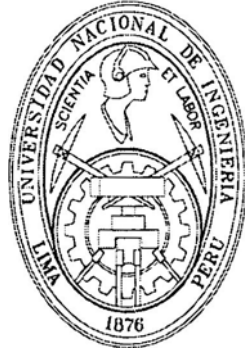


# **Universidad Nacional de Ingeniería**

Facultad de Ingeniería Industrial y de Sistemas



## **IMPLEMENTACIÓN DE LOS SISTEMAS DE ASEGURAMIENTO DE CALIDAD PARA LA FASE DE PRUEBAS EN EL ÁREA DE DESARROLLO DE SISTEMAS DE UNA ENTIDAD BANCARIA**

**Informe de Suficiencia**

**Para Optar por el Título Profesional de:  
INGENIERO DE SISTEMAS**

**MIRIAN LILIANA SAUSA HUAMÁN**

**Lima - Perú  
2002**

*Quiero dedicar este trabajo*

***A Dios***

*Por darme la sabiduría y el tiempo  
para lograr esta meta y estar siempre a mi lado.*

***A mi padre y hermanos***

*Por darme siempre su apoyo incondicional,  
su amor y confianza a lo largo de toda mi vida.*

# INDICE

<b>RESUMEN EJECUTIVO .....</b>	<b>1</b>
<b>INTRODUCCION.....</b>	<b>3</b>
<b>CAPITULO I : ANTECEDENTES .....</b>	<b>5</b>
1.1. EL BANCO .....	5
1.1.1. <i>VISIÓN DEL BANCO</i> .....	6
1.1.2. <i>MISIÓN DEL BANCO</i> .....	7
1.2. DIAGNÓSTICO ESTRATÉGICO .....	7
1.2.1 <i>DIVISIÓN DE SISTEMAS</i> .....	7
1.2.2 <i>ANÁLISIS FODA DE LA DIVISIÓN DE SISTEMAS</i> .....	8
1.3. DIAGNÓSTICO FUNCIONAL.....	9
1.3.1 <i>PRODUCTOS</i> .....	9
1.3.2 <i>CLIENTES</i> .....	9
1.3.3 <i>PROCESOS</i> .....	9
1.3.4 <i>ORGANIZACIÓN</i> .....	10
1.3.4.1 Organización del Banco .....	10
1.3.4.2 Organización de la División de Sistemas .....	10
<b>CAPITULO II : MARCO TEÓRICO.....</b>	<b>11</b>
2.1. ASEGURAMIENTO DE LA CALIDAD EN LA ETAPA DE PRUEBAS .....	11
2.2. CONCEPTOS DE PRUEBAS .....	13
2.2.1. <i>FACTORES QUE DETERMINAN CANTIDAD DE PRUEBAS</i> .....	13
2.2.2 <i>NIVELES DE PRUEBA</i> .....	14
2.2.2.1. <i>PRUEBAS DE VERIFICACIÓN</i> .....	15
2.2.2.2. <i>PRUEBAS DE VALIDACIÓN</i> .....	17
2.2.3. <i>CATEGORÍAS DE PRUEBAS</i> .....	19
<b>CAPITULO III : PROCESO DE TOMA DE DECISIONES .....</b>	<b>21</b>
3.1. PLANTEAMIENTO DEL PROBLEMA .....	21
3.2. ALTERNATIVAS DE SOLUCIÓN .....	23
3.3. METODOLOGÍA DE SOLUCIÓN.....	24

