

# **Universidad Nacional de Ingeniería**

Facultad de Ingeniería Industrial y de Sistemas



## **IMPLEMENTACION DE UN PROCESO DE HOMOLOGACIÓN DE SISTEMAS EN LA SUPERINTENDENCIA NACIONAL DE ADMINISTRACIÓN TRIBUTARIA**

Informe de Suficiencia

Para optar el Título Profesional de:

**INGENIERO DE SISTEMAS**

**CESAR AUGUSTO CHAVARRY ARIAS**

Lima - Perú  
2002

## DEDICATORIA

*A mis padres por las cosas buenas que me dieron e incluso por las que  
desearon y no pudieron darme.*

*Por su esfuerzo y sacrificio en darme una buena educación,  
a partir de la cual surge todo lo demás.*

*A mi adorada esposa Yulisa por su apoyo incondicional, por su amor e inteligencia  
que alimentó la realización de este esfuerzo por titularme,  
y soportarme cuando estuve ausente,  
aun estando en casa.*

*A mi bebé Sebastian, que con su nacimiento  
me ha llenado de energía  
para seguir siendo cada día mejor.*

*A la memoria de mi bisabuela Clarita por enseñarme  
desde chico el gusto de hacer las cosas bien hechas.*

*A mi abuelita Lidia por su ejemplo de trabajo  
y buen humor.*

## **AGRADECIMIENTO**

*Mi agradecimiento profundo a todos mis profesores de la facultad por las enseñanzas, conocimiento y experiencias compartidas.*

*A las autoridades y personal administrativo de la facultad por su apoyo en este esfuerzo de ayudarnos a lograr el título.*

*A las autoridades de mi centro de trabajo por el apoyo prestado y por las oportunidades brindadas para poner en práctica mis conocimientos y seguir así desarrollarme como profesional.*

*A mis suegros por el apoyo brindado en la crianza de mi bebé durante sus primeros meses de nacimiento, pudiendo tener el tiempo necesario para cumplir con mis labores profesionales y sobre todo culminar con el presente informe.*

*Mi agradecimiento a todos mis amigos que honrándome con su confianza y su amistad me han valorado y alentado, sumando sus energías a las mías para poner la piedra final en mi titulación como ingeniero.*

# INDICE

	Página
DEDICATORIA.....	i
AGRADECIMIENTO.....	ii
INDICE .....	3
DESCRIPTORES TEMÁTICOS.....	5
RESUMEN EJECUTIVO .....	6
INTRODUCCIÓN.....	8
CAPITULO I: ANTECEDENTES .....	10
1.1 Estructura de SUNAT .....	10
1.2 Estructura de la Intendencia Nacional de Sistemas de Información .....	13
1.3 Diagnóstico Estratégico de la Gerencia de Homologación de Sistemas .....	15
CAPITULO II: MARCO TEORICO .....	17
2.1 Definición del proceso de pruebas de software .....	17
2.2 Evolución del enfoque de pruebas de software .....	17
2.3 Técnicas de pruebas .....	18
2.3.1 Revisiones e Inspecciones.....	18
2.3.2 Pruebas de Caja Negra .....	19
2.3.3 Pruebas de Caja Blanca .....	20
2.3.4 Pruebas de Caja Gris .....	22
CAPITULO III: PROCESO DE TOMA DE DECISIONES .....	26
3.1 Planteamiento del problema .....	26
3.2 Cuantificación del problema .....	27
3.3 Solución Planteada .....	28
3.4 Metodología actual de Desarrollo de Software .....	29
3.5 Visión sistémica del proceso de homologación .....	32
3.6 Proceso de Homologación de Sistemas .....	33
3.6.1 Reconocimiento del Sistemas .....	34
3.6.2 Planificación del proceso de Homologación .....	35
3.6.3 Homologación del Sistema .....	37
3.6.4 Evaluación de Resultados .....	40
CAPITULO IV: EVALUACIÓN DE RESULTADOS .....	42

CAPITULO V: CONCLUSIONES .....	45
BIBLIOGRAFÍA .....	47
ANEXOS: .....	48
ANEXO N°1: Tipos de Pruebas .....	48
ANEXO N°2: Formato - Plan de Homologación .....	53
ANEXO N°3: Formato - Caso de Pruebas .....	56
ANEXO N°4: Formato - Reporte de Errores en Pruebas .....	57
ANEXO N°5: Formato - Informe de Homologación .....	58

## **DESCRIPTORES TEMATICOS**

- Homologación de Sistemas
- Calidad de Software
- Pruebas de Sistemas
- Pruebas de Software
- Fallas de Sistemas
- Metodología de Pruebas
- Pruebas de Caja Negra
- Pruebas de Caja Blanca
- Pruebas de Caja Gris
- Metodología de Desarrollo
- Clases de Equivalencia
- Análisis de Valores Límites

## **RESUMEN EJECUTIVO**

El uso de las Tecnologías de Información ha permitido que la Superintendencia Nacional de Administración Tributaria (SUNAT) logre avances considerables en los últimos años, llegando a ser considerada como una de las instituciones más eficientes del Estado Peruano. El punto de partida de mejoramiento fue el año 1993 y uno de los pilares fundamentales de este cambio fue diseñar un nuevo sistema de información integrado, que serviría de soporte al nuevo modelo tributario adoptado. Si bien se lograron desarrollar los principales sistemas de recaudación (función principal de SUNAT), este desarrollo se basó principalmente en la competencia y heroísmo del personal involucrado que hizo el trabajo poniendo como máxima prioridad el cronograma y sacrificando la calidad del sistema.

Este sacrificio de la calidad de los sistemas conllevó a una serie de ineficiencias en la arquitectura de los sistemas de SUNAT, los cuales reflejan altos costos de mantenimiento y gran cantidad de fallas reportadas. Una causa principal de todos estos problemas es la falta de una cultura de calidad y un proceso adecuado de verificación de los sistemas desarrollados antes de ponerlos a producción.

Dada esta problemática, se planteó la necesidad de implementar un proceso que se encargue de evaluar los criterios de calidad (funcionales y no-funcionales) de los sistemas y asegurar que los elementos informáticos, de procedimientos y legales estén alineados de acuerdo a las características contractuales (definiciones) del sistema desarrollado, dicho proceso fue denominado Proceso de Homologación de Sistemas.

La implantación del Proceso de Homologación de Sistemas en SUNAT, ha ido encaminado con una reestructuración organizacional de la Intendencia Nacional de Sistemas de Información (INSI), creándose una Gerencia de Homologación, la cual se encuentra al mismo nivel jerárquico que las Gerencias de Desarrollo de Sistemas. Esto ha permitido establecer y formalizar rápidamente la metodología de trabajo, explicada en el presente informe, y a partir de su puesta en funcionamiento se ha establecido un orden en el proceso de producción de sistemas

y una cultura que vela por la calidad de los sistemas en forma integral. El alcance del proceso de Homologación, abarcó tanto al componente informático y documentación técnica como a los aspectos legales que normalizan los procedimientos del sistema.

A lo largo del presente trabajo se describirá detalladamente el proceso de homologación de sistemas implementado y los resultados obtenidos, como la disminución de la tasa de errores reportados en los sistemas puestos en producción, se alcanzó una disminución del 35% de errores en el primer año de implementado el proceso de homologación.

Con la experiencia obtenida en la implementación de este tipo de proceso, podemos afirmar que toda organización que desarrolle o implemente un sistema informático debe contemplar un proceso formal de homologación para asegurar la calidad del sistema y minimizar la presencia de errores en los ambientes de producción.

## INTRODUCCIÓN

El objetivo del presente informe es describir el proceso de Homologación de Sistemas implementado en la Superintendencia Nacional de Administración Tributaria (SUNAT), como marco de un proceso de aseguramiento de la calidad de los distintos desarrollos informáticos de esta entidad. Esta iniciativa metodológica desarrollada en SUNAT, tiene por objetivo principal la verificación de la calidad de cada componente que conforma los sistemas de información y su homologación con los procedimientos internos a los que da soporte.

El desarrollo de sistemas de información implica una serie de actividades en las que las posibilidades de que aparezca el fallo humano son enormes. Los errores pueden empezar a darse desde el primer momento del proceso, en el que los requerimientos pueden estar especificados de forma errónea o ambigua, así como los posteriores pasos de diseño y desarrollo. Para garantizar la calidad de los sistemas es necesario establecer un proceso de pruebas, tanto de los requerimientos iniciales, los diseños y la codificación del software.

La prueba de sistemas, tiene diferentes perspectivas, una de las más conocidas es la perspectiva del desarrollador, la cual considera que las pruebas se realizan al final de proceso de desarrollo para demostrar que el sistema funciona perfectamente, sin embargo la visión más aceptada y que mejores resultados reportan es la perspectiva destructiva, la cual se basa en crear casos de pruebas para demostrar que el sistema falla. De hecho la prueba es uno de los pasos de la ingeniería de software que se puede ver (por lo menos, psicológicamente) como destructiva en lugar de constructiva (Glenford Myers [9]).

El uso de las Tecnologías de Información ha permitido que la SUNAT logre avances considerables en la década de los 90, logrando importantes avances en la administración de la recaudación de los tributos bajo su cargo. Así por ejemplo, se implementaron sistemas centrales como el Registro Único de Contribuyentes-RUC, el Sistema de Comprobantes de Pago, el Sistema de Recaudación de Principales Contribuyentes (PRICOS) y el Sistema de Recaudación para Medianos y Pequeños contribuyentes (MEPECOS). Los sistemas de información

permitieron la transformación de los procesos manuales en procesos automatizados, además estandarizaron el uso de nuevas técnicas y métodos de trabajo, generando la modernidad de toda la Administración Tributaria. Bajo este contexto la calidad de los sistemas informáticos se convierte en un factor crítico para aumentar la eficiencia organizacional.

## **CAPITULO I ANTECEDENTES**

La SUNAT es una institución pública descentralizada del Sector Economía y Finanzas, creada por Ley N° 24829, dotada de personería jurídica de Derecho Público, patrimonio propio y autonomía administrativa, funcional, técnica y financiera.

La SUNAT tiene por finalidad administrar y aplicar los procesos de recaudación y fiscalización de los tributos internos, así como proponer y participar en la reglamentación de las normas tributarias. Para ello cuenta con la Intendencia Nacional de Principales Contribuyentes, la Intendencia Nacional de Servicios al Contribuyente, la Intendencia Nacional de Cumplimiento Tributario, la Intendencia Nacional de Sistemas de Información, la Intendencia Nacional de Planeamiento, la Intendencia Nacional de Administración, la Intendencia Nacional Jurídica; 10 Intendencias Regionales (Lima, Arequipa, La Libertad, Lambayeque, Piura, Cusco, Ica, Tacna, Loreto y Junín) y 8 Oficinas Zonales (Huacho, Juliaca, Chimbote, Cajamarca, Cañete, Ucayali, San Martín y Huánuco), así como 21 Oficinas Remotas (Huaraz, Mollendo, Camaná, Puno, Madre de Dios, Abancay, Andahuaylas, Sicuani, Quillabamba, Ayacucho, Tarma, Huancavelica, Pasco, Pacasmayo, Chachapoyas, Jaén, Moyobamba, Tumbes, Talara, Moquegua e Ilo).

### **I.1 ESTRUCTURA DE LA SUNAT**

El Decreto Legislativo N° 639, promulgado el 18 de mayo de 1991, establecía la estructura orgánica básica de la SUNAT, la misma que fue ampliada a través de diversas resoluciones de superintendencia. La Resolución de Superintendencia N° 027-2001/SUNAT, promulgada el 14 de febrero de 2001, establecía la siguiente estructura organizacional de la SUNAT.

#### **ALTA DIRECCIÓN**

- Superintendente Nacional de administración Tributaria
- Superintendente Nacional Adjunto

#### **ORGANOS DE APOYO**

- Comité de Alta Dirección

- Secretaría General
- Instituto de Administración Tributaria
- Oficina de Control Interno

## **ORGANOS DE LINEA**

- **Intendencia Nacional de Administración**

Tiene por objetivo principal asegurar el funcionamiento de las diferentes unidades organizacionales que conforman la SUNAT, a través de la administración de los recursos internos disponibles.

