar el Método de Prueba				
Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba	
1. Posicionar el mouse encima de la palabra: "canteros" y se muestre una ventanita.		1.1 Se muestre en la ventanita el significado de la palabra: "canteros".	1.1 Satisfactorio	
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Insatisfactorio	
		1.3 El significado mostrado sea el correcto.	1.3 Satisfactorio	
	2. Posicionar el mouse encima de la palabra: "canteros" y no se muestre la ventanita.		2. Insatisfactorio	
	1. Posicionar el mouse encima de la palabra: "cosechas" y se muestre una ventanita.		1.1 Se muestre en la ventanita el significado de la palabra: "cosechas".	1.1 Satisfactorio
			1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Insatisfactorio
1.3 El significado mostrado sea el correcto.			1.3 Satisfactorio	
2. Posicionar el mouse		2. Insatisfactorio		

	<p>encima de la palabra: “cosechas” y no se muestre la ventanita.</p>		
<p><b>1.</b> Posicionar el mouse encima de la palabra: “abono” y se muestre una ventanita.</p>		<p><b>1.1</b> Se muestre en la ventanita el significado de la palabra: “abono”.</p>	<p><b>1.1</b> Satisfactorio</p>
		<p><b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.</p>	<p><b>1.2</b> Satisfactorio</p>
		<p><b>1.3</b> El significado mostrado sea el correcto.</p>	<p><b>1.3</b> Satisfactorio</p>
	<p><b>2.</b> Posicionar el mouse encima de la palabra: “abono” y no se muestre la ventanita.</p>		<p><b>2.</b> Insatisfactorio</p>
<p><b>1.</b> Posicionar el mouse encima de la palabra: “estiércoles” y se muestre una ventanita.</p>		<p><b>1.1</b> Se muestre en la ventanita el significado de la palabra: “estiércoles”.</p>	<p><b>1.1</b> Satisfactorio</p>
		<p><b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.</p>	<p><b>1.2</b> Insatisfactorio</p>
		<p><b>1.3</b> El significado mostrado sea el correcto.</p>	<p><b>1.3</b> Satisfactorio</p>
	<p><b>2.</b> Posicionar el mouse encima de la palabra: “estiércoles” y no se</p>		<p><b>2.</b> Insatisfactorio</p>

muestre la ventanita.

**Tabla 5:** CP3: Mostrar el contenido que se aborda en: Materiales y Herramientas.

<b>Requisito</b>	RF2		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Materiales y Herramientas”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Materiales y Herramientas’ y poder acceder al mismo.		1.1 Se muestren los materiales y herramientas.	1.1 Satisfactorio
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Satisfactorio
		1.3 Aparezca la locución.	1.3 Satisfactorio
		1.4 La locución se corresponda con el texto.	1.4 Insatisfactorio
	2. Hacer clic en el vínculo: “Materiales y Herramientas” y no		

poder acceder al mismo.

**Tabla 6:** CP4: Permitir relacionar el nombre de los materiales y herramientas con la imagen correspondiente.

<b>Requisito</b>	RF2		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede corresponder el nombre de los materiales y herramientas con sus respectivas imágenes.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Materiales y Herramientas”, luego seleccionar el material o herramienta que se desea conocer.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1 y al vínculo: “Materiales y Herramientas”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en cualquier nombre de material o herramientas y activar alguna imagen.		1.1 Se active la imagen correspondiente al nombre del material o herramienta seleccionado.	1.1 Insatisfactorio
	2. Hacer clic en el cualquier nombre de material o herramientas y no activar ninguna imagen.		2. Insatisfactorio

**Tabla 7:** CP5: Mostrar el contenido que se aborda en: Preparación del Terreno.

<b>Requisito</b>	RF3		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Preparación del Terreno”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Preparación del Terreno” y poder acceder al mismo.		1.1 Se muestre la información referente a la preparación del terreno.	1.1 Satisfactorio
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Satisfactorio
		1.3 Aparezca la locución.	1.3 Insatisfactorio
		1.4 La locución se corresponda con el texto.	1.4 -----
	2. Hacer clic en el vínculo: “Preparación del Terreno” y no poder acceder al mismo.		

**Tabla 8:** CP7: Mostrar las orientaciones de la actividad: Complete la definición.

<b>Requisito</b>	RF4		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder a la actividad, se muestre la orientación de la misma tanto en forma de locución como de texto, concuerde el texto con la locución, no hayan errores gramaticales u ortográficos, las orientaciones sean las correctas para esa actividad.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Actividades” y luego en “Complete la definición”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1 y al vínculo: “Actividades”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Complete la definición” y poder acceder al mismo.		<b>1.1</b> Aparezca la locución explicando las orientaciones de la actividad.	<b>1.1</b> Satisfactorio
		<b>1.2</b> Aparezcan las orientaciones de la actividad en forma de texto.	<b>1.2</b> Satisfactorio
		<b>1.3</b> La locución se corresponda con el texto.	<b>1.3</b> Satisfactorio
		<b>1.4</b> Las orientaciones no contengan ningún tipo de error gramatical u	<b>1.4</b> Satisfactorio

	ortográfico.	
	<b>1.5</b> Las orientaciones mostradas sean las correctas y se expliquen bien todos los pasos a seguir en la realización de la misma.	<b>1.5</b> Insatisfactorio
<b>2.</b> Hacer clic en el vínculo: “Complete la definición” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio

**Tabla 9:** CP8: Permitir realizar la actividad: Complete la definición.

<b>Requisito</b>	RF4		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede realizar la actividad sin ningún inconveniente y que la aplicación sepa reconocer cuando la realización de la misma está correcta y cuando no.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Actividades” y luego en “Complete la definición”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1, al vínculo: “Actividades” y dentro del mismo al vínculo: “Complete la definición”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Seleccionar una		<b>1.1</b> Al colocar la palabra en la	<b>1.1</b> Insatisfactorio

<p>palabra con el mouse y poder arrastrarla.</p>	<p>definición, la aplicación reconozca correctamente cuando la actividad está bien o mal realizada.</p> <p><b>1.2</b> Cuando esté mal realizada la actividad se active el botón: Inténtalo nuevamente.</p> <p><b>1.3</b> Cuando esté bien realizada la actividad se active el cartel: Muy bien.</p> <p><b>2.</b> Seleccionar una palabra con el mouse y no poder arrastrarla.</p>		<p><b>1.2</b> Satisfactorio</p> <p><b>1.3</b> Satisfactorio</p> <p><b>2.</b> Insatisfactorio</p>
--	---	--	--

**Tabla 10:** CP10: Mostrar las orientaciones de la actividad: Pareo.

<p><b>Requisito</b></p>	<p>RF5</p>		
<p><b>Objetivos</b></p>	<p>Los objetivos de este caso de prueba son: Probar acceder a la actividad, se muestre la orientación de la misma tanto en forma de locución como de texto, concuerde el texto con la locución, no hayan errores gramaticales u ortográficos, las orientaciones sean las correctas para esa actividad.</p>		
<p><b>Pasos</b></p>	<p>Una vez dentro del módulo 1, hacer clic en: “Actividades” y luego en “Pareo”.</p>		
<p><b>Condiciones de Ejecución</b></p>	<p>Poder acceder al módulo 1 y al vínculo: “Actividades”.</p>		
<p><b>Iteraciones realizadas al aplicar el Método de Prueba</b></p>			
<p><b>Clase Válida</b></p>	<p><b>Clase Inválida</b></p>	<p><b>Resultado Esperado</b></p>	<p><b>Resultado de la</b></p>

		Prueba
<p><b>1.</b> Hacer clic en el vínculo: “Pareo” y poder acceder al mismo.</p>	<p><b>1.1</b> Aparezca la locución explicando las orientaciones de la actividad.</p>	<p><b>1.1</b> Satisfactorio</p>
	<p><b>1.2</b> Aparezcan las orientaciones de la actividad en forma de texto.</p>	<p><b>1.2</b> Satisfactorio</p>
	<p><b>1.3</b> La locución se corresponda con el texto.</p>	<p><b>1.3</b> Satisfactorio</p>
	<p><b>1.4</b> Las orientaciones no contengan ningún tipo de error gramatical u ortográfico.</p>	<p><b>1.4</b> Satisfactorio</p>
	<p><b>1.5</b> Las orientaciones mostradas sean las correctas y se expliquen bien todos los pasos a seguir en la realización de la misma.</p>	<p><b>1.5</b> Insatisfactorio</p>
<p><b>2.</b> Hacer clic en el vínculo: “Pareo” y no poder acceder al mismo.</p>		<p><b>2.</b> Insatisfactorio</p>

**Tabla 11:** CP11: Permitir realizar la actividad: Pareo.

<b>Requisito</b>	RF5		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede realizar la actividad sin ningún inconveniente y que la aplicación sepa reconocer cuando la realización de la misma está correcta y cuando no.		
<b>Pasos</b>	Una vez dentro del módulo 1, hacer clic en: “Actividades” y luego en “Pareo”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 1 y al vínculo: “Actividades” y dentro del mismo al vínculo: “Pareo”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Seleccionar una imagen con el mouse y poder arrastrarla.		1.1 Al colocar la imagen en el cuadro de la definición, la aplicación reconozca correctamente cuando la actividad está bien o mal realizada.	1.1 Satisfactorio
		1.2 Cuando esté mal realizada la actividad se active el botón: Inténtalo nuevamente.	1.2 Insatisfactorio
		1.3 Cuando esté bien realizada la actividad se active el cartel: Muy bien.	1.3 Satisfactorio
	2. Seleccionar una		2. Insatisfactorio

imagen con el mouse y  
no poder arrastrarla.