3.4. TOMA DE DECISIONES.....	25
3.5. ESTRATEGIAS ADOPTADAS .....	28
3.5.1 REQUERIMIENTOS .....	28
3.5.2 MISIÓN DE LA DIVISIÓN DE SISTEMAS.....	29
3.5.3 ESTABLECIMIENTO DE UNA POLITICA DE CALIDAD .....	29
3.5.4 PROCESO DE DESARROLLO DE SOFTWARE.....	30
3.5.4.1 Políticas y Estrategias.....	30
3.5.4.2 Conceptos Pilares en el desarrollo de Software.....	31
3.5.5. IMPLEMENTACIÓN DE LA METODOLOGÍA DE PRUEBAS.....	33
3.5.5.1 Estrategias para la implementación.....	33
3.5.5.2 Descripción del proceso de Pruebas.....	35
3.5.5.3 Ambientes de Prueba.....	36
3.5.5.4 Estrategia de pruebas.....	38
3.5.5.5 Herramientas .....	39
3.5.5.6 Estándares .....	40
3.5.5.7 Preparación y Ejecución de la pruebas.....	40
3.5.5.8 Registros.....	41
3.5.5.9 Documentación .....	41
3.5.5.10 Estadísticas.....	42
3.5.6. MEJORAS EN EL PROCESO DE PRUEBAS.....	44
<b>CAPITULO IV : EVALUACIÓN DE RESULTADOS .....</b>	<b>48</b>
<b>CAPITULO V : CONCLUSIONES Y RECOMENDACIONES.....</b>	<b>51</b>
<b>BIBLIOGRAFIA.....</b>	<b>53</b>
<b>GLOSARIO DE TERMINOS.....</b>	<b>54</b>
<b>ANEXOS.....</b>	<b>55</b>

## **DESCRIPTORES TEMÁTICOS**

1. Calidad
2. Aseguramiento de la Calidad
3. Pruebas de Software
4. Metodología de Pruebas de Software
5. Metodología de Pruebas en Banco
6. Desarrollo de Software
7. Metodología de Desarrollo de Software

## **RESUMEN EJECUTIVO**

El banco es una institución financiera que realiza todas las actividades permitidas a la banca múltiple: banca personal, banca corporativa, banca empresarial y banca de inversión. Cuenta con los servicios virtuales y se encuentra desarrollando diversos negocios, repotenciando y renovando sus estrategias comerciales en función a los requerimientos de los clientes.

El área de sistemas brinda el soporte tecnológico de los productos y servicios del banco, siendo la tecnología un elemento clave para la consecución de las metas y objetivos de liderazgo.

Hace algunos años en el área de sistemas no se contaba con una metodología formal de desarrollo de software, ni una metodología de pruebas que permitiera desarrollar productos de alta calidad. No existía una cultura de pruebas en la organización y no existían los ambientes de pruebas adecuados.

A través de un proceso de reingeniería y con la decisión de reemplazar el software bancario, se implementó una metodología de pruebas que incluyó diversas tareas, tales como la creación de los ambientes de prueba, implementación de estándares y metodología de pruebas, y en general la creación de toda una cultura de pruebas dentro del banco.

Es así, que en el banco usando la metodología de desarrollo de software que incluye la metodología de pruebas se han logrado tener experiencias exitosas de grandes proyectos. Con el compromiso y cultura de pruebas de todo el equipo de los proyectos se han obtenido reducciones importantes en los errores y reclamos y disminución en las tareas de mantenimiento y soporte, manteniendo así productos y servicios de calidad y competitivos dentro del mercado.

## INTRODUCCIÓN

La tecnología es hoy fundamental en un banco y comprende el soporte de los productos y servicios que el banco ofrece a sus clientes.

La metodología de desarrollo de software es una guía del modo en que deben interactuar las distintas áreas para el uso eficiente de los recursos tecnológicos, que permitan lograr el desarrollo de sistemas de información adecuados y que sean el soporte de los productos y servicios bancarios.

El aplicar un sistema de aseguramiento de la calidad a los procesos de desarrollo de software y la Metodología de pruebas, son elementos críticos especialmente en ambientes en los que como el financiero un error puede traer graves consecuencias para la organización.

El objetivo de este trabajo es presentar la Metodología de pruebas implementada y usada en los proyectos de software del banco que aseguran la correcta certificación de los componentes informáticos y no informáticos del banco.

La experiencia presentada forma parte del proceso de Transformación de Interbank. Una vez realizada la toma de control, cuando la nueva administración tomó las riendas de Interbank entre 1994 y 1995, se hizo necesario ejecutar las acciones que pongan a Interbank en situación para competir. Es así que dentro del Plan estratégico del banco se determinó entre muchas otras cosas la necesidad de



iniciar un profundo proceso de Reingeniería, el cual se inició a mediados del 1995 hasta fines de 1997.