- **Intendencia Nacional de Cumplimiento Tributario**

Tiene por objetivo elaborar los programas operativos y procedimientos técnicos, para controlar la deuda tributaria y los procesos de fiscalización. Así como asegurar el funcionamiento de las unidades operativas en el ámbito nacional, relevando las necesidades relacionadas a los sistemas, programas operativos, procedimientos técnicos y normas relacionadas a los temas de su competencia.

- **Intendencia Nacional Jurídica**

Elaborar proyectos de normas legales que debe emitir la SUNAT; así como otros proyectos de normas legales que sean requeridos por los órganos de la SUNAT. Interpretar el sentido y alcance de las normas tributarias y defender los intereses y derechos de la institución o de sus trabajadores emplazados en el ejercicio de sus funciones.

- **Intendencia Nacional de Planeamiento**

Tiene por objetivo identificar los objetivos institucionales, proponer las estrategias necesarias para su logro y elaborar los planes estratégicos; así mismo analizar y desarrollar las propuestas de política tributaria.

Efectúa el seguimiento y evalúa los indicadores de gestión de las unidades operativas, del cumplimiento del plan estratégico y de la situación presupuestaria de la institución.

- **Intendencia Nacional de Servicios al Contribuyente**

Tiene por objetivos promover el cumplimiento voluntario y la mejora permanente de los servicios personalizados al contribuyente a través de un óptimo diseño y administración de los programas y planes operativos relativos a dichos servicios. Asegurar el funcionamiento de las unidades operativas en el ámbito nacional, relevando las necesidades relacionadas a los sistemas, programas y procedimientos de los temas de su competencia, y medir el grado de satisfacción del contribuyente respecto de las acciones de la SUNAT.

- **Intendencia Nacional de Sistemas de Información**

Tiene por objetivo principal proveer a la institución de los sistemas, procedimientos e infraestructura informática requeridos para apoyar la labor de la Intendencia Nacional de Cumplimiento Tributario, la Intendencia de Servicios al Contribuyente y demás órganos de la institución.

- **Intendencia Nacional de Principales Contribuyentes**

Tiene por objetivo procurar la consecución de las metas de recaudación establecidas, a través de la aplicación de los sistemas y procedimientos de control de cumplimiento tributario y de servicios al contribuyente aprobados respecto a los contribuyentes que componen su directorio.

## **ORGANOS DESCONCENTRADOS**

- **Intendencias Regionales**

Tienen por objetivo asegurar la obtención de las metas de recaudación establecidas, a través de la aplicación de los sistemas y procedimientos de recaudación y fiscalización aprobados; así como mediante la ejecución de las acciones de atención y orientación de los contribuyentes de su jurisdicción.

- **Oficinas Zonales**

Tienen por objetivos asegurar la obtención de las metas de recaudación establecidas, a través de la aplicación de los sistemas y procedimientos de recaudación y fiscalización aprobados; así como mediante la ejecución de las acciones de atención y orientación de los contribuyentes de su jurisdicción.

El organigrama de SUNAT se muestra en la siguiente figura.

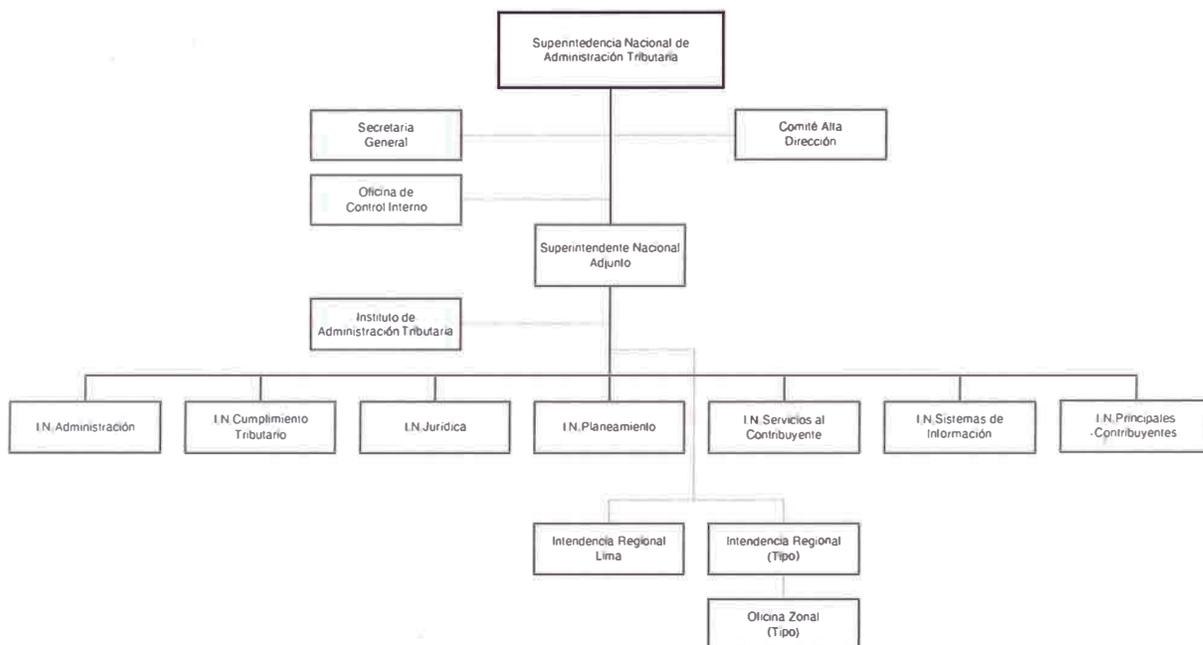


Figura 1. Organigrama de SUNAT

Fuente Manual de organización y Funciones de SUNAT al 22 de Marzo de 2001.

## I.2 ESTRUCTURA DE LA INTENDENCIA NACIONAL DE SISTEMAS DE INFORMACIÓN

La Intendencia Nacional de Sistemas de Información (INSI), está compuesta de cuatro gerencias de línea y dos oficinas de apoyo.

### GERENCIA DE HOMOLOGACIÓN DE SISTEMAS

Tiene por objetivo asegurar que los sistemas y procedimientos elaborados por la INSI cumplan eficientemente los objetivos para los cuales fueron creados.

### GERENCIA DE PRODUCCION

Tiene por objetivo poner a disposición de los usuarios oportunamente la infraestructura informática y de comunicaciones, y los sistemas desarrollados por la Intendencia Nacional de

Sistemas de Información. Asimismo, brindar asistencia técnica para el adecuado uso de los recursos informáticos.

#### **GERENCIA DE SISTEMAS PARA EL CUMPLIMIENTO TRIBUTARIO**

Tiene por objetivo proveer a la institución los sistemas y procedimientos requeridos para apoyar la labor de sus unidades organizacionales, en el ámbito de los sistemas relacionados al cumplimiento tributario.

#### **GERENCIA DE SISTEMAS PARA LOS SERVICIOS AL CONTRIBUYENTE**

Tiene por Objetivo proveer a la institución los sistemas y procedimientos requeridos para apoyar la labor de sus unidades organizacionales, en el ámbito de los sistemas relacionados con el servicio al contribuyente.

#### **OFICINA DE SEGURIDAD INFORMATICA**

Tiene por objetivos, diseñar e implantar los mecanismos de seguridad que impidan accesos no autorizados a la infraestructura informática y de comunicaciones de la institución, así como elaborar los planes de contingencia que aseguren una adecuada disponibilidad de los servicios informáticos implantados.

#### **OFICINA DE TECNOLOGÍA E INGENIERIA DE SOFTWARE**

Tiene por objetivo promover el perfeccionamiento de los servicios informáticos brindados por la Intendencia Nacional de Sistemas de Información, a través de una continua investigación, evaluación, incorporación y normalización de las tecnologías de información en la institución.

El organigrama de la INSI se muestra en la siguiente figura.

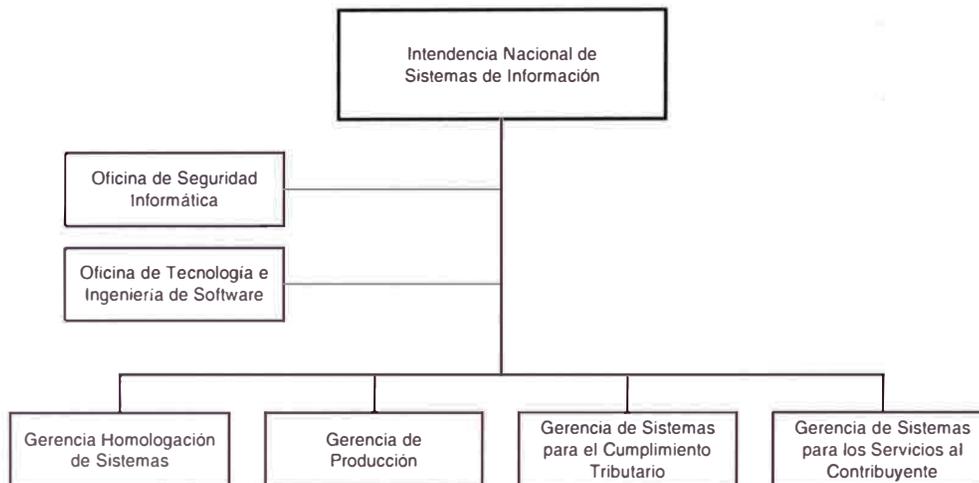


Figura 2. Organigrama de Intendencia Nacional de Sistemas de Información de SUNAT  
Fuente Manual de organización y Funciones de SUNAT al 22 de Marzo de 2001.

### **I.3 DIAGNOSTICO ESTRATÉGICO DE LA GERENCIA DE HOMOLOGACIÓN DE SISTEMAS**

Inicialmente la Intendencia de Sistemas, tenía como objetivo principal ser un proveedor de sistemas de información e infraestructura tecnológica, sin embargo con la reestructuración organizacional realizada en febrero de 2001, se le adiciona la función de proveedor de procedimientos, con lo cual esta Intendencia se hace responsable de todos los elementos que componen un sistema de información: Componente informático, Infraestructura y Procedimientos.

De igual forma la verificación de la calidad de los sistemas a través del proceso de pruebas rudimentario que se tenía establecido, se amplía a un proceso de homologación de sistemas, es decir verificar la calidad de los sistemas informáticos y verificar que los procedimientos establecidos en circulares correspondan al funcionamiento del sistema informático antes de su publicación.

A continuación se desarrolla un análisis interno (fortalezas y debilidades) y externo (oportunidades y amenazas) de la Gerencia de Homologación.

## **FORTALEZAS**

- El personal cuenta con una visión del negocio tributario desde una perspectiva funcional y desde una perspectiva técnica de los sistemas.
- Se tiene acceso a la información relevante de todos los sistemas, desde su concepción hasta su implantación y post evaluación.
- El uso de un método formal (el cual se describe en el presente trabajo), claro y eficiente para realizar el proceso de homologación
- Apoyo de la Alta Dirección, conciente que la calidad de los sistemas son vitales para el logro de los objetivos estratégicos.

## **DEBILIDADES**

- Pocos recursos asignados, la escasez de recursos ha obligado a que el personal de Homologación se especialice en sistemas específicos, generando una dependencia de personas para realizar el proceso de Homologación
- El proceso de homologación se realiza en forma manual.
- No existe un proceso de gestión y seguimiento de los errores, detectados en la etapa de homologación y/o en post-producción
- Generación de cuellos de botella por la acumulación de pases a producción que no pueden ser homologados en paralelo, sino en forma secuencial.
- Todavía no se cuenta con un ambiente de pruebas adecuado para la simulación de los diversos ambientes de producción.