### Casos de Prueba Funcionales del Módulo 2

**Tabla 12:** CP13: Mostrar el contenido que se aborda en: Compostero (Definición e importancia).

<b>Requisito</b>	RF6		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Compostero Definición e Importancia”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Compostero Definición e Importancia” y poder acceder al mismo.		1.1 Se muestre un subtítulo: “Compostero”.	1.1 Satisfactorio
	2. Hacer clic en el vínculo: “Compostero Definición e Importancia”		2. Insatisfactorio

	y no poder acceder al mismo.		
<b>3.</b>	Hacer clic en el subtítulo: “Compostero” y poder acceder al contenido.	<b>3.1</b> Se muestre la definición e importancia del compostero. <b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico. <b>3.3</b> Aparezca la locución. <b>3.4</b> La locución se corresponda con el texto.	<b>3.1</b> Satisfactorio <b>3.2</b> Satisfactorio <b>3.3</b> Satisfactorio <b>3.4</b> Insatisfactorio
<b>4.</b>	Hacer clic en el subtítulo: “Compostero” y no poder acceder al contenido.		<b>4.</b> Insatisfactorio

**Tabla 13:** CP14: Mostrar el significado de: abonera, estiércoles, abono orgánico, compost.

<b>Requisito</b>	RF6
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que al posicionar el mouse encima de las palabras: abonera, estiércoles, abono orgánico, compost aparezca una ventanita con su significado, el cual debe mostrarse sin errores gramaticales u ortográficos.
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Compostero Definición e Importancia”, luego posicionar el mouse encima de las palabras: abonera, estiércoles, abono orgánico o compost.

Condiciones de Ejecución		Poder acceder al módulo 2 y al vínculo: “Compostero Definición e Importancia”	
Iteraciones realizadas al aplicar el Método de Prueba			
Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
1. Posicionar el mouse encima de la palabra: “abonera” y se muestre una ventanita.		1.1 Se muestre en la ventanita el significado de la palabra: “abonera”.	1.1 Satisfactorio
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Satisfactorio
		1.3 El significado mostrado sea el correcto.	1.3 Satisfactorio
	2. Posicionar el mouse encima de la palabra: “abonera” y no se muestre la ventanita.		2. Insatisfactorio
1. Posicionar el mouse encima de la palabra: “estiércoles” y se muestre una ventanita.		1.1 Se muestre en la ventanita el significado de la palabra: “estiércoles”.	1.1 Satisfactorio
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Insatisfactorio
		1.3 El significado mostrado sea el correcto.	1.3 Satisfactorio
	2. Posicionar el mouse		2. Insatisfactorio

	encima de la palabra: “estiércoles” y no se muestre la ventanita.		
<b>1.</b>	Posicionar el mouse encima de la palabra: “abono orgánico” y se muestre una ventanita.	<b>1.1</b> Se muestre en la ventanita el significado de la palabra: “abono orgánico”.	<b>1.1</b> Satisfactorio
		<b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>1.2</b> Insatisfactorio
		<b>1.3</b> El significado mostrado sea el correcto.	<b>1.3</b> Satisfactorio
	<b>2.</b> Posicionar el mouse encima de la palabra: “abono orgánico” y no se muestre la ventanita.		<b>2.</b> Insatisfactorio
<b>1.</b>	Posicionar el mouse encima de la palabra: “compost” y se muestre una ventanita.	<b>1.1</b> Se muestre en la ventanita el significado de la palabra: “compost”.	<b>1.1</b> Satisfactorio
		<b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>1.2</b> Satisfactorio
		<b>1.3</b> El significado mostrado sea el correcto.	<b>1.3</b> Satisfactorio
	<b>2.</b> Posicionar el mouse encima de la palabra:		<b>2.</b> Insatisfactorio

“compost” y no se  
muestre la ventanita.

**Tabla 14:** CP16: Mostrar el significado de: demarca, material orgánico, capas.

<b>Requisito</b>	RF7		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que al posicionar el mouse encima de las palabras: demarca, material orgánico y capas aparezca una ventanita con su significado, el cual debe mostrarse sin errores gramaticales u ortográficos.		
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Compostero Proceso de Elaboración”, luego posicionar el mouse encima de las palabras: demarca, material orgánico, capas.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2 y al vínculo: “Compostero Proceso de Elaboración”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Posicionar el mouse encima de la palabra: “demarca” y se muestre una ventanita.		1.1 Se muestre en la ventanita el significado de la palabra: “demarca”.	1.1 Satisfactorio
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 Satisfactorio
		1.3 El significado mostrado	1.3 Satisfactorio

		sea el correcto.	
	<b>2.</b> Posicionar el mouse encima de la palabra: “demarca” y no se muestre la ventanita.		<b>2.</b> Insatisfactorio
<b>1.</b> Posicionar el mouse encima de la palabra: “material orgánico” y se muestre una ventanita.		<b>1.1</b> Se muestre en la ventanita el significado de la palabra: “material orgánico”.	<b>1.1</b> Satisfactorio
		<b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>1.2</b> Satisfactorio
		<b>1.3</b> El significado mostrado sea el correcto.	<b>1.3</b> Satisfactorio
	<b>2.</b> Posicionar el mouse encima de la palabra: “material orgánico” y se muestre la ventanita		<b>2.</b> Insatisfactorio
<b>1.</b> Posicionar el mouse encima de la palabra: “capas” y se muestre una ventanita.		<b>1.1</b> Se muestre en la ventanita el significado de la palabra: “capas”.	<b>1.1</b> Satisfactorio
		<b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>1.2</b> Insatisfactorio
		<b>1.3</b> El significado mostrado sea el correcto.	<b>1.3</b> Satisfactorio

<p><b>2.</b> Posicionar el mouse encima de la palabra: “capas” y no se muestre la ventanita.</p>	<p><b>2.</b> Insatisfactorio</p>
--	----------------------------------

**Tabla 15:** CP19: Mostrar el contenido que se aborda en: Canteros (Definición e Importancia).

<b>Requisito</b>	RF9		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en Canteros: “Definición e Importancia”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<p><b>1.</b> Hacer clic en el vínculo: “Canteros Definición e Importancia” y poder acceder al mismo.</p>	<p><b>1.1</b> Se muestre un subtítulo: “Canteros”.</p>	<p><b>1.1</b> Satisfactorio</p>	<p><b>1.1</b> Satisfactorio</p>
<p><b>2.</b> Hacer clic en el vínculo: “Canteros Definición e Importancia”</p>	<p><b>2.</b> Hacer clic en el vínculo: “Canteros Definición e Importancia”</p>	<p><b>2.</b> Insatisfactorio</p>	<p><b>2.</b> Insatisfactorio</p>

	y no poder acceder al mismo.		
<b>3.</b> Hacer clic en el subtítulo: “Canteros” y poder acceder al contenido.		<b>3.1</b> Se muestre la definición e importancia del cantero.	<b>3.1</b> Satisfactorio
		<b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>3.2</b> Satisfactorio
		<b>3.3</b> Aparezca la locución.	<b>3.3</b> Satisfactorio
		<b>3.4</b> La locución se corresponda con el texto.	<b>3.4</b> Insatisfactorio
	<b>4.</b> Hacer clic en el subtítulo: “Canteros” y no poder acceder al contenido.		<b>4.</b> Insatisfactorio

**Tabla 16:** CP22: Mostrar el contenido que se aborda en: Semillero (Definición e Importancia).

<b>Requisito</b>	RF11
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Semillero Definición e Importancia”.
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2.
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>	

Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
<p><b>1.</b> Hacer clic en el vínculo: “Semillero Definición e Importancia” y poder acceder al mismo.</p>		<p><b>1.1</b> Se muestre el subtítulo “Semillero”.</p>	<p><b>1.1</b> Satisfactorio</p>
	<p><b>2.</b> Hacer clic en el vínculo: “Semillero Definición e Importancia” y no poder acceder al mismo.</p>		
<p><b>3.</b> Posicionar el mouse encima del subtítulo: “Semillero” y se despliegue el contenido.</p>		<p><b>3.1</b> Se muestre la definición e importancia del semillero.</p>	<p><b>3.1</b> Satisfactorio</p>
		<p><b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.</p>	<p><b>3.2</b> Satisfactorio</p>
		<p><b>3.3</b> Aparezca la locución.</p>	<p><b>3.3</b> Satisfactorio</p>
		<p><b>3.4</b> La locución se corresponda con el texto.</p>	<p><b>3.4</b> Insatisfactorio</p>
	<p><b>4.</b> Posicionar el mouse encima del subtítulo: “Semillero” y no se despliegue el contenido.</p>		<p><b>4.</b> Insatisfactorio</p>

**Tabla 17:** CP26: Mostrar las orientaciones de la actividad: Elabora virtualmente un compostero.

<b>Requisito</b>	RF13		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder a la actividad, se muestre la orientación de la misma tanto en forma de locución como de texto, concuerde el texto con la locución, no hayan errores gramaticales u ortográficos, las orientaciones sean las correctas para esa actividad.		
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Actividades” y luego en “Elabora virtualmente un compostero”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2 y al vínculo: “Actividades”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Elabora virtualmente un compostero” y poder acceder al mismo.		<b>1.1</b> Aparezca la locución explicando las orientaciones de la actividad.	<b>1.1</b> Satisfactorio
		<b>1.2</b> Aparezcan las orientaciones de la actividad en forma de texto.	<b>1.2</b> Satisfactorio
		<b>1.3</b> La locución se corresponda con el texto.	<b>1.3</b> Satisfactorio
		<b>1.4</b> Las orientaciones no contengan ningún tipo de error gramatical u	<b>1.4</b> Satisfactorio

	ortográfico.	
	<b>1.5</b> Las orientaciones mostradas sean las correctas y se expliquen bien todos los pasos a seguir en la realización de la misma.	<b>1.5</b> Insatisfactorio
<b>2.</b> Hacer clic en el vínculo: “Elabora virtualmente un compostero” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio

**Tabla 18:** CP27: Permitir realizar la actividad: Elabora virtualmente un compostero.

<b>Requisito</b>	RF13		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede realizar la actividad sin ningún inconveniente y que la aplicación sepa reconocer cuando la realización de la misma está correcta y cuando no.		
<b>Pasos</b>	Una vez dentro del módulo 2 hacer clic en: “Actividades” y luego en “Elabora virtualmente un compostero”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2, al vínculo: “Actividades” y dentro del mismo al vínculo: “Elabora virtualmente un compostero”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>