A pesar que la experiencia presentada abarca todo el proceso de desarrollo de software, este informe se centra en el Proceso de Pruebas, por ser una de las fases más importante en el desarrollo de software y en la cual se consume al menos la mitad del tiempo y recursos asignados de manera formal a un proyecto.

Tareas de planificación y cambio deben enfocarse para, no solo realizar sus propias funciones, sino también para ayudar a reducir el periodo de pruebas. Decisiones como la estrategia de cambio, información de qué se ha cambiado o la realización de los cambios con calidad, pueden allanar el camino de pruebas y por tanto, reducir los recursos y tiempo necesarios para la realización de dichas tareas.

Es importante mencionar que la prueba de software es una actividad que debe estar presente a lo largo de todo el ciclo de vida y tanto el plan de pruebas como los casos de prueba deben ir siendo desarrollados en cada una de las etapas del ciclo de vida de Desarrollo de Software.

Si bien es cierto que en la experiencia presentada no se llegaron a implementar las estadísticas y mediciones de manera formal, en el presente informe se plantean propuestas de mejoras que incluyen principalmente registros y controles que permitan realizar mediciones y que nos permitan realizar un proceso de mejoramiento continuo.

# CAPITULO I

## ANTECEDENTES

### 1.1. EL BANCO

Interbank es una institución financiera resultado de la adquisición en una subasta de privatización convocada por el gobierno peruano en agosto de 1994, finalizando con más de dos décadas como banco estatal. Sin embargo sus orígenes se remontan al 11 de mayo de 1897, fecha de la creación del banco, teniendo a la fecha 104 años.

Hacia 1996, cuando se cambió el nombre a **Interbank**, se decidió convertir cada agencia en una auténtica tienda financiera en la que el cliente sintiera que accedía a un banco confiable y sólido, donde podía encontrar productos y servicios financieros brindados con asesoría necesaria y un trato especial.

Los resultados obtenidos por las tiendas financieras fueron motivo de gran satisfacción y, adicionalmente, Interbank decidió ser el primero en acercarse al cliente a través de sus tiendas ubicadas en la cadena de supermercados E.Wong y Metro, una iniciativa sin precedentes en el sistema financiero nacional, que logró llevar la banca al supermercado.

Interbank ha continuado progresando significativamente en su desarrollo hacia un banco moderno. Su sede principal, Torre Interbank, ubicada entre las avenidas Javier Prado y Paseo de la República, marca el inicio de una nueva era, con mejores servicios integrados, tecnología de avanzada y con los

mismos valores, filosofía y compromiso que garantizan que Interbank brinde un excelente servicio a sus clientes.

El banco ofrece a sus clientes:

- Una extensa red de tiendas a nivel nacional.
- Tiendas ubicadas en los Supermercados E. Wong y Metro, con atención de lunes a domingo de 9:00 a.m. a 9:00 p.m., inclusive feriados.

Además la más variada gama de canales de atención a distancia, que facilitan y agilizan el acceso a información y transacciones. Entre ellos destacan:

- Redactiva24, una red de cajeros a nivel nacional
- Fonoactivo 24, la banca por teléfono.
- Netactiva 24, la banca por internet.
- Saldoactivos, para consultas de saldos y movimientos de las cuentas

### **1.1.1. VISIÓN DEL BANCO**

Ser reconocidos como la mejor empresa de servicios financieros del Perú.

- Siendo la primera opción para los clientes en servicios y soluciones.
- Creando un ambiente de trabajo divertido donde los colaboradores se comprometen a dar lo mejor de sí, se valora la diversidad, se reconocen los aportes y hay oportunidad para el desarrollo personal.
- Alcanzando un crecimiento sostenido en utilidades, con una rentabilidad entre las más altas en la industria de los servicios financieros, y una condición financiera sólida.

### **1.1.2. MISIÓN DEL BANCO**

Ser una institución financiera líder que brinda un excelente servicio a sus clientes, ayudándolos a lograr sus aspiraciones, creando valor a los accionistas, dentro de un ambiente que propicie la realización del personal.

## **1.2. DIAGNÓSTICO ESTRATÉGICO**

### **1.2.1. DIVISIÓN DE SISTEMAS**

El área de sistemas se encontraba inmersa en la operativa de un banco antiguo, el cual no había tenido revisiones de sus procesos, pues venía de ser un banco del estado.