## **OPORTUNIDADES**

- Avances en las tecnologías y herramientas de pruebas, que permitirían asimilar técnicas y herramientas para mejorar la calidad y disminuir los tiempos del proceso de pruebas.
- Recientemente se viene impulsando el establecimiento de una Metodología formal para el proceso de desarrollo de sistema en SUNAT. Esta metodología permitiría planificar y gestionar adecuadamente el proceso de homologación de sistemas.

## **AMENAZAS**

- Sobredimensionamiento de la cartera de proyectos a ser homologados, debido a la escasez de recursos y la creciente demanda de servicios de desarrollo a la INSI.
- Incompatibilidades entre el proceso de desarrollo a establecer en SUNAT y el proceso de homologación establecido.

## **CAPITULO II            MARCO TEORICO**

### **II.1    DEFINICION DEL PROCESO DE PRUEBAS DE SOFTWARE**

La Prueba de software es aquella actividad que pretende evaluar un atributo o capacidad de un programa o sistema con el propósito de encontrar errores que impidan que éste satisfagan correctamente los resultados requeridos, así como que cumpla con las restricciones y condicionantes establecidas formalmente. (Bill Hetzel [3], G. Myers [9])

Para reforzar esta definición, la prueba de un sistema como cualquier proceso de una cadena productiva, debe agregar valor al producto final, esto se logra aumentando su calidad o su confiabilidad, a través de la detección y eliminación de errores. Por lo tanto no se debe probar para demostrar que todo esta bien o todo funciona, sino es conveniente comenzar con la suposición de que un programa contiene errores, los cuales serán encontrados durante el proceso de pruebas.

Es importante que las pruebas deban servir, en buena medida, para aprender de los errores cometidos, de tal forma que una organización logre una retro-alimentación sistemática de las experiencias obtenidas en cada proyecto. Lo cual conduce a mejorar el proceso de desarrollo, llevándolo de un esquema de corrección de errores (enfoque destructivo) a un enfoque de prevención de errores.

### **II.2    EVOLUCION DEL ENFOQUE DE PRUEBAS DE SOFTWARE**

Una concisa descripción de la evolución de la prueba de software es mostrada por Gelperin y Hetzel [8], en su publicación “The Growth of Software Testing – La Evolución de la Prueba de Software”, en 1988. Ellos identificaron los siguientes periodos y/o enfoques de pruebas de software que pueden adoptar las empresas de desarrollo de software.

Orientado al Código

El proceso de pruebas es visto como una ayuda para remover fallas, es realizado por los mismos programadores luego de escribir los programas. El criterio para seleccionar los casos de

	prueba es completamente ad hoc y basado en la experiencia del programador.
Orientado a la Demostración	Las pruebas tienen por objetivo demostrar que el sistema satisface las especificaciones, es decir demostrar que funciona.
Orientado a la Destrucción	La prueba se ve como una actividad para descubrir fallas (efecto destructivo). El criterio para la selección de los casos de la prueba es revelar las faltas particulares que están presentes en el sistema.
Orientado a la Evaluación	La prueba se incorpora al Ciclo de Vida de Software, como una fase de evaluación al final de cada fase del proceso de desarrollo. Este enfoque empezó con la publicación en 1983, acerca de cuanto más temprano se descubre un error menor es su costo de corrección. Para cada fase del ciclo de vida hay requisitos y productos los cuales tienen que ser evaluados.
Orientado a la Prevención	El proceso de pruebas está bien definido e incluye actividades de revisión. La filosofía es prevenir los errores en cada fase del modelo de ciclo de vida realizando pruebas y evaluando los progresos de la fase. El criterio se orienta ahora a encontrar lugares donde podrían presentarse los errores.

## II.3 TECNICAS DE PRUEBAS

### II.3.1 REVISIONES E INSPECCIONES

Son técnicas de control de calidad orientada a verificar que el producto de software a construir satisfaga las necesidades del usuario y/o las especificaciones. El propósito de esta técnica es incrementar la eficiencia del proceso de desarrollo y proveer de un método para medir la calidad de cada producto, reduciendo los retrocesos o repetición del trabajo. Se aplican, principalmente, a los productos de la fase de requerimientos, diseño, programación y plan de pruebas.

Dada la naturaleza intuitiva de estas técnicas, es conveniente mencionar que las inspecciones constituyen uno de los métodos más efectivos en el proceso de prueba de software. La mayoría de las empresas que necesitan producir software de alta calidad lo consideran como una parte fundamental de este proceso, por ejemplo la NASA, IBM, MICROSOFT, etc. (AMERICA XXI [2])

### II.3.2 PRUEBAS DE CAJA NEGRA

Este enfoque, llamado también Prueba Funcional o Prueba de Comportamiento está orientado a la construcción de casos de prueba a partir de las especificaciones y requerimientos del sistema. El software es visto como una caja negra que ante ciertas entradas debe producir determinados resultados. El código y la arquitectura del sistema son desconocidas, lo que se conoce es que ante entradas válidas se tienen respuestas esperadas.

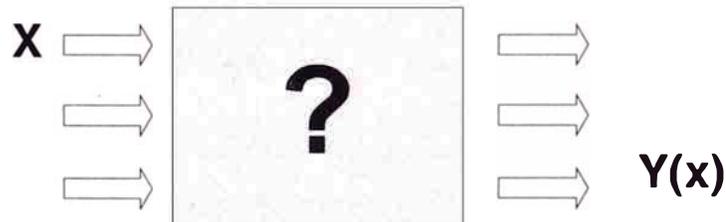


Figura 3: Prueba de Caja Negra

La prueba de caja negra no es una alternativa a las técnicas de caja blanca, mas bien se trata de un enfoque complementario que intenta descubrir errores de las siguientes categorías: (1) funciones incorrectas o ausentes, (2) errores en interfaces, (3) errores en estructuras de datos o accesos a base de datos, y (4) errores de rendimiento.

### CLASES DE EQUIVALENCIA

Es un método para la generación de casos de pruebas de caja negra. Divide el dominio de entrada en clases de datos, llamadas clases de equivalencia, representan un conjunto de estados válidos o no validos para condiciones de entrada.

Cubren un conjunto extenso de otros casos de prueba posibles. Es decir, si un caso de prueba perteneciente a una clase de equivalencia detecta un error, se espera que cualquier otro caso de

prueba perteneciente a la misma clase detecte el mismo error. Inversamente, si un caso de prueba no detecta un error, no se espera que otros casos de prueba de la misma clase puedan detectarlo.

A continuación se muestran unos ejemplos de identificación de 5 clases de equivalencia, para 2 reglas de validación.

Reglas de negocio o Validaciones a realizar	Clases de Equivalencia	
	Válida	Inválida
La numeración de un campo puede ser de 1 a 999	(1) $1 \leq \text{campo} \leq 999$	(2) $\text{campo} < 1$ (3) $\text{campo} > 999$
El primer carácter de una clave debe ser una letra	(4) Es una letra	(5) No es una letra

## ANÁLISIS DE VALORES LÍMITES

El análisis de valores límites es una técnica complementaria a las clases de equivalencia. La experiencia muestra que los casos de prueba que ejercitan situaciones límites producen mejores resultados que aquellas que no lo hacen. Las condiciones límites son las situaciones que se hallan directamente arriba o debajo de los márgenes de las clases de equivalencia.

### II.3.3 PRUEBAS DE CAJA BLANCA

Este enfoque está orientado a la construcción de casos de pruebas a partir del código del programa. Los casos que se construyen están destinados a probar el funcionamiento de los ciclos iterativos, variables, rutinas, caminos, etc. En esta categoría se ubican las técnicas

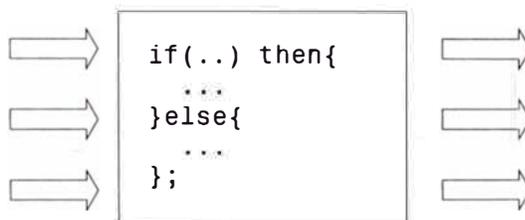


Figura 4: Prueba de Caja Blanca

Mediante los métodos de prueba de caja blanca se puede obtener casos de prueba que (1) garanticen que se ejerciten por lo menos una vez todos los caminos independientes de cada módulo; (2) ejerciten todas las decisiones lógicas en sus vertientes verdadera y falsa; (3) ejecuten todos los bucles en sus límites y con sus límites operacionales, y (4) ejerciten las estructuras internas de datos para asegurar su validez.

A primera vista pareciera que una prueba de caja blanca muy profunda nos llevaría a tener programas 100% correctos. Lamentablemente, la prueba exhaustiva presenta ciertos problemas logísticos, incluso para programas pequeños el número de caminos lógicos posibles puede ser enorme. Consideremos el flujo de control de un programa pequeño de aproximadamente 100 líneas de código, tal como se muestra en la figura 5, se determinan aproximadamente  $10^{14}$  caminos posibles los cuales tardarían unos 3,170 años en ser probados a un ritmo de un milisegundo por cada caso de prueba. (G.Myers [9])

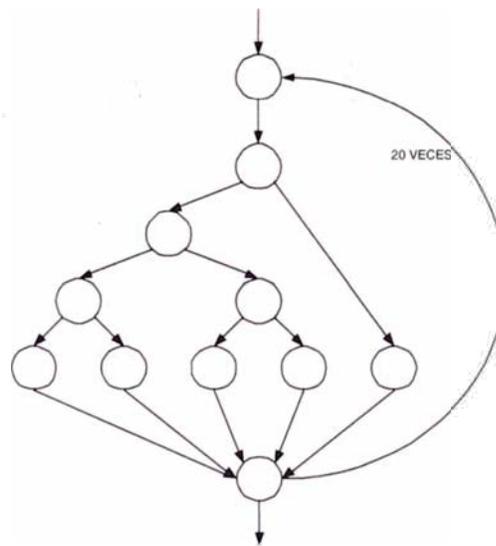


Figura 5: Gráfico del flujo de control de un programa pequeño

La prueba de caja blanca, sin embargo, no se debe desechar como impracticable. Se puede elegir y ejercitar una serie de caminos lógicos importantes. Se pueden combinar los atributos de la prueba de caja blanca así como los de caja negra, para llegar a un método que valide la interfase del software y asegure selectivamente que el funcionamiento interno del software es correcto.

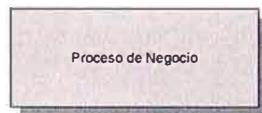
### II.3.4 PRUEBAS DE CAJA GRIS

Esta técnica es una innovación metodológica personal, denominada Pruebas de Caja Gris como consecuencia de la experiencia obtenida en los procesos de homologación realizados en SUNAT. Dada la complejidad del proceso de pruebas y la casuística de los sistemas de gestión, se combinaron las técnicas de Caja Blanca y de Caja Negra para dar origen a esta técnica de pruebas.

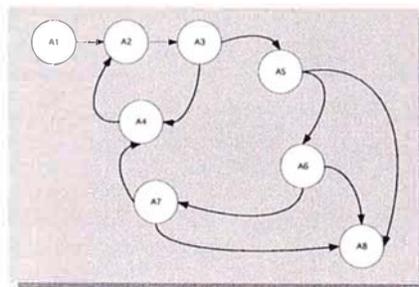
La técnica de Caja Gris se basa en describir el proceso de negocio relacionado al sistema mediante las técnicas de Caja Blanca (grafos), para determinar los caminos a ser probados. Una vez determinados los caminos a probar, se inicia la prueba de cada nodo o actividad usando la técnica de Caja Negra. Dependiendo del nivel de abstracción empleado, podría descomponerse este proceso en un nuevo grafo hasta llegar al nivel en que se deba realizar la prueba de Caja Negra.

A continuación se describe el proceso a realizar al usar esta técnica.

**Paso 01.** Determinar el Dominio del Problema o en su defecto el Proceso de Negocio a ser soportado por el sistema



**Paso 02.** Determinar las actividades o eventos del proceso de negocio, utilizando para ello las técnicas de grafos (Caja Blanca).



**Paso 03.** Se determina los ciclos a ser probados y que darán como resultados los casos de pruebas, para esto se utiliza la técnica de caja blanca. A continuación, a manera de ejemplo se muestra algunos ciclos determinados que deben ser verificados en el proceso de pruebas.