<p><b>1.</b> Hacer clic en el botón: “Jugar” y poder acceder a la actividad.</p>	<p><b>1.1</b> Se muestre la actividad para comenzar su realización.</p>	<p><b>1.1</b> Satisfactorio</p>
<p><b>2.</b> Hacer clic en el botón: “Jugar” y no poder acceder a la actividad.</p>		<p><b>2.</b> Insatisfactoria</p>
<p><b>3.</b> Seleccionar un material con el mouse y poder arrastrarlo.</p>	<p><b>3.1</b> Al colocar el material en el compostero, la aplicación reconozca correctamente cuando la actividad está bien o mal realizada.</p>	<p><b>3.1</b> Satisfactorio</p>
	<p><b>3.2</b> Cuando haya realizado varios intentos errados, permita intentarlo nuevamente.</p>	<p><b>3.2</b> Insatisfactorio</p>
	<p><b>3.3</b> Cuando esté bien realizada la actividad se muestre el cartel: Muy bien.</p>	<p><b>3.3</b> Satisfactorio</p>
<p><b>4.</b> Seleccionar un material con el mouse y no poder arrastrarlo.</p>		<p><b>4.</b> Insatisfactorio</p>

**Tabla 19:** CP29: Mostrar las orientaciones de la actividad: Sopa de letras.

<b>Requisito</b>	RF14		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder a la actividad, se muestre la orientación de la misma tanto en forma de locución como de texto, concuerde el texto con la locución, no hayan errores gramaticales u ortográficos, las orientaciones sean las correctas para esa actividad.		
<b>Pasos</b>	Una vez dentro del módulo 2, hacer clic en: “Actividades” y luego en “Sopa de letras”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2 y al vínculo: “Actividades”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en el vínculo: “Sopa de letras” y poder acceder al mismo.		1.1 Aparezca la locución explicando las orientaciones de la actividad.	1.1 Satisfactorio
		1.2 Aparezcan las orientaciones de la actividad en forma de texto.	1.2 Insatisfactorio
		1.3 La locución se corresponda con el texto.	1.3 Insatisfactorio
		1.4 Las orientaciones no contengan ningún tipo de error gramatical u	1.4 Satisfactorio

	ortográfico.	
	<b>1.5</b> Las orientaciones mostradas sean las correctas y se expliquen bien todos los pasos a seguir en la realización de la misma.	<b>1.5</b> Insatisfactorio
<b>2.</b> Hacer clic en el vínculo: “Sopa de letras” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio

**Tabla 20:** CP30: Permitir realizar la actividad: Sopa de letras.

<b>Requisito</b>	RF14		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede realizar la actividad sin ningún inconveniente y que la aplicación sepa reconocer cuando la realización de la misma está correcta y cuando no.		
<b>Pasos</b>	Una vez dentro del módulo 2 hacer clic en: “Actividades” y luego en “Sopa de letras”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 2, al vínculo: “Actividades” y dentro del mismo al vínculo: “Sopa de letras”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Seleccionar la		<b>1.1</b> Cambie el color de la	<b>1.1</b> Satisfactorio

<p>primera letra y la última de la palabra encontrada.</p>	<p>palabra encontrada en la fila izquierda de nombre: Palabras a buscar.</p> <p><b>1.2</b> Al seleccionar varias palabras no se formen otras que no existen.</p> <p><b>1.3</b> Al encontrar todas las palabras muestre un cartel reconociendo que se realizó bien la actividad.</p>	<p><b>1.2</b> Insatisfactorio</p> <p><b>1.3</b> Insatisfactorio</p>
<p><b>2.</b> No poder seleccionar la primera letra y la última de la palabra encontrada.</p>		<p><b>2.</b> Insatisfactorio</p>

**Casos de Prueba Funcionales del Módulo 3**

**Tabla 21:** CP32: Mostrar el contenido que se aborda en: Transplante del semillero al cantero.

<p><b>Requisito</b></p>	<p>RF15</p>
<p><b>Objetivos</b></p>	<p>Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.</p>
<p><b>Pasos</b></p>	<p>Una vez dentro del módulo 3, hacer clic en: “Transplante”.</p>
<p><b>Condiciones de Ejecución</b></p>	<p>Poder acceder al módulo 3.</p>
<p><b>Iteraciones realizadas al aplicar el Método de Prueba</b></p>	

Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
<p><b>1.</b> Hacer clic en el vínculo: “Transplante” y poder acceder al mismo.</p>		<b>1.1</b> Se muestre el concepto de transplante.	<b>1.1</b> Satisfactorio
		<b>1.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>1.2</b> Insatisfactorio
		<b>1.3</b> Aparezca la locución.	<b>1.3</b> Satisfactorio
		<b>1.4</b> La locución se corresponda con el texto.	<b>1.4</b> Insatisfactorio
	<p><b>2.</b> Hacer clic en el vínculo: “Transplante” y no poder acceder al mismo.</p>		<b>2.</b> Insatisfactorio
<p><b>3.</b> Posicionar el mouse encima del subtítulo: “Transplante del semillero al cantero” y se despliegue el contenido.</p>		<b>3.1</b> Se muestre el proceso de transplante del semillero al cantero.	<b>3.1</b> Satisfactorio
		<b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>3.2</b> Satisfactorio
	<p><b>4.</b> Posicionar el mouse encima del subtítulo: “Transplante del semillero al cantero” y no se despliegue el contenido.</p>		<b>4.</b> Insatisfactorio

**Tabla 22:** CP33: Mostrar el contenido que se aborda en: Sistema de riego.

<b>Requisito</b>	RF16		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 3, hacer clic en: “Sistema de Riego”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 3.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Hacer clic en el vínculo: “Sistema de Riego” y poder acceder al mismo.		<b>1.1</b> Se muestre el subtítulo “Sistema de Riego”.	<b>1.1</b> Satisfactorio
	<b>2.</b> Hacer clic en el vínculo: “Sistema de Riego” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio
<b>3.</b> Posicionar el mouse encima del subtítulo: “Sistema de Riego” y se despliegue el contenido.		<b>3.1</b> Se muestre la descripción del sistema de riego. <b>3.2</b> La información no contenga ningún tipo de error gramatical u	<b>3.1</b> Satisfactorio <b>3.2</b> Insatisfactorio

	ortográfico.	
	<b>3.3</b> Aparezca la locución.	<b>3.3</b> Insatisfactorio
	<b>3.4</b> La locución se corresponda con el texto.	<b>3.4</b> -----
<b>4.</b> Posicionar el mouse encima del subtítulo: “Sistema de Riego” y no se despliegue el contenido.		<b>4.</b> Insatisfactorio

**Tabla 23:** CP35: Mostrar el contenido que se aborda en: Tipos de plagas.

<b>Requisito</b>	RF18		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 3, hacer clic en: “Tipos de Plagas”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 3.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Hacer clic en el vínculo: “Tipos de Plagas” y poder		<b>1.1</b> Se muestre un subtítulo con una pestañita.	<b>1.1</b> Satisfactorio

acceder al mismo.	<b>2.</b> Hacer clic en el vínculo: “Tipos de Plagas” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio
<b>3.</b> Hacer clic en la pestañita y poder acceder al contenido.		<b>3.1</b> Se muestren los tipos de plagas.	<b>3.1</b> Satisfactorio
		<b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.	<b>3.2</b> Satisfactorio
		<b>3.3</b> Aparezca la locución.	<b>3.3</b> Satisfactorio
		<b>3.4</b> La locución se corresponda con el texto.	<b>3.4</b> Insatisfactorio
	<b>4.</b> Hacer clic en la pestañita y no poder acceder al contenido.		<b>4.</b> Insatisfactorio

**Tabla 24:** CP36: Mostrar el contenido que se aborda en: Control de plagas.

<b>Requisito</b>	RF19
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.
<b>Pasos</b>	Una vez dentro del módulo 3, hacer clic en: “Control de Plagas”.
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 3.

Iteraciones realizadas al aplicar el Método de Prueba			
Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
1. Hacer clic en el vínculo: “Control de Plagas” y poder acceder al mismo.		1.1 Se muestre un subtítulo con una pestañita.	1.1 Satisfactorio
	2. Hacer clic en el vínculo: “Control de Plagas” y no poder acceder al mismo.		2. Insatisfactorio
3. Hacer clic en la pestañita y poder acceder al contenido.		3.1 Se muestren los productos que se atizan para el control de plagas.	3.1 Satisfactorio
		3.2 La información no contenga ningún tipo de error gramatical u ortográfico.	3.2 Satisfactorio
		3.3 Aparezca la locución.	3.3 Satisfactorio
		3.4 La locución se corresponda con el texto.	3.4 Insatisfactorio
	4. Hacer clic en la pestañita y no poder acceder al contenido.		4. Insatisfactorio

**Tabla 25:** CP37: Mostrar la actividad: Auto Evaluación.

<b>Requisito</b>	RF20		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede acceder a la actividad, se muestra la orientación de la misma tanto en forma de locución como de texto, concuerde el texto con la locución, no hayan errores gramaticales u ortográficos, las orientaciones sean las correctas para esa actividad.		
<b>Pasos</b>	Una vez dentro del módulo 3, hacer clic en: “Actividades”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 3.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Hacer clic en el vínculo: “Actividades” y poder acceder al mismo.		<b>1.1</b> Se muestre el subtítulo “Auto Evaluación”.	<b>1.1</b> Satisfactorio
	<b>2.</b> Hacer clic en el vínculo: “Actividades” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio
<b>3.</b> Posicionar el mouse encima del subtítulo: “Auto Evaluación” y poder observar el		<b>3.1</b> Observar las orientaciones de la actividad. <b>3.2</b> Aparezca la locución explicando las	<b>3.1</b> Satisfactorio <b>3.2</b> Insatisfactorio



**Tabla 26:** CP38: Permitir realizar la actividad: Auto Evaluación.

<b>Requisito</b>	RF20		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede realizar la actividad sin ningún inconveniente y que la aplicación sepa reconocer cuando la realización de la misma está correcta y cuando no.		
<b>Pasos</b>	Una vez dentro del módulo 3, hacer clic en: “Actividades”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 3 y al vínculo: “Actividades”.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Seleccionar la palabra deseada.		1.1 Al seleccionar todas las palabras (realizar todas las preguntas) la aplicación señale cuales están bien y cuales no (con una cruz las mal y una pleca las bien).	1.1 Satisfactorio
		1.2 Cuando todas se respondan correctamente la aplicación reconozca que ha ganado.	1.2 Insatisfactorio
	2. No poder seleccionar la palabra deseada.		2. Insatisfactorio

**Casos de Prueba Funcionales del Módulo 4**

**Tabla 27:** CP39: Mostrar el contenido que se aborda en: Recolección y clasificación de la cosecha.

<b>Requisito</b>	RF21		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.		
<b>Pasos</b>	Una vez dentro del módulo 4, hacer clic en: “Recolección y Clasificación de la Cosecha”.		
<b>Condiciones de Ejecución</b>	Poder acceder al módulo 4.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
<b>1.</b> Hacer clic en el vínculo: “Recolección y Clasificación de la Cosecha” y poder acceder al mismo.		<b>1.1</b> Se muestre el subtítulo “Recolección y Clasificación de la Cosecha de los Cultivos Organopónicos”.	<b>1.1</b> Satisfactorio
	<b>2.</b> Hacer clic en el vínculo: “Recolección y Clasificación de la Cosecha” y no poder acceder al mismo.		<b>2.</b> Insatisfactorio
<b>3.</b> Posicionar el mouse		<b>3.1</b> Se muestre el proceso de	<b>3.1</b> Satisfactorio

<p>encima del subtítulo:                  “Recolección y                  Clasificación de la                  Cosecha de los                  Cultivos                  Organopónicos” y se                  despliegan el                  contenido.</p> <p><b>4.</b> Posicionar el mouse                  encima del subtítulo:                  “Recolección y                  Clasificación de la                  Cosecha de los Cultivos                  Organopónicos” y no se                  despliegan el contenido.</p>	<p>recolección y clasificación                  de la cosecha.</p> <p><b>3.2</b> La información no                  contenga ningún tipo de                  error gramatical u                  ortográfico.</p> <p><b>3.3</b> Aparezca la locución.</p> <p><b>3.4</b> La locución se                  corresponda con el texto.</p>	<p><b>3.2</b> Satisfactorio</p> <p><b>3.3</b> Satisfactorio</p> <p><b>3.4</b> Insatisfactorio</p> <p><b>4.</b> Insatisfactorio</p>
---	---	--

**Tabla 28:** CP40: Mostrar el contenido que se aborda en: Embalaje y clasificación.

<p><b>Requisito</b></p>	<p>RF22</p>
<p><b>Objetivos</b></p>	<p>Los objetivos de este caso de prueba son: Probar acceder al contenido, el contenido se muestre sin errores gramaticales u ortográficos, la información que se muestre sea correcta, aparezca la locución al acceder al contenido y se corresponda con el mismo.</p>
<p><b>Pasos</b></p>	<p>Una vez dentro del módulo 4, hacer clic en: “Embalaje y Clasificación”.</p>
<p><b>Condiciones de Ejecución</b></p>	<p>Poder acceder al módulo 4.</p>
<p><b>Iteraciones realizadas al aplicar el Método de Prueba</b></p>	

Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
<p><b>1.</b> Hacer clic en el vínculo: “Embalaje y Clasificación” y poder acceder al mismo.</p>		<p><b>1.1</b> Se muestre el subtítulo “Embalaje y Comercialización de la Cosecha de los Cultivos Organopónicos”.</p>	<p><b>1.1</b> Insatisfactorio</p>
	<p><b>2.</b> Hacer clic en el vínculo: “Embalaje y Clasificación” y no poder acceder al mismo.</p>		
<p><b>3.</b> Posicionar el mouse encima del subtítulo: “Embalaje y Comercialización de la Cosecha de los Cultivos Organopónicos” y se despliegue el contenido.</p>		<p><b>3.1</b> Se muestre el proceso de recolección y clasificación de la cosecha.</p>	<p><b>3.1</b> Satisfactorio</p>
		<p><b>3.2</b> La información no contenga ningún tipo de error gramatical u ortográfico.</p>	<p><b>3.2</b> Satisfactorio</p>
		<p><b>3.3</b> Aparezca la locución.</p>	<p><b>3.3</b> Satisfactorio</p>
		<p><b>3.4</b> La locución se corresponda con el texto.</p>	<p><b>3.4</b> Insatisfactorio</p>
	<p><b>4.</b> Posicionar el mouse encima del subtítulo: “Embalaje y Comercialización de la Cosecha de los Cultivos Organopónicos” y no se despliegue el contenido.</p>		<p><b>4.</b> Insatisfactorio</p>

### 2.3.3.2 Casos de Prueba Funcionales del Nivel de Sistema

**Tabla 29:** CP41: Mostrar la presentación particular de la aplicación.

<b>Requisito</b>	RF23			
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar acceder a la multimedia una vez que esta sea ejecutada, se muestre una bienvenida e introducción al contenido de la misma, una vez que se esté en la pantalla principal indique como acceder a cada uno de los módulos por los que está compuesta.			
<b>Pasos</b>	Una vez que tengo el ejecutable de la multimedia tratar de acceder a la misma.			
<b>Condiciones de Ejecución</b>	Contar con el ejecutable de la multimedia.			
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>				
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>	
1. Ejecutar la multimedia.		1.1 Aparezca la bienvenida e introducción al contenido de la multimedia en forma de texto.	1.1 Insatisfactorio	
		1.2 La información no contenga ningún tipo de error gramatical u ortográfico.	1.2 -----	
		1.3 Aparezca la bienvenida e introducción al contenido de la multimedia en forma	1.3 Satisfactorio	

		de locución.	
		<b>1.4</b> La locución se corresponda con el texto.	<b>1.4</b> -----
	<b>2.</b> No poder ejecutar la multimedia.		<b>2.</b> Insatisfactorio
<b>3.</b> Hacer clic en cualquier lugar de la pantalla y poder acceder a la pantalla principal.		<b>3.1</b> Se muestre como poder comenzar a navegar a través de los módulos de la multimedia.	<b>3.1</b> Satisfactorio
	<b>4.</b> Hacer clic en cualquier lugar de la pantalla y no poder acceder a la pantalla principal.		<b>4.</b> Insatisfactorio

**Tabla 30:** CP43: Permitir activar o desactivar el sonido desde cualquier pantalla de la aplicación.

<b>Requisito</b>	RF25
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede activar o desactivar el sonido de la multimedia y que la imagen utilizada para realizar esta actividad (la bocinita) muestra siempre el estado en que se encuentra el sonido en ese momento (activado o desactivado).
<b>Pasos</b>	Una vez que se accedió a la multimedia entrar a todas las pantallas de la misma a probar el sonido.
<b>Condiciones de Ejecución</b>	Poder acceder a la multimedia y a todas sus pantallas.
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>	

Clase Válida	Clase Inválida	Resultado Esperado	Resultado de la Prueba
1. Hacer clic en la imagen del sonido (la bocinita) y poder desactivar o activar el sonido.		1.1 La imagen cambie de alguna manera para que muestre el estado en que se encuentra el sonido.	1.1 Insatisfactorio
	2. Hacer clic en la imagen del sonido (la bocinita) y no poder desactivar o activar el sonido.		2. Insatisfactorio

**Tabla 31:** CP44: Permitir salir de la aplicación desde cualquier pantalla.

<b>Requisito</b>	RF26
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que la aplicación permite salir de la misma desde cualquiera de sus pantallas y antes de realizar dicha actividad pregunte si está seguro de querer salir o no.
<b>Pasos</b>	Una vez que se accedió a la multimedia entrar a todas las pantallas de la misma a probar si se puede abandonar la aplicación, con la previa confirmación de salida.
<b>Condiciones de Ejecución</b>	Poder acceder a la multimedia y a todas sus pantallas.
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>	
<b>Clase Válida</b>	<b>Clase Inválida</b>
<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>

<p><b>1.</b> Hacer clic en el botón de cerrar y acceder a la pantalla de confirmación de salida.</p>	<p><b>2.</b> Hacer clic en el botón de cerrar y no acceder a la pantalla de confirmación de salida.</p>	<p><b>1.1</b> Aparezca en la pantalla el mensaje: “¿Seguro que desea abandonar la aplicación?”</p>	<p><b>1.1</b> -----</p> <p><b>2.</b> Satisfactorio</p>
<p><b>3.</b> Hacer clic en el botón de “Si” y acceder a la pantalla de los créditos.</p>	<p><b>4.</b> Hacer clic en el botón de “Si” y no acceder a la pantalla de los créditos.</p>	<p><b>3.1</b> Pasen todos los créditos y luego se cierre completamente la multimedia.</p>	<p><b>3.1</b> Satisfactorio</p> <p><b>4.</b> Insatisfactorio</p>
<p><b>3.</b> Hacer clic en el botón de “No” y retornar a la pantalla donde se encontraba.</p>	<p><b>4.</b> Hacer clic en el botón de “No” y no retornar a la pantalla donde se encontraba.</p>		<p><b>3.</b> Satisfactorio</p> <p><b>4.</b> Insatisfactorio</p>