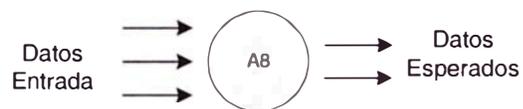
Ciclo 01: 1-2-3-5-6-8

Ciclo 02: 1-2-3-4

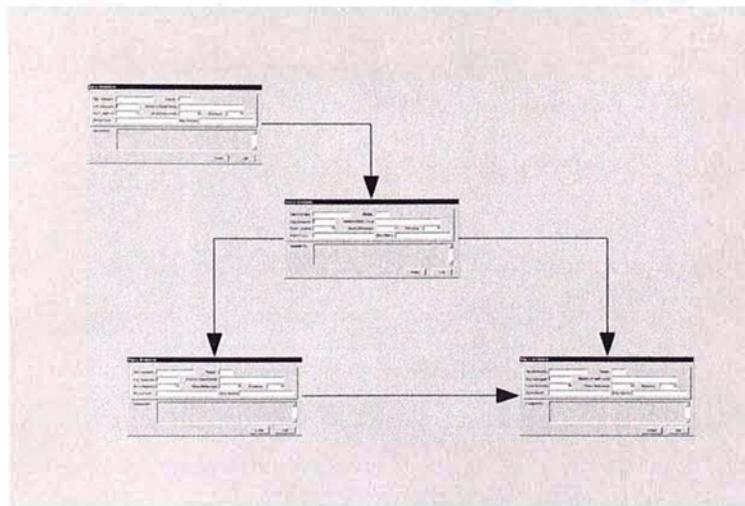
Ciclo 03: 1-2-3-5-6-7-4-2

Ciclo 04: 1-2-3-5-6-7-8

**Paso 04.** Para cada ciclo determinado en el paso anterior, se realizan los casos de pruebas determinando los datos que estimularán cada nodo o actividad y estableciéndose, también, los datos esperados. En este paso se utiliza las técnicas de caja negra.



El nivel de detalle a que se debe llegar en el paso 04 debe ser de tal forma que la actividad unitaria a ser probada sea un módulo, si no es así, el proceso debe ser recursivo, es decir descomponer nuevamente dicha actividad en actividades o eventos de mayor detalle, de tal forma que cada nodo sea una interfase (pantalla) del sistema para poder, luego aplicar, las técnicas de caja negra, tal como se grafica en la siguiente figura.



A manera de ejemplo, se muestra una actividad a ser probada, luego de haber determinado el caso de prueba.

**Nuevo Informante**

Tipo Informante:  Período:

RUC Informante:  Nombre o Razón Social:

Monto Imputado: .00 Monto Determinado: .00 Diferencia: .00

Motivo Cruce:  Otros Motivos:

Lineamientos:

Para determinar los casos de prueba propios de esta actividad del proceso general de negocio, se utiliza las técnicas de caja negra como son las Clases de Equivalencia, ya descritas anteriormente.

#### CLASE VALIDAS

1	Tipo Informante	Controlado por la Aplicación
2	Periodo	>= 1999
3	Ruc Informante	Cumpla Modulo 11 AND Este registrado en el Padrón de Informantes AND Tipo de Informante en el Padrón sea de acuerdo al Tipo de Informante Seleccionado
4	Monto Imputado	Monto Imputado debe ser > 8400
5	Monto Determinado	<> Monto Imputado
6	Diferencia	Valor Calculado: Monto Determinado – Monto Imputado
7	Motivo del Cruce	Valor Calculado: En función al Tipo de Informante. Controlado por la Aplicación

#### CLASE INVALIDAS

8	Tipo Informante	Controlado por la Aplicación
9	Periodo	< 1999
10	Ruc Informante	No Cumpla Modulo 11 OR No este registrado en el Padrón de Informantes OR

		Tipo de Informante en el Padrón no sea de acuerdo al Tipo de Informante Seleccionado
11	Monto Imputado	Tipo de Informante DAOT AND $\leq 8400$
12	Monto Determinado	= Monto Imputado
13	Diferencia	Valor Calculado: Monto Determinado – Monto Imputado
14	Motivo del Cruce	Valor Calculado: En función al Tipo de Informante. Controlado por la Aplicación

En términos generales la técnica de Caja Gris, nos permite enfrentar las pruebas de sistemas de gestión complejos, tomando como punto de partida la visión del Negocio el cual es válido para cualquier metodología de desarrollo.

## **CAPITULO III PROCESO DE TOMA DE DECISIONES**

### **III.1 PLANTEAMIENTO DEL PROBLEMA**

Para una Administración Tributaria disponer de sistemas de información de alta calidad constituye en elemento esencial para cumplir con su misión. Ante la falta de un proceso formal de pruebas, los sistemas se ponían en producción sin una adecuada verificación de su calidad, introduciendo un alto número de fallas en los sistemas, manifestándose a través de ciertos síntomas, que se detallan a continuación:

#### **Síntomas Funcionales**

- Los sistemas puestos en producción no satisfacen los requerimientos iniciales de los usuarios.
- Algunas funcionalidades de los sistemas son contradictorias con los procedimientos establecidos y/o circulares.
- La oportunidad en que se publican las circulares o procedimientos internos que describen cambios o nuevas funcionalidades de los sistemas van desfasados con respecto a la puesta en producción de los sistemas, generando confusión en los usuarios internos.

#### **Síntomas Técnicos**

- Los sistemas se desarrollan sin el cumplimiento de los cronogramas y recursos previamente establecidos.
- Los sistemas implantados presentan errores en su funcionamiento.
- La cantidad de quejas de los usuarios sobre el uso de los sistemas cada año se incrementa.

#### **Causas Posibles**

Luego de analizar las características de los sistemas actuales de la Administración Tributaria, se puede afirmar que las causas son determinadas por:

- No existe un proceso de desarrollo de software que garantice la calidad de los sistemas desarrollados.
- El proceso de Prueba es un factor fundamental para verificar la calidad de software; sin embargo, no es realizada con planes y técnicas formales, siendo la primera tarea en sacrificarse cuando los proyectos están retrasados.
- No existe una adecuada comunicación y coordinación entre las áreas involucradas en el desarrollo de las aplicaciones informáticas. Ocasionando diferencias entre lo requerido, lo establecido en procedimiento o circulares y lo que realmente realiza el sistema.
- No existe una comunicación fluida y permanente con los usuarios, lo cual impide obtener retroalimentación acerca de su grado de satisfacción con los sistemas implantados y conocer sus necesidades de información.
- Los usuarios no conocen totalmente la funcionalidad de los sistemas implantados, debido a la falta de documentación actualizada y a un entrenamiento inadecuado.

### **III.2 CUANTIFICACION DEL PROBLEMA**

Un indicador para medir la mala calidad de los sistemas es la cuantificación de los defectos encontrados en los sistemas en producción. Este indicador nos permitirá medir la cantidad de fallas reportadas en los sistemas de SUNAT puestos en producción.

Dichas cantidades han sido tomadas del área de Atención a Usuarios, encargadas de interactuar con los usuarios ante problemas en los sistemas. El periodo a evaluar comprende desde enero de 1999 a enero de 2001, en el cual se puede apreciar la tasa tan creciente de reportes mensuales de fallas.



Figura 6. Evolución de fallas reportadas en los sistemas en producción.  
Fuente: Sistema de atención a Usuarios. Periodo: Enero 1999 a Enero 2001

Como se puede apreciar a fines del año 2000, el promedio mensual de fallas llegó a alcanzar una tasa de 1000 fallas por mes, manteniéndose esta misma tasa para enero del 2001.

### III.3 SOLUCION PLANTEADA

La calidad en el software es algo que todos quieren. Los jefes de proyectos exigen software de alta calidad, los desarrolladores de software saben que tienen que producir un producto de calidad; y los usuarios insisten que el software trabaje en forma consistente de acuerdo a los requerimientos iniciales y que sea fiable.

Bajo esta perspectiva muchas organizaciones forman grupos encargados del aseguramiento de calidad del software, los cuales muchas veces desarrollan diferentes roles y pueden usar diferentes procedimientos. En algunas organizaciones, la prueba del software es responsabilidad de un grupo, en otros casos es responsabilidad del grupo de desarrollo o de un grupo independiente de la organización.

En el caso de la SUNAT, la solución planteada fue crear la Gerencia de Homologación de Sistemas con la función de asegurar que los sistemas y procedimientos elaborados por la INSI cumplan eficientemente los objetivos para los cuales fueron creados, es decir, verificar que los

sistemas funcionen de acuerdo a las definiciones iniciales o requerimientos de usuarios y que los procedimientos sean coherentes con la funcionalidad del sistema.

Dado que el proceso de homologación de sistemas, se debe establecer sobre un proceso de desarrollo formal, se describe brevemente, en el punto 3.4 la Metodología de Desarrollo de SUNAT.

### III.4 METODOLOGIA DE DESARROLLO ACTUAL

Actualmente en SUNAT, el proceso de desarrollo de los proyectos institucionales de sistemas de información se divide en 6 etapas:

- Planeamiento
- Definición
- Construcción
- Homologación
- Implantación
- Post implantación

En la figura 7, se muestra el Ciclo de Vida del proceso de desarrollo.

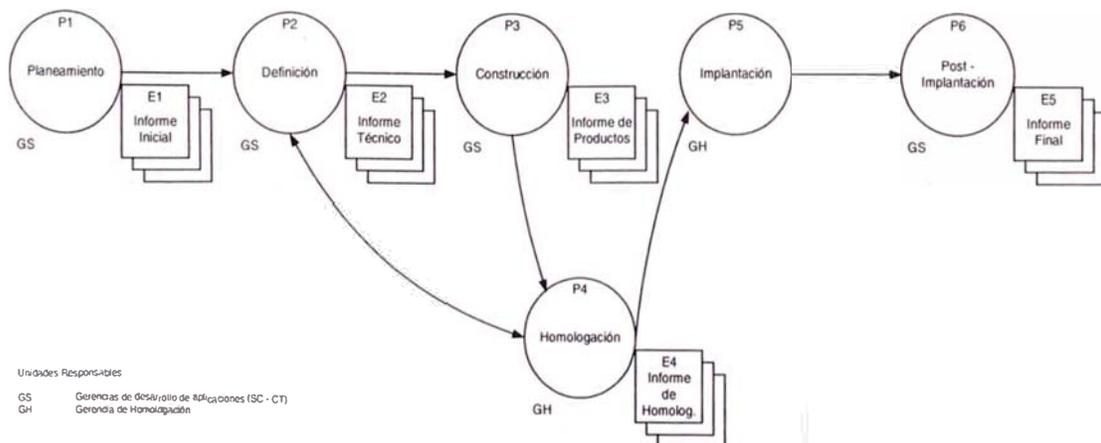


Figura 7: Esquema de la Metodología de Desarrollo

A continuación, se exponen los objetivos de cada una de las etapas que comprende el proceso de desarrollo de los proyectos de sistemas de información.

#### **III.4.1 Planeamiento**

El objetivo de esta etapa es definir un nuevo proceso de negocio acorde con las estrategias institucionales. Previamente, se debe entender el proceso de negocio actual que es materia de estudio, para luego identificar y evaluar – técnica y económicamente – las alternativas de solución que conduzcan a la formulación de un plan global para el desarrollo del nuevo proceso, así como del soporte requerido de sistemas de información. La ejecución de esta etapa permitirá lograr un consenso sobre los aspectos globales del proyecto entre todos los participantes a niveles estratégico – aquellos involucrados con el logro de los objetivos del proceso de negocio – y táctico/operativo – aquellos directamente involucrados en la realización de las actividades y tareas del proceso de negocio.

En resumen, la etapa de Planeamiento define una estrategia de desarrollo que permite llevar a cabo un proyecto coherente con la dirección del negocio.

#### **III.4.2 Definición**

El objetivo de esta etapa es diseñar o modelar el nuevo proceso de negocio – así como el soporte de sistemas de información – en función al análisis detallado de los requerimientos globales del proceso.

Es en esta etapa cuando la idea resultante de la etapa anterior empieza a tomar forma y se va plasmando en modelos formales, cobra especial importancia la participación del nivel táctico/operativo (usuarios).

#### **III.4.3 Construcción**

El objetivo de esta etapa es producir un conjunto de programas y procedimientos computacionales que permitan realizar las tareas definidas en la etapa previa; asimismo, generar los manuales, normas y procedimientos necesarios para la normal ejecución del nuevo proceso de negocio. Culminada esta etapa, se deberá contar con un producto lo suficientemente estable para proseguir con la etapa de Homologación.

#### **III.4.4 Homologación**

El objetivo de esta etapa es realizar un conjunto de pruebas exhaustivas al sistema y documentación desarrollados, con el fin de encontrar defectos y fallas que impidan su cumplimiento - de manera eficaz y eficiente – con los objetivos para los cuales fueron creados.

#### **III.4.5 Implantación**

El objetivo de esta etapa es realizar las actividades destinadas a asegurar la oportuna puesta en funcionamiento del sistema, así como la óptima utilización del mismo por parte de los usuarios.

#### **III.4.6 Post- Implantación**

El objetivo de esta etapa es evaluar la gestión global del proyecto, con el fin de realizar los ajustes requeridos a la versión implantada o plantear las consideraciones a tomar en cuenta para futuras versiones. Asimismo, a través de esta evaluación global, se pretende el aprendizaje para un mejor planeamiento y control de futuros proyectos de sistemas de información.

### III.5 VISION SISTEMICA DEL PROCESO DE HOMOLOGACIÓN

La Homologación de Sistemas abarca la verificación de los procedimientos contenidos tanto en las normas legales, las circulares, así como el componente informático y su manual de usuario.

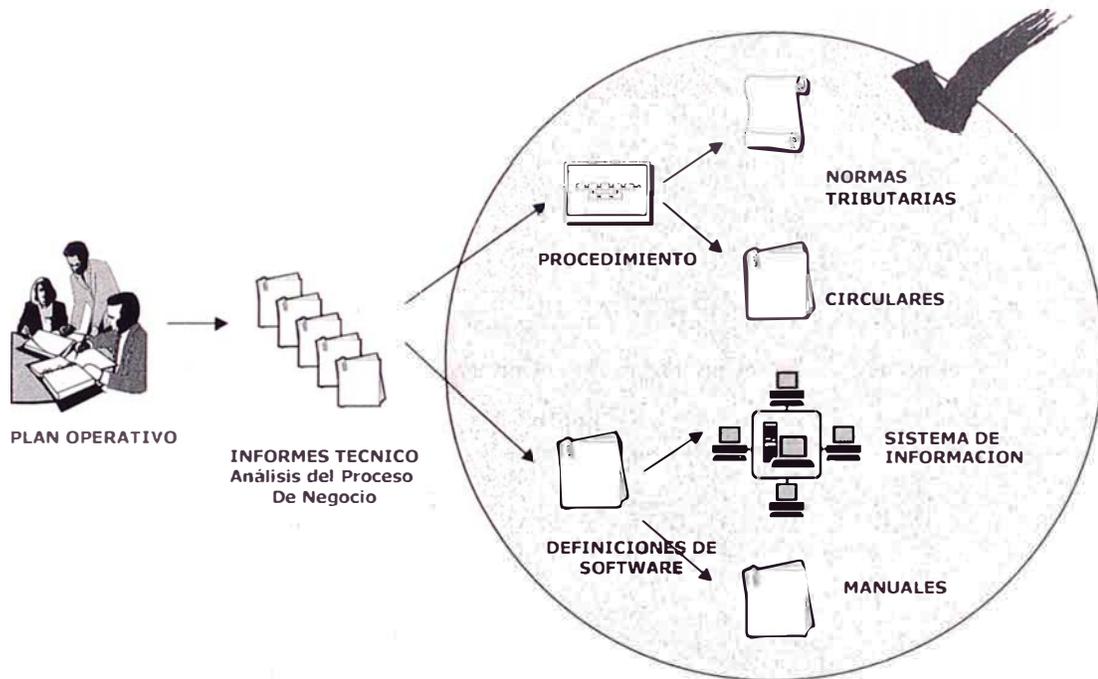


Figura 9: Visión Sistémica del Proceso de Homologación

Como parte del proceso de Homologación del Sistema, las Pruebas del Sistema Informático es el principal elemento de la Homologación e implica realizar un conjunto de pruebas exhaustivas al software desarrollado, con la finalidad de identificar fallas que impidan que éste satisfaga correctamente todos los requerimientos del sistema, así como que cumpla con las restricciones y condicionantes establecidos formalmente en el documento de definiciones de software. Esta prueba puede darse a dos niveles:

- **FUNCIONAL**

Su alcance es la homologación de los procedimientos que soportan al sistema, así como a la documentación legal que lo respalda, ya sea Norma Legal o Circular.

- **TECNICO**

Su alcance es la homologación del sistema informático con su respectiva documentación técnica.

### III.6 PROCESO DE HOMOLOGACIÓN DE SISTEMAS

Tomando como referencia el proceso de desarrollo descrito anteriormente, se describe a continuación los procesos principales de la Homologación de los Sistemas.



Figura 8: Procesos Principales de la Gerencia de Homologación

Desde la perspectiva planteada y tomando como referencia el Manual de Organización de Funciones de SUNAT, el objetivo principal de la Homologación de Sistemas es asegurar que el sistema elaborado por las Gerencias de Desarrollo (Gerencia de Sistemas para el Servicio al Contribuyente y la Gerencia de Sistemas para el Cumplimiento Tributario) cumplan eficientemente los fines para los cuales fueron creados.

Para la homologación de los sistemas se han planteado los siguientes 4 principales actividades:

- Reconocimiento del Sistema
- Planificación del Proceso de Homologación
- Homologación del Sistema
- Evaluación de Resultados

### III.6.1 RECONOCIMIENTO DEL SISTEMA

#### OBJETIVO

El objetivo de esta actividad es tomar conocimiento del sistema desde el inicio, es decir, desde que se elabora el documento de análisis del proceso de negocio; con la finalidad de comenzar a programar la homologación del sistema. Esta tarea se extiende hasta la elaboración de las definiciones del sistema informático y los procedimientos.

#### ACTIVIDADES

En la figura 10 se detalla las actividades que se realizan en el proceso de Reconocimiento del Sistema.

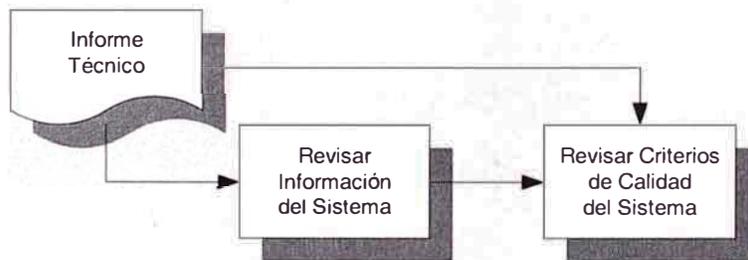


Figura 10: Actividades involucradas en el Reconocimiento del Sistema

#### REVISAR INFORMACIÓN DEL SISTEMA.

En esta actividad se debe revisar la información concerniente a la funcionalidad y diseño del sistema:

- Casos de uso.
- Funcionalidad.
- Modelamiento de datos.
- Diseño del Sistema Informático.
- Prototipo de interfase con el usuario.

#### REVISAR CRITERIOS DE CALIDAD DEL SISTEMA

También se debe verificar que el Informe Técnico contenga la información mínima para realizar las pruebas del sistema informático:

- Requerimientos de configuración de hardware y/o software definidas como aceptables.
- Arquitectura del Sistema.

- Estimación de valores máximos de volumen de la base de datos.
- Límites esperados de capacidad de procesamiento y de almacenamiento.
- Tiempo de respuesta esperado del sistema informático.
- Cantidad máxima de usuarios accediendo concurrentemente al sistema.
- Especificaciones del modelo de fallas y el plan de contingencia o uso de backups.

Esta información servirá para establecer los tipos de pruebas de sistemas que se realizará.

### III.6.2 PLANIFICACION DEL PROCESO DE HOMOLOGACIÓN

#### OBJETIVO

En esta etapa se define el Plan de Homologación que tendrá como objetivo realizar la homologación del sistema. Esta tarea será realizada por el Coordinador del Proceso de Homologación.

#### ACTIVIDADES

Las tareas que se detallan a continuación, definen el proceso de Plan de Homologación. En esta fase se debe elaborar el documento denominado *Plan de Homologación* (Anexo N° 2), el cual será tomado como línea base en el proceso de Homologación.

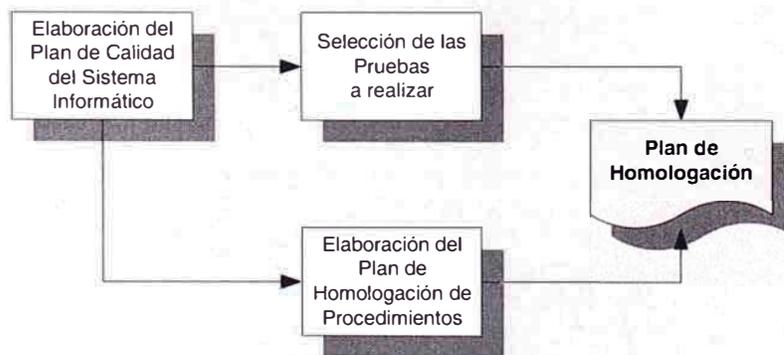


Figura 11: Actividades de la Planificación del Proceso de Homologación

#### 1. ELABORACIÓN DEL PLAN DE CALIDAD DEL SISTEMA INFORMÁTICO

Esta actividad tiene por finalidad:

- Definir los objetivos de la prueba.
- Definir los niveles de calidad esperados.

- Determinar el tipo de pruebas que serán necesarios realizar sobre el sistema informático.
- Determinar los recursos necesarios, tanto de software, hardware, sistemas operativos, base de datos comunicaciones y otros.
- Estimar los Recursos Humanos que se requerirán para:
  - Construir el ambiente de pruebas
  - Generar los casos de pruebas
  - Ejecución de las pruebas
- Establecer las actividades y cronograma de las pruebas del sistema informático.

## **2. SELECCIÓN DE LAS PRUEBAS A REALIZAR**

De acuerdo a las características del sistema a homologar se establecerá los tipos de pruebas a desarrollar, y su estrategia de llevarlas a cabo. Las pruebas se clasifican en:

- Prueba Funcional o Pruebas de Caja Negra
- Pruebas de Caja Blanca
- Pruebas de Sistemas
  - De sentencia
  - De volumen
  - De esfuerzo
  - De facilidad de uso
  - De seguridad
  - De almacenamiento
  - De configuración
  - De compatibilidad / conversión
  - De recuperación
- Pruebas de Aceptación
- Pruebas de Instalación
- Pruebas de Regresión

La descripción de cada tipo de prueba se detalla en el Anexo N° 1.

## **3. ELABORAR PLAN DE HOMOLOGACIÓN DE PROCEDIMIENTOS**

Se establecerá las actividades y el cronograma para la homologación de los procedimientos.

### III.6.3 HOMOLOGACION DEL SISTEMA

#### OBJETIVO

El objetivo es la verificación de estándares de la documentación entregada, homologación de los procedimientos con el Informe Técnico y las Pruebas del Sistema Informático.