**Tabla 32:** CP46: Permitir navegar dentro de cada módulo mediante los botones: siguiente y anterior.

<b>Requisito</b>	RF28		
<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que se puede navegar dentro de cada módulo mediante los botones: siguiente y anterior.		
<b>Pasos</b>	Una vez que se accedió a la multimedia entrar a todos los módulos de la misma a probar si se puede navegar a través de los botones: siguiente y anterior.		
<b>Condiciones de Ejecución</b>	Poder acceder a la multimedia y a todos sus módulos.		
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>			
<b>Clase Válida</b>	<b>Clase Inválida</b>	<b>Resultado Esperado</b>	<b>Resultado de la Prueba</b>
1. Hacer clic en la imagen del botón siguiente (la flechita que apunta hacia la derecha) y poder acceder a la próxima pantalla del módulo.			1. Insatisfactorio
	2. Hacer clic en la imagen del botón siguiente (la flechita que apunta hacia la derecha) y no poder acceder a la próxima pantalla del módulo.		2. Satisfactorio

<p><b>1.</b> Hacer clic en la imagen del botón anterior (la flechita que apunta hacia la izquierda) y poder acceder a la pantalla anterior del módulo.</p>	<p><b>1.</b> Satisfactorio</p>
<p><b>2.</b> Hacer clic en la imagen del botón anterior (la flechita que apunta hacia la izquierda) y no poder acceder a la pantalla anterior del módulo.</p>	<p><b>2.</b> Insatisfactorio</p>

### 2.3.3.3 Casos de Prueba de Portabilidad

**Tabla 33:** CP48: Permitir que la multimedia sea multiplataforma.

<p><b>Objetivos</b></p>	<p>Los objetivos de este caso de prueba son: Probar que se puede ejecutar la multimedia en los sistemas operativos: Windows 98 o superior, Mac y Linux y se sigan cumpliendo todas sus funcionalidades correctamente.</p>
<p><b>Pasos</b></p>	<p>Ejecutar la multimedia en una computadora que tenga instalado alguno de los sistemas operativos a probar.</p>
<p><b>Condiciones de Ejecución</b></p>	<p>Tener instalados los sistemas operativos: Windows 98 o superior, Mac y Linux en la computadora que voy a realizar la prueba.</p>
<p><b>Iteraciones realizadas al aplicar el Método de Prueba</b></p>	

Requisito	Resultado Esperado	Resultado de la Prueba
1. Permitir que la aplicación funcione en el sistema operativo: Windows 98 o superior.	1.1 Al ejecutar la multimedia en el sistema operativo Windows 98 o superior funcione correctamente.	1.1 Satisfactorio
2. Permitir que la aplicación funcione en el sistema operativo: Mac.	2.1 Al ejecutar la multimedia en el sistema operativo Mac funcione correctamente.	2.1 Insatisfactorio
3. Permitir que la aplicación funcione en el sistema operativo: Linux	3.1 Al ejecutar la multimedia en el sistema operativo Linux funcione correctamente.	3.1 Satisfactorio

### 2.3.3.4 Casos de Prueba de Diseño

**Tabla 34:** CP49: Mostrar una interfaz amigable.

<b>Objetivos</b>	Los objetivos de este caso de prueba son: Probar que en la multimedia no se dejan grandes espacios en blanco sin necesidad, los vínculos aunque estén en pantallas distintas funcionan mediante el mismo evento, al acercarme con el mouse a algún vínculo cambie de color o de forma para que el usuario se interese en pincharlo y que el usuario conozca siempre el lugar de la multimedia donde se encuentra.
<b>Pasos</b>	Acceder a todas las pantallas de la multimedia para comprobar la uniformidad y la orientación del usuario mediante la misma.
<b>Condiciones de Ejecución</b>	Poder acceder a todas las pantallas de la multimedia.
<b>Iteraciones realizadas al aplicar el Método de Prueba</b>	

Requisito	Resultado Esperado	Resultado de la Prueba
1. Lograr uniformidad en todas las pantallas de la multimedia.	1.1 En las pantallas no se muestren grandes espacios en blanco sin necesidad.	1.1 Insatisfactorio
	1.2 Todos los vínculos que tengan correspondencia (los que aunque están en pantallas distintas realizan la misma actividad) funcionen mediante el mismo evento.	1.2 Insatisfactorio
	1.3 Todos los botones que tengan correspondencia (los que aunque están en pantallas distintas realizan la misma actividad) tengan el mismo tamaño, forma y color.	1.3 Satisfactorio
2. Lograr orientar al usuario mediante la navegación por la multimedia.	2.1 Al acercarme con el mouse a algún vínculo, el mismo cambie de forma, de color o se active de alguna manera que llame la atención del usuario.	2.1 Insatisfactorio
	2.2 Que el usuario conozca siempre el módulo en el que se encuentra y el contenido que está observando.	2.2 Satisfactorio

## 2.4 Conclusiones

En este capítulo se realizó una descripción de la multimedia “Organopónico” y luego de adecuado el proceso de pruebas a la multimedia se comenzaron a aplicar las actividades definidas.

Se puede decir que tanto el Plan de Prueba, como el Diseño de Prueba y la elección de los Casos de Prueba a ejecutar, tienen una gran importancia y son imprescindible para lograr el éxito del proceso de pruebas. Luego de la realización de estas actividades, se llegó a las siguientes conclusiones:

- Primeramente se deben de probar los módulos individuales y luego se integran para formar el sistema.
- Las pruebas imprescindibles son aquellas que prueban el cumplimiento de los requisitos de la multimedia.
- El objetivo del proceso de pruebas es descubrir errores en la multimedia, no la ausencia de ellos.
- Los casos de prueba más eficientes son aquellos que detecten mayor número de errores en su ejecución.

3

## Resultados del Proceso de Pruebas

### Introducción

En el proceso de pruebas es un requisito indispensable llevar toda la documentación pertinente, principalmente los defectos que se vayan encontrando, ya que encontrar la mayor cantidad posible antes que el cliente y el usuario final es de vital importancia, además si no se registran adecuadamente le resultaría muy complicado a los desarrolladores su corrección.

En este capítulo se registran y analizan los defectos encontrados en la multimedia “Organopónico” al ejecutar los casos de prueba. Además se calculan algunos costos en los que se incurrió durante la realización del proceso de pruebas.

### 3.1 Registro de No Conformidades

**Tabla 35: No Conformidades del Nivel de Unidad**

No.	No Conformidad	Lugar	CP	Importancia
1	Aparece una palabra mal escrita: “ <i>orgnaopónico</i> ”.	Módulo 1, pantalla: Definición e Importancia, posicionar el mouse en: “ <i>canteros</i> ”.	CP#2	Baja
2	Aparece una escritura en bloques: “ <i>cosechade</i> ”.	Módulo 1, pantalla: Definición e Importancia, posicionar el mouse en: “ <i>cosechas</i> ”.	CP#2	Baja

3	Aparece un error de concordancia: “ <i>Es la material residual</i> ”.	Módulo 1, pantalla: Definición e Importancia, posicionar el mouse en: “ <i>estiércoles</i> ”.	CP#2	Baja
4	La locución no se corresponde con el texto. En la locución se dice que se debe hacer clic sobre las imágenes para conocer la identificación y la multimedia funciona al revés. Además la utilidad de los materiales y herramientas solo aparecen en forma de locución.	Módulo 1, pantalla: Materiales y Herramientas.	CP#3	Alta
5	Se confundieron a la hora de relacionar los aislantes plásticos y el nylon (trocaron las imágenes).	Módulo 1, pantalla: Materiales y Herramientas.	CP#4	Alta
6	No aparece la locución.	Módulo 1, pantalla: Preparación del Terreno.	CP#5	Baja
7	Al orientar la actividad no se explica la respuesta del sistema para que el usuario sepa que no hizo bien la actividad.	Módulo 1, pantalla: Actividades, actividad: Complete la definición.	CP#7	Baja
8	Existe un caso en que la actividad está bien respondida y el sistema no lo reconoce, ya que para él no es lo mismo decir: “ <i>canteros o contenedores</i> ” que “ <i>contenedores o canteros</i> ”, solo reconoce como que está bien, la segunda opción.	Módulo 1, pantalla: Actividades, actividad: Complete la definición.	CP#8	Alta

<b>9</b>	Al orientar la actividad no se explica la respuesta de la aplicación para que el usuario sepa que no hizo bien la actividad.	Módulo 1, pantalla: Actividades, actividad: Pareo.	CP#10	Baja
<b>10</b>	Cuando está mal realizada la actividad se activa el botón: “Inténtalo Nuevamente”, pero automáticamente la aplicación resuelve la actividad, por lo que no tendría lógica volver a intentarlo, si ya se sabe la respuesta correcta.	Módulo 1, pantalla: Actividades, actividad: Pareo.	CP#11	Baja
<b>11</b>	La locución no se corresponde con el texto. En la locución se habla de la importancia del compostero y en el texto dan la definición solamente.	Módulo 2, pantalla: Definición e Importancia.	CP#13	Alta
<b>12</b>	Aparece un error de concordancia: “ <i>Es la material residual</i> ”.	Módulo 2, pantalla: Compostero Definición e Importancia, posicionar el mouse en: “estiércoles”.	CP#14	Baja
<b>13</b>	Se comieron una letra: “ <i>Es aquel que e prepara</i> ”.	Módulo 2, pantalla: Compostero Definición e Importancia, posicionar el mouse en: “abono”.	CP#14	Baja
<b>14</b>	Aparece un error de concordancia: “ <i>Es la capa superficial de la tierra sobre</i> ”.	Módulo 2, pantalla: Compostero Proceso de Elaboración, posicionar el mouse en: “capas”.	CP#16	Baja
<b>15</b>	La locución no se corresponde con el texto.	Módulo 2, pantalla:	CP#19	Alta