#### ACTIVIDADES

Las actividades que se describen a continuación, definen el proceso de homologación:

- Homologación de los procedimientos
- Pruebas del sistema informático

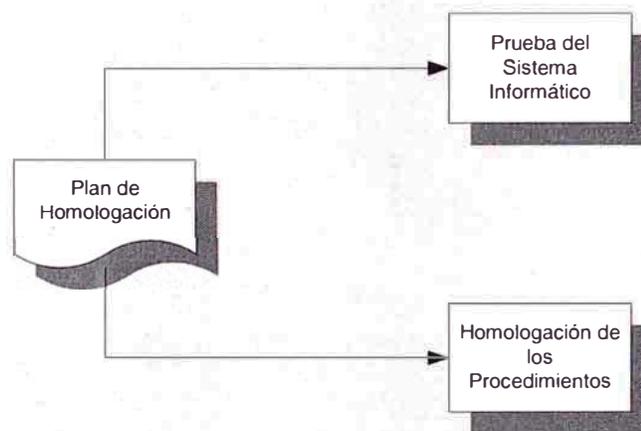


Figura 12: Actividades del proceso de homologación del sistema

#### 1. HOMOLOGACION DE LOS PROCEDIMIENTOS

La Homologación de los procedimientos implica asegurar la consistencia entre una propuesta de Norma Legal o Circular y el Modelo de Negocio al cual el sistema dará soporte.

#### 2. PRUEBAS DEL SISTEMA INFORMÁTICO

El objetivo de esta etapa es realizar un conjunto de pruebas exhaustivas al sistema informático desarrollado con el objeto de encontrar defectos y fallas que impidan que éste satisfaga correctamente todos los requerimientos del sistema, así como que cumpla con las restricciones y condicionantes establecidos formalmente en el documento de definiciones del sistema informático.

Las actividades que se realizan para llevar a cabo las pruebas del software son las siguientes:

- Reconocimiento del Software
- Construcción del Ambiente de Prueba
- Diseñar los Casos de Prueba
- Generación de los Datos de Prueba
- Ejecución de las Pruebas
- Revisión de los Manuales de Usuario

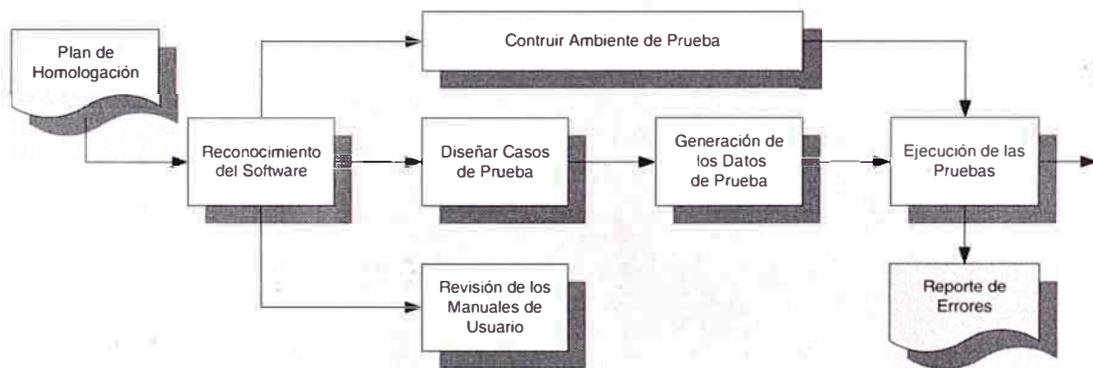


Figura 13: Actividades de la Prueba del Sistema Informático.

### **2.1 Reconocimiento del Software**

El objetivo verificar lo siguiente:

- Que los programas fuentes enviados a prueba estén completos.
- Cuento con el Instructivo del Pase a Producción correspondiente.
- Cuento con documentación para el uso del Software.

### **2.2 Construcción del Ambiente de Prueba**

Esta actividad tiene por objeto analizar, especificar y construir el ambiente de pruebas para la aplicación. El objetivo del ambiente de pruebas es que se pueda simular lo más fielmente posible las condiciones en que correrá la aplicación una vez en su ambiente de producción. Esta actividad es llevada a cabo por el Coordinador del Proceso de Homologación con el apoyo de los Analistas de Homologación y si es necesario se coordinará con el Departamento de Soporte Técnico de la Gerencia de Producción.

### ***2.3 Diseñar los Casos de Prueba***

Esta actividad tiene por objeto diseñar los casos de pruebas que permitan someter al software en prueba a un conjunto de escenarios que den una seguridad razonable de que el sistema informático construido se comportará de acuerdo a lo esperado.

Se establecerá como línea base los Casos de Prueba, de acuerdo a los formatos establecidos. El formato del Caso de Prueba se adjunta en el Anexo N° 3.

### ***2.4 Generación de los Datos de Prueba***

Esta actividad tiene por objeto transformar los casos de pruebas en datos físicos que permitan someter al sistema informático a pruebas concretas con el fin de detectar fallas y defectos.

### ***2.5 Ejecución de las Pruebas***

Esta actividad tiene por objeto realizar las pruebas del software a través de inspecciones o ejecuciones del sistema informático según el tipo de prueba y técnica en particular establecida.

Para la ejecución de la prueba de aceptación se convocará a los usuarios, con la finalidad de generar una participación activa de los usuarios en la determinación de la aceptación del sistema, con lo cual se valida formalmente que el sistema se ajusta a sus necesidades.

Para las pruebas de esfuerzo, volumen, comportamiento y almacenamiento es necesario coordinar con el Departamento de Soporte Técnico y/o el Departamento de Telecomunicaciones con la finalidad de poder contar con sus respectivos aportes para la medición del desempeño del sistema en la red de comunicaciones y servidores.

Para las pruebas de seguridad del sistema informático, se coordinará con la Oficina de Seguridad Informática para contar con el apoyo técnico respectivo.

### **Registro de Errores**

Los errores que se vayan encontrando en el sistema informático, se deben reportar inmediatamente para su corrección y deben realizarse en los formatos ***Reporte de Errores en Pruebas*** (Anexo N° 4).

### ***2.6 Revisión de los Manuales de Usuario***

Esta actividad tiene por objeto realizar la verificación del manual de usuario, respecto de la adecuada descripción del sistema informático a implementarse.

### III.6.4 EVALUACION DE RESULTADOS

#### OBJETIVO

El objetivo de este proceso es analizar los resultados de las pruebas del sistema de información y dependiendo de los resultados obtenidos se formalizará la aprobación del sistema, mediante el Informe de Homologación.

#### ACTIVIDADES

Las actividades que se describen a continuación, definen el proceso de Evaluación de Resultados:

- Evaluación de resultados
- Preparación del Informe de Homologación

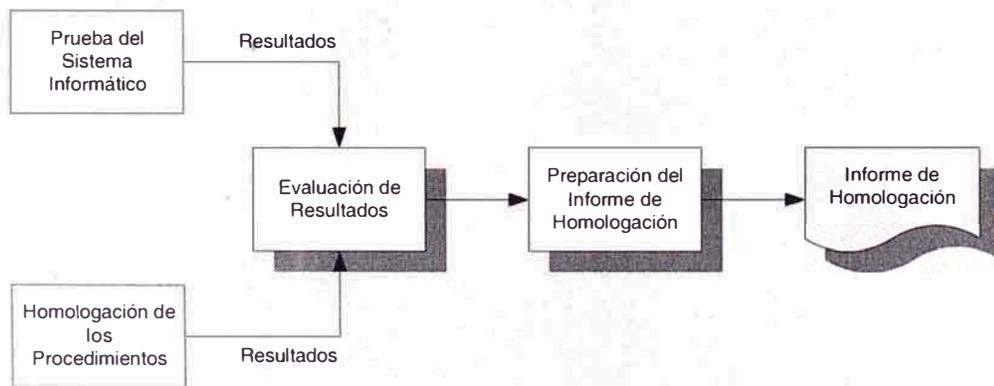


Figura 14: Actividades de la Evaluación de Resultados

#### 1. EVALUACION DE RESULTADOS

Se evalúan los resultados de las pruebas analizando las incidencias recibidas y se comprueba que se han llevado a cabo todos los casos de pruebas establecidos en el plan de pruebas. Una vez comprobado que el sistema cumple con todos los requisitos se registra dicho resultado en el documento de Informe de Homologación, incluyendo la aprobación o rechazo del sistema.

## **2. PREPARACIÓN DEL INFORME DE HOMOLOGACIÓN**

- Consiste en la elaboración física del informe de homologación de acuerdo al estándar establecido (Anexo 5)

## CAPITULO IV EVALUACIÓN DE RESULTADOS

El proceso de Homologación implementado en SUNAT, como elemento vital del proceso de desarrollo ha permitido mejorar la calidad de los sistemas puestos en producción, disminuyendo la cantidad de errores detectados en producción y en forma indirecta la cantidad de solicitudes de atención por parte de los usuarios a la INSI. A continuación se muestra en forma cuantificable los resultados obtenidos.

### EVOLUCION DE LOS ERRORES

Analizando el mismo indicador utilizado para la cuantificación del problema podemos apreciar en la figura N° 15, la evolución de los errores desde el momento de la implementación del proceso de homologación, hecha en Febrero de 2001.

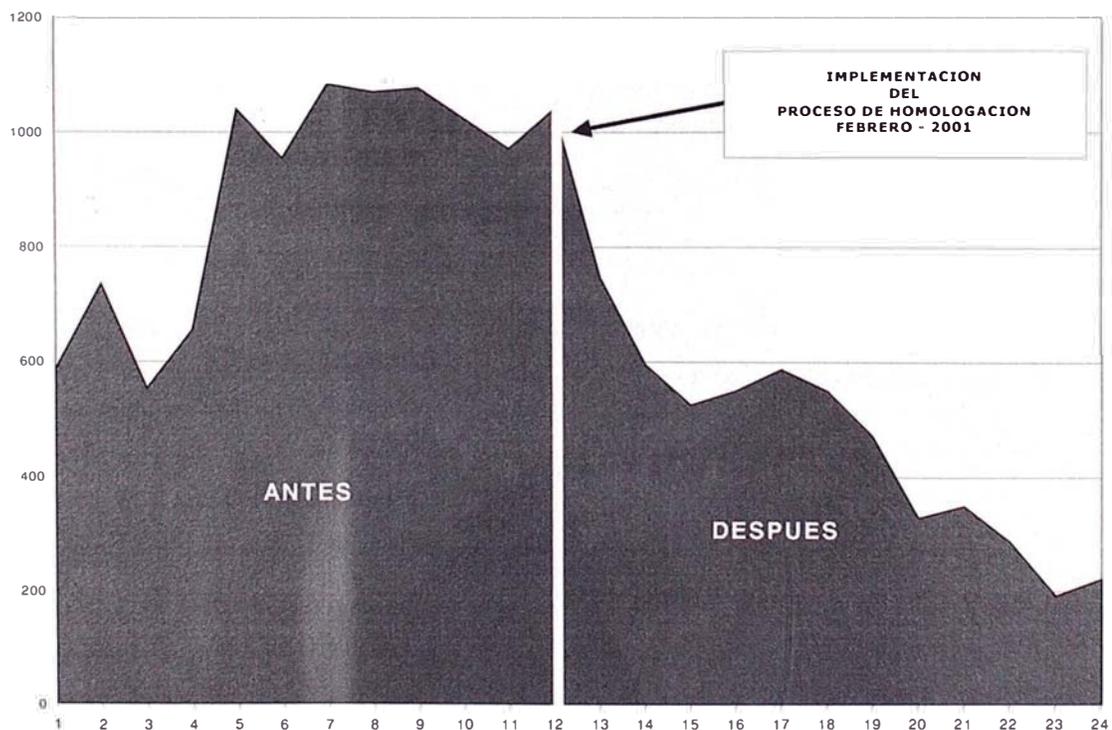


Figura 15. Evolución de fallas reportadas en los sistemas en producción.  
Fuente: Sistema de atención a Usuarios. Periodo: Enero 1999 a Enero 2002

En resumen, podemos afirmar que el proceso de homologación a contribuido directamente en la disminución de los errores de los sistemas pasados a producción, lo cual ha permitido mejorar la satisfacción de los usuarios, mejorar la productividad de los procesos internos y en términos generales mejorar la efectividad de la Intendencia Nacional de Sistemas de Información (INSI) como proveedor de sistemas de calidad a las necesidades de la institución.