		Canteros Definición e Importancia.		
<b>16</b>	La locución no se corresponde con el texto. En la locución explican la importancia y en el texto la definición.	Módulo 2, pantalla: Semillero Definición e Importancia.	CP#22	Alta
<b>17</b>	Al orientar la actividad no se explica la respuesta de la aplicación para que el usuario sepa que no hizo bien la actividad.	Módulo 2, pantalla: Actividades, actividad: Elabora virtualmente un compostero.	CP#26	Baja
<b>18</b>	Cuando se realizan varios intentos errados se activa el botón: “Inténtalo Nuevamente”, pero automáticamente la aplicación resuelve la actividad, por lo que no tendría lógica volver a intentarlo, si ya se sabe la respuesta correcta.	Módulo 2, pantalla: Actividades, actividad: Elabora virtualmente un compostero.	CP#27	Baja
<b>19</b>	No aparecen las orientaciones de la actividad en forma de texto. Solo aparecen, en forma de texto, las palabras a buscar, lo demás está solamente en la locución.	Módulo 2, pantalla: Actividades, actividad: Sopa de letras.	CP#29	Alta
<b>20</b>	En la locución se plantea que para seleccionar las palabras hay que sombrearlas y no es así como funciona, hay que marcar la primera letra y la última.	Módulo 2, pantalla: Actividades, actividad: Sopa de letras.	CP#29	Alta
<b>21</b>	La sopa de letras no está bien realizada, pues al encontrar un conjunto de palabras se forman otras que no existen. (ver	Módulo 2, pantalla: Actividades, actividad: Sopa de letras.	CP#30	Baja

<b>Anexo 3)</b>				
<b>22</b>	Al encontrar todas las palabras no se muestra un cartel reconociendo que se realizó bien la actividad.	Módulo 2, pantalla: Actividades, actividad: Sopa de letras.	CP#30	Baja
<b>23</b>	Aparece un error de concordancia: “ <i>Es el proceso mediante la cual</i> ”.	Módulo 3, pantalla: Transplantes.	CP#32	Baja
<b>24</b>	En la locución plantean que hay que hacer clic sobre el subtítulo: “Transplante del semillero al cantero” y la aplicación no funciona de esta manera, sino posicionando el mouse encima del subtítulo.	Módulo 3, pantalla: Transplantes.	CP#32	Baja
<b>25</b>	Al final del texto omitieron una palabra: ( <i>máxima protección</i> ).	Módulo 3, pantalla: Sistema de Riego.	CP#33	Baja
<b>26</b>	No Aparece la locución	Módulo 3, pantalla: Sistema de Riego.	CP#33	Baja
<b>27</b>	La locución no se corresponde con el texto. En la locución se saltan párrafos completos del texto, además algunas veces intercalan la información.	Módulo 3, pantalla: Tipos de plagas.	CP#35	Alta
<b>28</b>	La locución no se corresponde con el texto. La locución se salta pedazos y al final sigue explicando contenido que no está en el texto.	Módulo 3, pantalla: Control de plagas.	CP#36	Alta
<b>29</b>				

	No aparece la locución explicando las orientaciones de la actividad.	Módulo 3, pantalla: Actividades, actividad: Auto Evaluación.	CP#37	Baja
<b>30</b>	Cuando todas las preguntas se responden correctamente la aplicación no reconoce que se ha ganado.	Módulo 3, pantalla: Actividades, actividad: Auto Evaluación.	CP#38	Baja
<b>31</b>	La locución no se corresponde con el texto. En la locución se habla de la recolección y en el texto de la clasificación, además la locución dice que se haga clic en el subtítulo y así no es como funciona.	Módulo 4, pantalla: Recolección y Clasificación de la Cosecha.	CP#39	Alta
<b>32</b>	El nombre del vínculo: “Embalaje y Clasificación” no coincide con el nombre del subtítulo del contenido a tratar, el cual se ve una vez que se accede al vínculo ( <i>no se habla de la Clasificación sino de Comercialización</i> ).	Módulo 4, pantalla: Embalaje y Clasificación.	CP#40	Baja
<b>33</b>	La locución no se corresponde con el texto. Al final el texto se acaba y la locución sigue brindando información del tema.	Módulo 4, pantalla: Embalaje y Clasificación.	CP#40	Alta

**Tabla 36: No Conformidades del Nivel de Sistema**

No.	No Conformidad	Lugar	CP	Importancia
34	No aparece la bienvenida e introducción al contenido de la multimedia en forma de texto.	Pantalla de presentación de la multimedia.	CP#41	Baja
35	La imagen del sonido (la bocinita) no cambia de ninguna manera que muestre que el sonido se encuentra desactivado o activado.	Todas las pantallas.	CP#43	Baja
36	Desde la pantalla de presentación de la multimedia, se puede salir sin confirmación.	Pantalla de presentación de la multimedia.	CP#44	Baja
37	En los módulos 1, 2, y 3 no se puede pasar a la pantalla: “Actividades” desde la pantalla anterior a ella, es decir, el botón: “Siguiente” no funciona cuando se está en este caso.	Módulos 1, 2 y 3.	CP#46	Baja
38	Al ejecutar la multimedia en el sistema operativo Mac se comienza a ver bien la multimedia y en cuanto entra a la locución de bienvenida se recoge hacia una esquina y no se puede ver a pantalla completa.	A partir de la pantalla de presentación de la multimedia.	CP#48	Baja
39	Se dejan grandes espacios en blanco en	Todas las pantallas.	CP#49	Baja

	la barra donde se encuentran los botones que dirigen al usuario hacia el contenido de la multimedia.			
<b>40</b>	Se dejan grandes espacios en blanco sin necesidad en las pantallas de los contenidos y se crean vínculos innecesarios al contenido, en vez de mostrarlo en esos mismos espacios.	Módulo 1, pantallas: Definición e Importancia y Actividades. Los restantes módulos en todas sus pantallas.	CP#49	Baja
<b>41</b>	En las pantallas se encuentran vínculos que realizan la misma función y lo hacen de forma diferentes: lo mismo haciendo clic, que acercando el mouse a ellos.	Módulo 1, pantallas: Definición e Importancia y Actividades. Los restantes módulos en todas sus pantallas.	CP#49	Baja
<b>42</b>	En la actividad: Elabora virtualmente un compostero, sale un cartel que dice: Inténtelo Nuevamente, el cual es un botón y cuando se acerca el mouse a él no cambia en nada. Lo mismo sucede en la Sopa de Letras con un botón de nombre: Ver Solución.	Módulo 2, pantalla: Actividades.	CP#49	Baja

### 3.2 Análisis de Resultados

Para poder realizar un análisis de los resultados obtenidos al aplicar el proceso de pruebas a la multimedia “Organopónico”, se va a tener en cuenta los resultados por cada uno de los niveles primeramente, para luego pasar a realizar un análisis general.

### 3.2.1 Resultados del Nivel de Unidad

**Tabla 37:** Resultados del Nivel de Unidad.

Módulo	Requisitos Probados	Casos de Prueba Total	Casos de Prueba Emiten Defectos	%	Defectos	Defectos Importancia Alta	%
1	5	12	8	66,7	10	3	30,0
2	9	19	9	47,4	12	5	41,6
3	6	7	6	85,7	8	2	25,0
4	2	2	2	100	3	2	66,7
<b>Totales</b>	<b>22</b>	<b>40</b>	<b>25</b>	<b>62,5</b>	<b>33</b>	<b>12</b>	<b>36,4</b>

Como se puede interpretar en la tabla anterior, en el módulo 2 fue donde más requisitos se probaron y por tanto es el que cuenta con una mayor cantidad de casos de prueba, sin embargo cuando se hace una comparación en % entre la totalidad de los casos de prueba y los casos de prueba que emitieron defectos, se puede observar que los módulos donde la realización de los casos de prueba fue más eficiente, fueron el 4 y luego el 3 y el menos eficiente fue el 2.

El módulo más defectuoso fue el 2 y además el que más defectos de alta importancia reportó, comparando en por ciento la totalidad de defectos y los defectos de importancia alta, se puede observar que el módulo 4 fue el de mayor porcentaje, pues el 66,7% de sus defectos fueron de alta importancia seguido por el módulo 2 con un 41,6%, lo cual quiere decir que en estos módulos casi todos los defectos encontrados son de alta importancia.

En general se puede decir que los casos de prueba realizados en el nivel de unidad fueron eficientes, ya que más de la mitad (un 62,5%) reportó errores en la aplicación, además de 33 defectos encontrados solo 12 fueron de importancia alta, para un 36,4%, lo cual significa menos de la mitad, por tanto se puede decir que en los módulos de la multimedia aunque existen errores no son catastróficos los resultados.

### 3.2.2 Resultados del Nivel de Sistema

**Tabla 38:** Resultados del Nivel de Sistema.

Prueba	Requisitos Probados	Casos de Prueba Total	Casos de Prueba Emiten Defectos	%	Defectos	Defectos Importancia Alta	%
Funcionalidad	7	7	4	57,1	4	0	0
Portabilidad	3	1	1	100	1	0	0
Diseño	2	1	1	100	4	0	0
<b>Totales</b>	<b>12</b>	<b>9</b>	<b>6</b>	<b>66,7</b>	<b>9</b>	<b>0</b>	<b>0</b>

Como se puede interpretar en la tabla anterior, en la prueba de funcionalidad fue donde más requisitos se probaron y por tanto para su realización se tuvieron que hacer más casos de prueba que en las restantes, no obstante los casos de prueba más eficientes en cuanto a una comparación porcentual entre la cantidad de casos de prueba total y los que emiten defectos fueron los de las pruebas de: Portabilidad y Diseño con un 100%.

En las pruebas que se realizaron en el nivel de sistema no se encontraron defectos de alta importancia para la aplicación, aunque se detectaron 9 defectos, por lo tanto se puede decir que los resultados fueron buenos. Además los casos de prueba de este nivel fueron eficientes, ya que el % entre los casos de prueba totales y los que emiten defectos es de un 66,7%, lo cual significa más de la mitad.

### 3.2.3 Resultados Generales

A la multimedia “Organopónico” se le realizaron pruebas de funcionalidad, portabilidad y diseño. Se probaron en general 34 requisitos, 29 funcionales y 5 no funcionales. Se realizaron en total 49 casos de prueba, de los cuales 31 emitieron defectos, lo cual significa un 63,3%. Se encontraron en total 42 defectos, de los cuales 12 fueron de alta importancia, lo cual significa un 28,6%.