### **EVOLUCION DE LAS ATENCIONES A USUARIOS**

Otro indicador interesante para medir la calidad de los servicios que brinda la INSI a la institución es la cantidad de atenciones a usuarios que se reciben, en dicho indicador una gran participación se debe a las fallas de funcionalidad de los sistemas que reportan diariamente los usuarios, dicha participación ha disminuido en 2001 como consecuencia de la disminución de los errores de los sistemas, el siguiente cuadro muestra los resultados estadísticos mencionados, en el periodo 1999 a Enero del 2002 (Fuente: Sistema de Atención a Usuarios).

<b>Año</b>	<b>Atenciones a Usuarios (A)</b>	<b>Fallas de los Sistemas (B)</b>	<b>Participación: ( B/A )</b>
1999	10,926	6,536	59.82 %
2000	14,644	9,628	65.75 %
2001	12,246	6,224	50.82 %
2002* (solo enero)	676	230	34.02 %

Tomando como referencia los datos del cuadro anterior, se grafica la evolución de las atenciones de usuarios reportadas a la INSI y su relación con la cantidad de fallas reportadas (Figura N° 16).

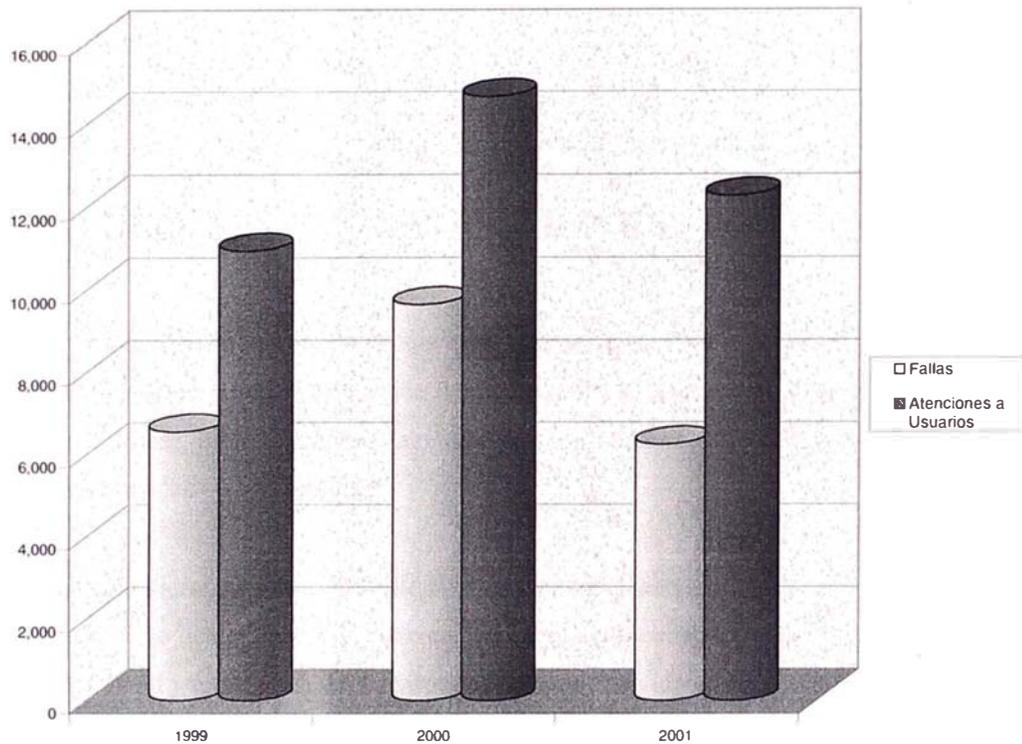


Figura 16. Evolución comparativa de las atenciones a usuarios y fallas por año  
Fuente: Sistema de atención a Usuarios. Periodo: Enero/1999 a Diciembre/2001

En la figura N° 16, se puede apreciar que la cantidad de fallas en los sistemas han disminuido significativamente en el año 2001 ( 6,224 fallas ) con respecto al año 2000 ( 9,628 fallas ), **mejorando la calidad de los sistemas en 35.4 %.**

El efecto de la caída de las fallas en los sistemas en 34.5 %, tiene un efecto directo en la caída de atenciones a usuarios de 14,644 atenciones en el año 2000 a 12,246 en el año 2001, es decir, **una disminución de 16.4 %.**

Con los resultados mostrados podemos afirmar, que el proceso de homologación ha cumplido su principal objetivo: disminuir la tasa de errores en los sistemas puestos en producción, y como consecuencia aumentar la confiabilidad de los sistemas.

## **CAPITULO V            CONCLUSIONES**

Con los resultados obtenidos en la disminución de la tasa de errores en el primer año de implantado el proceso de homologación queda demostrado que la homologación de sistemas es un proceso eficaz para minimizar las fallas en los sistemas y como consecuencia de ello aumentar su calidad y confiabilidad.

Los resultados logrados sólo es posible si el proceso de pruebas se constituye como un proceso formal en el ciclo de desarrollo de sistemas, se cuenta con el apoyo de la Alta Dirección dada la importancia vital y estratégica de los sistemas en la operatividad de SUNAT.

Es importante resaltar, que un factor crítico en la implantación del proceso de homologación de sistemas fue conformar un equipo independiente de personas dedicadas exclusivamente a desarrollar las actividades de homologación de sistemas y contar con el mismo nivel de autoridad que los equipos de desarrollo de sistemas.

La metodología planteada y diseñada para una institución pública como es el caso de SUNAT, tiene muchos elementos importantes que pueden ser reutilizables y adaptables a cualquier realidad o empresa que desee implementar un proceso de pruebas formal como parte del ciclo de vida del desarrollo de sistemas de información.

Es necesario tener una enfoque integral (visión holística) para establecer el proceso de pruebas formal, ya que no se puede descuidar los elementos que acompañan y sustentan la implementación de un sistema, es decir dentro del proceso de pruebas del sistema se han considerado como insumos los componentes técnicos y legales.

A pesar de la necesidad de probar software, existe poca difusión del tema en nuestro medio ya sea a nivel profesional o académico, lo que ha generado que su aplicación en los ámbitos productivos de software sea en forma intuitiva y rudimentaria sin aplicación de técnicas o

métodos como los descritos en el presente trabajo, el cual pretende y espera haber contribuido sobre el tema de Pruebas de Software, y demostrado a través de la implementación del proceso de Homologación en la SUNAT, los éxitos inmediatos que se pueden alcanzar asegurando la calidad de los sistemas en beneficio de la eficiencia organizacional.

## BIBLIOGRAFIA

- [ 1 ] ALBERTO G. ALEXANDER.: “*La mala calidad y su costo*”. Addison-Wesley Iberoamericana, 1994.
- [ 2 ] AMERICA XXI Consultores Informáticos, “*Metodología de desarrollo de proyectos Softcal*”, Versión 3.0, Chile, 1996.
- [ 3 ] BILL HETZEL, WILLIAM HETZEL. “*The Complete Guide to Software Testing*” (*La Guía Completa de la Prueba de Software*). J.Wiley, 1993.
- [ 4 ] BORIS BEIZER.: “*Black Box Testing*” (*Pruebas de Caja Negra*). J.Wiley, 1995.
- [ 5 ] BORIS BEIZER.: “*Software Testing Techniques*” (*Técnicas de Prueba de Software*). Van Nostrand Reinhold, 1990.
- [ 6 ] CARNEGIE MELLON UNIVERSITY.: “*The Capabilty Maturity Model – Guidelines for improving the software process*” (*El Modelo de Maduración de Capacidades – Pautas para mejorar el proceso de software*). Addison-Wesley, 1994.
- [ 7 ] CEM KARNER, JACK FALK, HUNG QUOC.: “*Testing Computer Software*” (*Prueba de Software de computadoras*). J.Wiley, 1999.
- [ 8 ] DAVID GELPERIN, BILL HETZEL.: “*The growth of software testing*” (*La evolución de la prueba de software*). Communications of ACM (Association for Computing Machinery – [www.acm.org](http://www.acm.org)). Junio 1988, volumen 31, N°6, Páginas 687-695.
- [ 9 ] GLENFORD J. MYERS.: “*El arte de probar el software*”. El Ateneo, 1983.
- [ 10 ] HELGA DRUMMOND.: “*La Calidad Total – El movimiento de la calidad*”. Deusto, 2001.
- [ 11 ] JAMES A. SENN.: “*Análisis y Diseño de Sistemas de Información*”. McGraw-Hill, 1992.
- [ 12 ] ROGER S. PRESSMAN.: “*Ingeniería de Software, Un enfoque práctico*”. McGraw-Hill, 1998.
- [ 13 ] STEVE McCONNELL.: “*Desarrollo y Gestión de Proyectos Informático*”. McGraw-Hill, 1997.
- [ 14 ] WILLIAM E. LEWIS.: “*Software Testing and Continuous Quality Improvement (Prueba de Software y Mejoramiento Continuo de la Calidad)*”. Auerbach, 2000.

# ANEXO N° 1

## TIPOS DE PRUEBAS

### 1 PRUEBA DEL SISTEMA

La prueba de sistemas tiene un propósito particular de comparar el sistema o programa con sus objetivos originales (qué debe hacer el programa y con qué calidad debe realizarlo). Dos consecuencias de este propósito son:

- La prueba de sistema no se limita a “sistemas”. Si el producto es un programa, la prueba de sistemas es el proceso de tratar de demostrar cómo el programa no cumple sus objetivos.
- La prueba de sistemas, por definición, es imposible si el proyecto no ha producido un conjunto escrito de objetivos medibles para su producto.

No existe una metodología identificable para probar sistemas, por lo mismo, se requiere un elevado nivel de creatividad, inteligencia y experiencia.

Se presentan a continuación algunas categorías más importantes de casos de prueba. No se pretende que todas las categorías sean aplicables a cada programa, pero cuando se diseñan casos de prueba se debe explorar la posibilidad de ejecutar las categorías más adecuadas al sistema que se requiere probar (por ejemplo, para los aplicativos que operarán a través de internet es necesario realizar, además de otras pruebas de sistemas, la prueba de seguridad del sistema, pruebas de esfuerzo, etc).

Las pruebas de sistemas son las siguientes:

#### 1.1 Pruebas de Sentencia

La forma más obvia de prueba de sistemas es determinar si cada sentencia mencionada en los objetivos, ha sido implementada realmente. El procedimiento es recorrer los objetivos, sentencia por sentencia y cuando la sentencia especifica algo, determinar si el programa satisface a ese ‘algo’. Frecuentemente, puede efectuarse este tipo de prueba sin necesidad de emplear un computador; a veces es suficiente una comparación mental de los objetivos con la documentación.

## **1.2 Prueba de Volumen**

Es utilizado para probar el funcionamiento del sistema en situaciones de carga pesada continua. No siempre será necesario realizarlo.

Como precondiciones para su planificación se necesita que el Informe Técnico incluya los requerimientos no funcionales de volumen (estimación de valores máximos de volumen de la base de datos y de información a procesar).

Al ejecutarlo, se analiza el comportamiento del sistema ante grandes volúmenes de información, evitando que se vuelva inoperable en caso de que esta situación pueda darse en la realidad.

Al realizar esta prueba se trata de verificar la cantidad de almacenamiento y la capacidad de procesamiento, y combinando ambas alternativa a la vez.

## **1.3 Prueba de Esfuerzo**

Es someter al programa a grandes cargas o esfuerzos considerables. No debe confundirse con la prueba de volumen: un esfuerzo grande en un pico de volumen de datos que se presenta en un corto periodo de tiempo. Por ejemplo si se ha diseñado el sistema para soportar 64 usuarios, la prueba consistirá en someter al programa a una presión de ingreso de 64 usuarios al mismo tiempo.

Como precondiciones para su planificación, se necesita que el Informe Técnico contenga los requerimientos de performance (podrían ser los tiempos de respuesta del sistema reemplazado), cantidad máxima de usuarios accediendo concurrentemente al sistema, etc..

Con su aplicación se detectan malas estrategias arquitectónicas o de programación, pudiendo ser corregidas y evitando con anterioridad a la implantación del sistema, problemas que hacen inservible al mismo.

#### **1.4 Pruebas de Facilidad de Uso**

Trata de encontrar problemas en el factor humano o en la facilidad de uso, por lo tanto en esta categoría de prueba se debe verificar lo siguiente:

- Que se encuentre ajustada cada interfase con el usuario a la inteligencia y al nivel educacional del usuario final y a las previsiones del ambiente, sobre él ejercidas.
- Que las salidas del programa puedan ser interpretadas con facilidad por parte del usuario y desprovistas de la jerga informática.
- Que los mensajes de error puedan ser comprendidos por cualquier usuario.
- Que el conjunto de las interfases de usuario tenga una razonable “integridad conceptual”, un fondo de consistencia y uniformidad de sintaxis, convenciones, semántica, formato, estilo y abreviaturas.
- Que el sistema no contenga un número excesivo de opciones, o contenga opciones que son muy poco probables que se utilicen.
- Que el programa sea fácil de usar.

#### **1.5 Pruebas de seguridad**

Esta prueba consiste en elaborar casos de prueba que permitan burlar los controles del sistema de seguridad.

#### **1.6 Pruebas de almacenamiento**

Los programas tienen ocasionalmente objetivos de almacenamiento que establecen, por ejemplo, las cantidades de los almacenamientos principal y secundario usados por el programa y los tamaños de los archivos temporarios o volátiles que se requieren. Se debe preparar casos de prueba para tratar de mostrar que el programa no cumple los objetivos de almacenamiento.

#### **1.7 Pruebas de configuración**

Los programas tales como sistemas operativos, sistemas para manejo de base de datos y programas para la transmisión de mensajes soportan diversas configuraciones de equipo (hardware). Con frecuencia el número de configuraciones posibles es demasiado grande para intentar una prueba de cada una de ellas, pero el programa debe probarse al menos en cada tipo

de dispositivo y con las configuraciones mínima y máxima posibles. Si el programa mismo pudiera configurarse, entonces debe probarse cada configuración posible del programa.

### **1.8 Pruebas de compatibilidad / conversión**

La mayoría de los programas que se desarrollan no son completamente nuevos; con frecuencia son reemplazados de partes diferentes, ya sea de sistemas de procesamiento de datos, o sistemas manuales. Con tales, programas tienen a menudo objetivos específicos con respecto a su compatibilidad y a sus procedimientos de conversión con el sistema existente. Nuevamente, al probar el programa con respecto a estos objetivos el propósito es demostrar que los objetivos de compatibilidad no han sido logrados y que los procedimientos de conversión no funcionan.

### **1.9 Pruebas de Recuperación**

Los programas tales como sistemas operativos, sistemas para manejo de base de datos y para teleprocesamiento tienen frecuentemente objetivos de recuperación, que establece cómo se recuperará el sistema de los posibles errores de programación, de fallas de equipo y de errores de datos. Uno de los objetivos de la prueba es que estas funciones de recuperación no trabajen adecuadamente. Los errores de programación pueden introducirse adrede en un sistema operativo para determinar si éste es capaz de recuperarse de ellos. Las fallas de equipo ( por ejemplo errores de paridad en memoria, errores en dispositivos de entrada/ salida) pueden ser simuladas. También pueden ser creados y/o simulados ciertos errores en los datos ( por ejemplo ruidos en las líneas de comunicaciones, un índice invalido en una base de datos), para analizar la reacción del sistema.

## **2 PRUEBAS DE INSTALACIÓN**

Es utilizado para probar el procedimiento de instalación del software y hardware. Para realizar esta prueba es necesario que el pase a producción cuente con los requerimientos de configuración, instructivos de instalación, programas instaladores, especificaciones que indiquen cuándo una instalación es exitosa (por ejemplo, archivos log que deben dejar).

De esta manera se exigen procesos robustos y sencillos de instalación de las componentes del sistema, evitando que el producto sea de difícil instalación o que sea posible una instalación que afecte su correcto funcionamiento.

### **3 PRUEBAS DE ACEPTACIÓN**

Es el proceso de comparar el programa con sus requerimientos iniciales y con las necesidades actuales del usuario final (para qué se necesita el programa). Este es un tipo de prueba particular pues la realiza generalmente el usuario final del programa y por lo común, no se considera responsabilidad de la organización de desarrollo. En el caso de un programa que su elaboración fue contratada, un cliente sensato deberá realizar una prueba de aceptación, para ver si el programa satisface sus necesidades; la mejor manera de realizar esto es diseñando casos de prueba para mostrar que el programa no satisface el contrato; si estos casos son infructuosos, el programa deberá ser aceptado.

### **4 PRUEBAS DE REGRESIÓN**

Este tipo de pruebas tiene como precondiciones para su ejecución que ya se hayan realizado pruebas de funcionalidad al sistema y éste haya arrojado errores.

Con la prueba de regresión se verifica que hayan sido subsanados los errores detectado en la prueba de funcionalidad. Hay dos tipos de pruebas de regresión:

- **Progresivo**

Cuando las definiciones se modifican sobre la base de un cambio de requerimientos; se debe probar una aplicación modificada contra las definiciones de software modificado; para lo cual se deben generar nuevos casos y datos de prueba.

- **Correctivo**

Las definiciones no cambian por lo que se debe probar el aplicativo con los casos y datos de prueba con que se detectaron los errores.

## ANEXO N° 2

### PLAN DE HOMOLOGACION

#### 1 DATOS GENERALES DEL SISTEMA A HOMOLOGAR

- Nombre del Sistema
- Nombre del Subsistema
- Nombre de cada módulo a probar
- Número de pases a producción y/o procesos batch

#### 2 OBJETIVOS DE LA HOMOLOGACION

Se especifica si la homologación abarca tanto las pruebas del sistema informático como la homologación de los procedimientos.

Respecto de la homologación del sistema informático, se especifica para qué se prueba, indicando qué factores de calidad serán evaluados.

Los factores de calidad a ser considerados son:

- Correctitud. Grado en que una aplicación satisface los requerimientos del usuario
- Confiabilidad. Grado de certeza de que la aplicación funcione de acuerdo a las definiciones
- Facilidad de uso. Grado de dificultad para aprender y utilizar la aplicación
- Rendimiento. Tiempo de respuesta de la aplicación, en función a objetivos medibles inicialmente establecidos
- Seguridad. Grado de seguridad para controlar los accesos según los distintos perfiles de usuario

#### 3 PRODUCTOS A HOMOLOGAR

Se indica qué productos se procederá a homologar:

- **Sistema Informático**
  - Software
  - Manuales de Usuarios
- **Procedimientos**
  - Circulares
  - Normas Legales

## **4 DETALLES DE LA HOMOLOGACIÓN**

### **4.1 Responsable de la Homologación**

### **4.2 Requerimiento de Recursos para la Homologación**

- Requerimiento de Personal, tanto de la Gerencia de Homologación como la participación de los usuarios.
- Requerimiento de infraestructura informática.

### **4.3 Aspectos a Homologar**

#### **4.3.1 Homologación del Sistema Informático**

##### **- Tipo de Pruebas a Realizarse**

Dependiendo de las características del sistema a homologar se especifican los tipos de pruebas a realizar.

##### **- Procesos a probar**

Se especifica qué procesos dentro de la aplicación van a ser probados. Para ello, se debe tomar en cuenta que todo proceso recibe entradas, realiza una determinada operación y produce salidas.

Las entradas pueden tener 3 tipos fuentes:

- Datos ingresados por pantalla.
- Datos leídos de una tabla.
- Datos leídos desde un archivo plano.

Asimismo, las salidas pueden tomar cuatro formas:

- Resultados mostrados en pantalla.
- Resultados mostrados en un reporte.
- Actualización en una o varias tablas (inserción, eliminación o modificación).
- Generación de un archivo plano

##### **- Otros detalles de la prueba**

Para cada proceso identificado en el punto anterior, se debe indicar el detalle de la prueba del proceso (incluyendo Casos y Datos de Pruebas)

#### **4.3.2 Homologación de Procedimientos**

- Homologación de Estándares de Documentos.
- Homologación de Procedimientos con el Informe Técnico.

#### **4.4 Criterios de Término de la Homologación**

Se debe especificar cuando terminarán las pruebas, según los siguientes criterios:

- Cuando todos los casos construidos hayan pasado sin problemas
- Cuando se termine el plazo asignado y es impostergable.

### **5 CRONOGRAMA DE TRABAJO**

Se anexa el cronograma de trabajo que incluirá las actividades de la fase de homologación del sistema.



## ANEXO N° 4

<b>REPORTE DE ERRORES EN PRUEBAS</b>
--------------------------------------

	<b>Día</b>	<b>Mes</b>	<b>Año</b>		<b>N°</b>	
<b>Fecha:</b>						
<b>Hora:</b>					<b>Cod</b>	<b>GHS/04</b>

**INFORMACIÓN GENERAL.**

SISTEMA :	
SUB-SISTEMA :	
VERSIÓN :	
MODULO :	
NÚMERO DE PASE A PRODUCCIÓN :	
REPORTADO POR :	
REPORTADO A :	
FECHA :	

**DESCRIPCIÓN DEL ERROR**

--

**DIAGNOSTICO DEL ERROR**

Origen de Error	Definiciones <input type="checkbox"/>	Aplicación <input type="checkbox"/>	Documentación <input type="checkbox"/>
Tipo de Error	Leve <input type="checkbox"/>	Grave <input type="checkbox"/>	Muy Grave <input type="checkbox"/>
Acción Tomada	Continuación de la Prueba <input type="checkbox"/>		Suspensión de la Prueba <input type="checkbox"/>

**RESULTADO**

--

	<b>RESPONSABLE DE LA PRUEBA</b>	<b>ANALISTA ENCARGADO DE LA PROGRAMACIÓN</b>
Apellidos y Nombres		
Firma		

## **ANEXO N° 5**

### **INFORME DE HOMOLOGACION**

#### **1 DATOS GENERALES DEL SISTEMA A HOMOLOGAR**

- Nombre del Sistema
- Nombre del Subsistema
- Nombre de cada módulo a probar
- Número de pases a producción y/o procesos batch

#### **2 DATOS GENERALES DE LA HOMOLOGACION**

- Antecedentes
- Responsable de la homologación
- Presupuesto planeado ( Recursos y Horas hombre )
- Presupuesto real
- Motivos del atraso o adelanto

#### **3 PRODUCTOS HOMOLOGADOS**

- Sistema Informático
  - Software
  - Manuales de Usuarios
- Procedimientos
  - Circulares
  - Normas Legales

#### **4 DATOS ESPECIFICOS DE LA HOMOLOGACION POR PRODUCTO**

##### **4.1 Prueba del Sistema Informático**

4.1.1 Tipos de pruebas que se realizaron

4.1.2 Datos específicos de la prueba

- Cantidad de casos de pruebas diseñados
- Cantidad de errores detectados clasificados por proceso y tipo

##### **4.2 Homologación de Procedimientos.**

Informar sobre los aspectos que se homologaron

- Homologación de los estándares de documentos
- Homologación de Procedimientos con el Informe Técnico

## **5 OBSERVACIONES**

### **5.1 Observaciones al sistema informático**

- Consignar los errores atribuibles al sistema informático o a definiciones que no fueron corregidos durante las pruebas.
- Consignar las observaciones al Manual de Usuario.
- Adicionar cualquier información relevante del proceso de pruebas.

### **5.2 Observaciones a los procedimientos**

- Consignar las observaciones de la verificación de la consistencia entre lo indicado en el Informe Técnico y lo consignado en los procedimientos contenidos en las circulares y en las normas legales.

## **6 SUGERENCIAS Y RECOMENDACIONES**

- Analizar las restricciones que se tuvieron durante el proceso de pruebas.
- Adicionar recomendaciones a tener en cuenta para los procesos futuros de prueba.