Con todos estos datos se puede concluir, que no fueron muchos los errores de alta importancia que se encontraron en la multimedia, además los casos de prueba que se realizaron fueron eficientes. Al realizar una comparación entre el nivel de unidad y el de sistema se puede ver que los defectos de alta importancia para la multimedia “Organopónico” son precisamente los detectados en el nivel de unidad, de ahí la importancia de realizar un proceso de pruebas que en su planificación cuente con una buena estrategia, pues en el sistema no son muchos ni tan importantes los defectos que se detecten, sin restarles mucha importancia, pues al final todos influyen aunque en mayor o menor proporción, en la calidad de la multimedia.

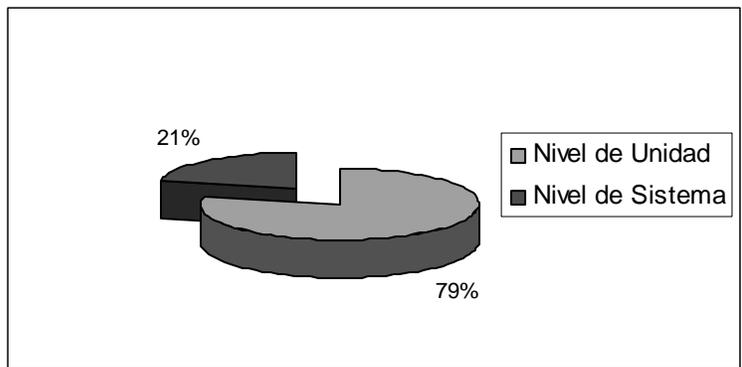


Figure 5: Defectos encontrados en la multimedia “Organopónico” por niveles.

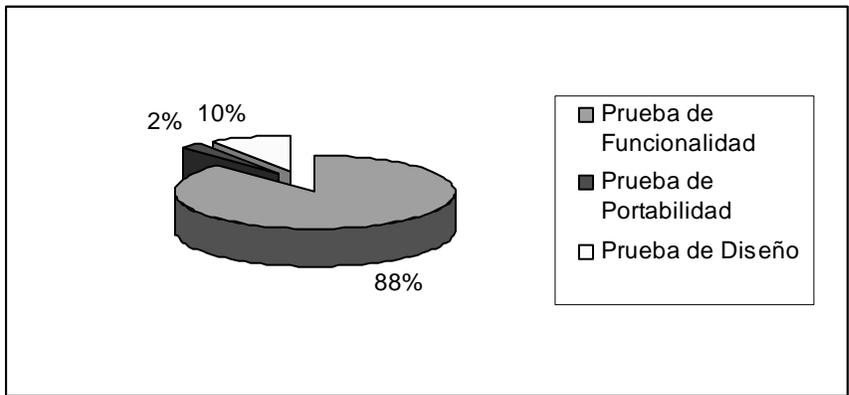


Figure 6: Defectos encontrados en la multimedia “Organopónico” por pruebas.

A continuación se muestra una tabla con un resumen de los defectos que más se repitieron, para lo cual se crearon 6 categorías:

- *Errores gramaticales u ortográficos:* En esta categoría se encuentran los defectos que se derivan de problema gramaticales en los textos de la aplicación, escritura en bloque, palabras mal escritas, entre otros.
- *Problemas con la locución:* Se divide en tres tipos de problemas con la locución, ya sea tanto en la concordancia de la misma con el texto, que no aparezca la locución en alguna pantalla o que no se cumpla alguna de las acciones que se orienten en la locución (errores en la locución).
- *Mala orientación de las actividades:* Los defectos que hacen al usuario perderse en el momento de realizar alguna actividad porque no se orientan bien los pasos para su realización.
- *La aplicación no reconoce la victoria del usuario al realizar bien una actividad:* Al realizar bien una actividad no se muestra ningún mensaje felicitando al usuario o expresándole que realizó bien la actividad.
- *Errores en la programación:* Los defectos de la multimedia que se derivan de un mal funcionamiento de la aplicación, es decir, que no se programaron de la forma correcta ciertos aspectos de la aplicación que provocaron problemas en algunas de sus funcionalidades.
- *Errores en el diseño:* Defectos que se derivaron de probar algunos aspectos de diseño a la multimedia.

**Tabla 39:** Defectos más comunes encontrados en la multimedia “Organopónico”.

Defectos más Comunes	Cantidad	No Conformidad
1. Errores gramaticales u ortográficos.	8	1, 2, 3, 12, 13, 14, 23, 25
2. Problemas con la locución:	15	
2.1 Problemas en la concordancia entre la locución y el texto.	8	4, 11, 15, 16, 27, 28, 31, 33
2.2 No aparece la locución.	3	6, 26, 29

2.3 Errores en la locución.	4	4, 20, 24, 31
3. Mala orientación de las actividades.	4	7, 9, 17, 19
4. La aplicación no reconoce la victoria del usuario al realizar bien una actividad.	2	22, 30
5. Errores en la programación.	6	5, 8, 10, 18, 36, 37
6. Errores en el diseño.	5	35, 39, 40, 41, 42

Como se puede ver en la tabla, los errores que más se cometieron en la realización de la multimedia “Organopónico” fueron: problemas con la locución, errores gramaticales u ortográficos y errores en la programación y el diseño.

Haciendo una breve comparación entre los resultados esperados y los reales que se obtuvieron, se puede decir que el software realiza sus funcionalidades principales, pero no de la forma esperada o de la forma más óptima para el usuario, por lo que se puede plantear que la multimedia “Organopónico” no cuenta con la calidad requerida para su exportación.

### 3.3 Cálculo de Costos

Para calcular los costos en que se incurrió para la realización del proceso de pruebas, se tuvieron en cuenta los gastos de recursos materiales, los gastos de salario y los gastos de electricidad.

#### Gastos de Recursos Materiales (G.RM)

Para calcular los gastos en que se incurrió en cuanto a recursos materiales, se realizó una tabla (tabla 40), donde se calculó primeramente el monto de cada recurso ( $Monto = Cantidad * Precio$ ) y luego se sumaron todos para calcular el gasto de recursos materiales ( $G.RM = \sum Monto = Monto_1 + Monto_2 = \dots Monto_N$ ).

**Tabla 40:** Gastos de Recursos Materiales

Recursos	Cantidad	Precio(\$)	Monto(\$)
Libreta	1	5.00	5.00
Lapicero	2	5.00	10.00
Hojas	100	1.50	150.00
CD/RW	1	25.00	25.00
<b>G.RM</b>	-----	-----	<b>190.00</b>

**Gastos de Electricidad (G.E)**

Para la realización del proceso de pruebas a la multimedia “Organopónico” se utilizó como único recurso consumidor de energía eléctrica, 1 computadora, la cual consume 150 watts/hora y 1 watts equivale a \$0,0016.

Para calcular el gasto energético, primeramente se calcula la cantidad de horas que se trabajó en la computadora cada mes ( $Horas\_Mensuales = Horas\_Diarias * Días$ ), luego se sumaron para calcular el total de horas que se trabajó en los 7 meses, después se calculó el consumo total de energía en esos meses ( $Cosumo\_Total = Total\_Horas\_Mensuales * 150$ ) y se calculó finalmente el gasto energético ( $G.E = Cosumo\_Total * 0,0016$ )

**Tabla 41:** Cálculo del total de horas trabajadas

Meses	Dic.	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Total
Días	8	13	20	22	21	23	3	110
Horas_Diarias	8	8	8	8	8	8	8	8
Horas_Mensuales	64	104	160	176	168	184	24	880

$$\text{Consumo\_Total} = 880 * 150$$

$$\text{Consumo\_Total} = 132\ 000 \text{ watts}$$

$$G.E = 132\ 000 * 0,0016$$

$$\mathbf{G.E} = \$ 211,2$$

### **Gasto de Salario (G.S)**

El proceso de pruebas fue realizado por una sola persona, que contaba con un salario mínimo mensual de \$150.00 por cuestiones de estipendio, por tanto el gasto salarial en el que incurrió fue:

$$G.S = \text{Cant\_Peronas} * \text{Salario\_Mensual} * \text{Meses\_Trabajo}$$

$$G.S = 1 * 150 * 5$$

$$\mathbf{G.S} = \$750$$

### **Costo Total (C.T)**

$$C.T = G.RM + G.E + G.S$$

$$C.T = 190 + 211,2 + 750$$

$$\mathbf{C.T} = \$1151,2$$

### **Análisis de Costos y Beneficios**

Luego de calcular los costos en que se debe incurrir para la realización del proceso de pruebas antes descrito, se puede concluir que una suma de \$1151,2 no es poco, pero si necesario para lograr mantener el prestigio de la Universidad y la confianza de los clientes. Además mientras más calidad tenga los productos, más caros podrán ser comercializados, y se podrán contrarrestar estos costos y obtener mayores ganancias. Por tanto se considera factible el proceso de pruebas.

### **3.4 Conclusiones**

En este capítulo se documentaron los resultados arrojados de la aplicación del proceso de pruebas a la multimedia “Organopónico”, lo cual sirve como base para argumentar en el análisis de los resultados, que a pesar de que no se hayan encontrado defectos críticos a la multimedia, la misma no posee la calidad requerida para su exportación.

También se hizo un breve estudio de la factibilidad del proceso realizado, donde se concluyó que aunque resultase costoso el proceso de pruebas a una multimedia, era factible gracias a todos los beneficios que se pueden obtener de él.

### **Conclusiones**

Mediante el estudio realizado a lo largo de este trabajo y los resultados alcanzados al aplicarle un proceso de pruebas al producto multimedia “Organopónico”, se ha podido llegar a las siguientes conclusiones:

- Los procesos de prueba son aplicables a los productos multimedia, solo depende de su adaptación a las características específicas del producto a probar.
- Se pueden realizar Casos de Prueba eficientes, mediante el Método de Caja Negra y la Técnica de Partición de Equivalencia a las multimedia.
- Mediante la realización de un buen proceso de pruebas quedan documentados todos los entregables que servirán de apoyo para la corrección de los defectos por parte de los desarrolladores del software y para la evaluación de la calidad del mismo.
- El proceso de pruebas realizado al producto multimedia “Organopónico” permitió detectar un gran número de defectos en el producto que no fueron previsto por sus desarrolladores.

## Recomendaciones

Luego de realizar el proceso de pruebas a la multimedia “Organopónico” se recomienda:

- Comenzar a realizar procesos de prueba, como el antes expuesto, mediante la etapa de desarrollo de los productos multimedia, pero teniendo en cuenta los cuatro niveles fundamentales de prueba (Unidad, Integración, Sistema y Aceptación).
- Realizar, además de las pruebas de Caja Negra, pruebas en el Nivel de Unidad mediante el Método de Caja Blanca, que permitan detectar errores en el código de la multimedia.
- Utilizar los defectos encontrados para lograr que los desarrolladores de productos multimedia adquieran conocimiento de los posibles errores que pueden cometerse durante el desarrollo de la multimedia que influyen en la calidad del producto final.
- Generalizar el documento obtenido mediante los Cursos Optativos de Calidad, principalmente el de Pruebas de Software.

## Referencias Bibliográficas

1. BEIZER, B. *Software System Testing and Quality Assurance*. New York: Van Nostrand Reinhold, 1984. 358 p.
2. CUEVA, J. M. *Calidad del Software* [Consultado el: 8 de febrero de 2007]. Disponible en: [http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad\\_software.PDF](http://gidis.ing.unlpam.edu.ar/downloads/pdfs/Calidad_software.PDF)
3. DE ANTONIO, A. *Gestión, control y garantía de la calidad del software* [Consultado el: 7 de febrero de 2007]. Disponible en: [http://www.inf.uach.cl/rvega/ asignaturas/info265/G\\_Calidad.pdf](http://www.inf.uach.cl/rvega/ asignaturas/info265/G_Calidad.pdf).
4. ESTRADA, T. *Pruebas a productos multimedia en el CIGET (Centro de Información y Gestión Tecnológica)*.
5. FERNÁNDEZ-MEDINA, E. *Pruebas del Software* [Consultado el: 10 de febrero de 2007]. Disponible en: <http://alarcos.inf-cr.uclm.es/doc/ISOFTWAREI/Tema09.pdf>.
6. GÓMEZ, L. *Estrategia de Calidad - ParqueSoft* [Consultado el: 24 de noviembre de 2006]. Disponible en: <http://www.greensqa.com/archivos/GSQA-Estrategia%20Calidad%20ParqueSoft.pdf>.
7. GUERRERO, G. M. *Pruebas a productos multimedia en CITMATEL*.
8. IEEE. *Standard Glossary of Software Engineering Terminology*. IEEE Software Engineering Standard Collection, 1990, nº ISSN 610.12-190.
9. ISO/IEC. International Standard. *Software Life Cycle Processes*. International Standard Organization/International Electronic Commite, 1995.
10. LOPEZ, J. A. *Pruebas del Software* [Consultado el: 5 de abril de 2007]. Disponible en: [http://dis.um.es/~lopezquesada/documentos/Tema\\_6.ppt#317,1,Ingeniería\\_del\\_Software](http://dis.um.es/~lopezquesada/documentos/Tema_6.ppt#317,1,Ingeniería_del_Software).

11. MEDINA, G. *Pruebas a productos multimedia en la UCI*.
12. MYERS, G. J. *The Art of Software Testing* New York: John Wiley & Sons, 1979. 192 p.
13. NAPAL, I. B. I. *Las pruebas de software, su aplicación el Config. Case*. Instituto Superior Politécnico José Antonio Echeverría, 2003.
14. PRESSMAN, R. S. *Ingeniería del Software: Un enfoque práctico*. México, DF: McGraw-Hill, 1993. 824p.
15. PRESSMAN, R. S. *Ingeniería del Software: Un enfoque práctico*. México, DF: McGraw-Hill, 2000.
16. PRESSMAN, R. S. *Ingeniería del Software. Un enfoque práctico. Parte 1*. La Habana: Félix Varela, 2005. 601 p.

## Bibliografía

1. COLLADO, M. *Pruebas de software*. [Consultado el: 31 de marzo de 2007]. Disponible en: <http://lml.ls.fi.upm.es/ftp/ed2/0203/Apuntes/pruebas.ppt>.
2. CONCEJERO, J. G. L. C. P. *Calidad de servicio percibida por el usuario en sistemas de telecomunicación*. [Consultado el: 20 de enero de 2007]. Disponible en: [http://www.tid.es/documentos/boletin/numero10\\_4.pdf](http://www.tid.es/documentos/boletin/numero10_4.pdf).
3. CRESPO, R. M. C. L. Y. *Estudio de la distribución docente de pruebas del software y refactoring para la incorporación de metodologías ágiles*. [Consultado el: 15 de marzo de 2007]. Disponible en: <http://bioinfo.uib.es/~joemiro/semDOC/Jen2004/ponencias/ponencia28.pdf>.
4. FLORES, A. D. K. M. L. *Determinación de los Requerimientos de Calidad del Producto Software Basados en Normas Internacionales*. [Consultado el: 7 de febrero de 2007]. Disponible en: [http://www.ewh.ieee.org/reg/9/etrans/vol4issue2April2006/4TLA2\\_6Davila.pdf](http://www.ewh.ieee.org/reg/9/etrans/vol4issue2April2006/4TLA2_6Davila.pdf).
5. GÚZMAN, O. H. *Aplicación práctica del diseño de pruebas de software a nivel de programación*. [Consultado el: 10 de febrero de 2007]. Disponible en: [http://www.willydev.net/descargas/oguzman-diseno\\_pruebas.pdf](http://www.willydev.net/descargas/oguzman-diseno_pruebas.pdf).
6. MEJIAS, J. J. G. M. J. E. M. *Pruebas del sistema en programación extrema*. [Consultado el: 7 de febrero de 2007]. Disponible en: [http://www.lsi.us.es/~javierj/investigacion\\_ficheros/PSISEXTREMA.pdf](http://www.lsi.us.es/~javierj/investigacion_ficheros/PSISEXTREMA.pdf).
7. MENDOZA, A. G. M. P. L. *Estrategia de Pruebas para Software OO que garantiza Requerimientos No Funcionales*. [Consultado el: 15 de marzo de 2007]. Disponible en: [http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad\\_09.pdf](http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_09.pdf).

8. POLO, M. *Mantenimiento Avanzado de Sistemas de Información. Pruebas del Software*. [Consultado el: 20 de enero de 2007]. Disponible en: <http://alarcos.inf-cr.uclm.es/doc/masi/doc/lec/parte5/polo-apuntesp5.pdf>.
9. REINA, J. G. M. E. M. M. A. *Modelo de pruebas para pruebas del sistema*. [Consultado el: 8 de febrero de 2007]. Disponible en: <http://www.lsi.us.es/~javierj/publications/MDA14.pdf>.
10. ROJAS, A. G. L. M. M. P. T. *Modelo de calidad de software educativo: Aplicación del estándar ISO/IEC 9126*. [Consultado el: 15 de marzo de 2007]. Disponible en: [http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad\\_32.pdf](http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_32.pdf).
11. ROJAS, M. O. M. P. T. *CONSTRUCTION OF A SYSTEMIC QUALITY MODEL FOR EVALUATING A SOFTWARE PRODUCT*. [Consultado el: 15 de marzo de 2007]. Disponible en: [http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad\\_10.pdf](http://www.lisi.usb.ve/publicaciones/02%20calidad%20sistemica/calidad_10.pdf).
12. ROMERO, V. A. *Implementación de una arquitectura de tres modelos para crear multimedia educativa*. [Consultado el: 15 de marzo de 2007]. Disponible en: <http://www.nonio.uminho.pt/challenges/05comunicacoes/Tema11/09VicenteZaldivar.pdf>.
13. TORRES, J. G. M. E. A. T. M. M. J. *Hacia una propuesta de pruebas tempranas del sistema*. [Consultado el: 15 de marzo de 2007]. Disponible en: <http://in2test.lsi.uniovi.es/pris2006/PRIS2006-GutierrezEscalonaMejiasTorres.pdf>.
14. VALENCIA, G. A. *Subcontratación del servicio de pruebas de software*. [Consultado el: 8 de febrero de 2007]. Disponible en: <http://www.greensga.com/archivos/Servicio%20Pruebas%20de%20Software.pdf>.

---

## Glosario de Términos

1. *Requisitos*: Conjunto de características que debe tener un producto o servicio para satisfacer las necesidades y expectativas del cliente.
2. *Requisitos Funcionales*: Los requerimientos funcionales especifican acciones que el sistema debe ser capaz de ejecutar sin tener en cuenta restricciones físicas.
3. *Requisitos No Funcionales*: Los requerimientos no funcionales describen atributos del sistema o del ambiente del sistema ayudando a determinar las propiedades que debe tener la aplicación.
4. *Proceso*: Un proceso es un conjunto de actividades o eventos que se realizan o suceden con un determinado fin.
5. *Multimedia*: Material digitalizado que combina textos, gráficos, imagen fija y en movimiento, así como sonido.
6. *Software*: La parte "que no se puede tocar" de un ordenador: los programas y los datos.
7. *Hardware*: Equipo físico del ordenador, formado por todos los periféricos (ratón, teclado, pantalla, etc.), circuitos, procesadores, etc.
8. *Endógeno*: Referente al interior o interno del organismo.
9. *Producto*: Un producto es cualquier objeto que puede ser ofrecido a un mercado que pueda satisfacer un deseo o una necesidad.
10. *Estrategia*: El arte de coordinar las acciones y de maniobrar para alcanzar un objetivo.
11. *Regresión*: Retorno a un estado o etapa anterior.

---

**12. Usuario:** El usuario es la persona que consume o usa el producto, bien o servicio.

**13. Cliente:** Quien habitualmente sostiene con una empresa comercial relaciones de demanda. Persona física o jurídica que realiza Transacciones de compra-venta de Contratos en el Mercado.