El área de sistemas en lugar de ser un apoyo a las diferentes áreas del banco empezó a ser un obstáculo en los negocios, debido a lo siguiente:

- Generación de aplicativos no acordes con la exigencia del mercado y de mala calidad
- Información de mala calidad, inoportuna y de difícil acceso
- Incapacidad de respuesta a requerimientos de los usuarios; los sistemas eran demasiado rígidos para soportar los cambios
- Falta de seriedad en los compromisos adquiridos
- Retraso en las instalaciones e ineficiente sistema de mantenimiento en Hardware y Software
- Constantes retrasos en la distribución de resultados, que en muchos casos no llegaban a su destino

- Frecuentes interrupciones y respuesta demasiado lenta en las redes de telecomunicaciones
- Mala atención a las consultas de los usuarios
- Carencia de Sistemas de Información Gerencial para la toma de decisiones

### 1.2.2. ANÁLISIS FODA DE LA DIVISIÓN DE SISTEMAS

Se presentará solo lo más resaltante, en relación al desarrollo de software.

Fortalezas	<ul style="list-style-type: none"> <li>• Buenos programadores</li> <li>• Capacidad de respuesta ante crisis</li> <li>• Capacidad de liderar proyectos</li> </ul>
Oportunidades	<ul style="list-style-type: none"> <li>• Nueva administración, con encargo de reingeniería</li> <li>• Rotación estratégica de puestos</li> <li>• Capacitación</li> </ul>
Debilidades	<ul style="list-style-type: none"> <li>• Desconocimiento detallado de aplicativos</li> <li>• Sistemas antiguos, obsoletos, poco flexibles</li> <li>• No había información de la competencia</li> <li>• No había documentación</li> <li>• Sistemas no integrados</li> </ul>
Amenazas	<ul style="list-style-type: none"> <li>• Personal no identificado con la nueva administración</li> <li>• Rápida obsolescencia del software de oficina</li> <li>• Priorización realizada por sistemas</li> </ul>

## **1.3. DIAGNÓSTICO FUNCIONAL**

### **1.3.1. PRODUCTOS**

Interbank contaba con aplicativos rígidos, mal diseñados y parte de ellos obsoletos (Sistema de cuentas corrientes del año 1969, otros del 70, 80, etc).

La operativa que existía en el banco, llevaba muchas veces a sistematizar procesos redundantes que algunas veces ni deberían existir.

### **1.3.2. CLIENTES**

Para el banco, como para toda organización lucrativa, existen dos tipos de clientes: los clientes internos y los clientes externos.

Los clientes internos, son los que laboran dentro de la organización, y que requieren los servicios del área de sistemas para ejecutar sus labores. Normalmente se les denomina usuarios y son los que definen y usan los sistemas del banco.

Los clientes externos, son las personas, que siendo externas al banco, le solicitan a éste que les satisfaga una necesidad específica relacionada con la actividad bancaria. Son la razón de ser de toda empresa de servicios, ya que ellos son usuarios de lo que ésta ofrece.

### **1.3.3. PROCESOS**

Se contaba con dos procesos básicos

- Desarrollo, en el cual se consideraban la ejecución de proyectos
- Mantenimiento, en el cual se consideraba toda modificación, mejora y error crítico

#### **1.3.4. ORGANIZACIÓN**

A continuación se presenta la organización actual tanto del banco, como de la División de sistemas, debido a que, a pesar de haber tenido cambios a lo largo de estos años, éstos no han tenido impacto en la Metodología de Pruebas y han sido producto de la búsqueda de mantener una mejor organización y aumentar la productividad de todas las áreas.

##### **1.3.4.1. Organización del Banco**

La estructura orgánica del banco busca enfatizar la rentabilidad de las distintas líneas de negocios y aumentar la productividad de todas las áreas.

Se ha definido que las áreas de negocios son: Corporativa, Empresa, Personas y Finanzas.

Las áreas de apoyo del banco son: Riesgos, Operaciones, Tecnología y Sistemas, Planeamiento, Legal, Auditoría y Gestión y Desarrollo Humano y Administración

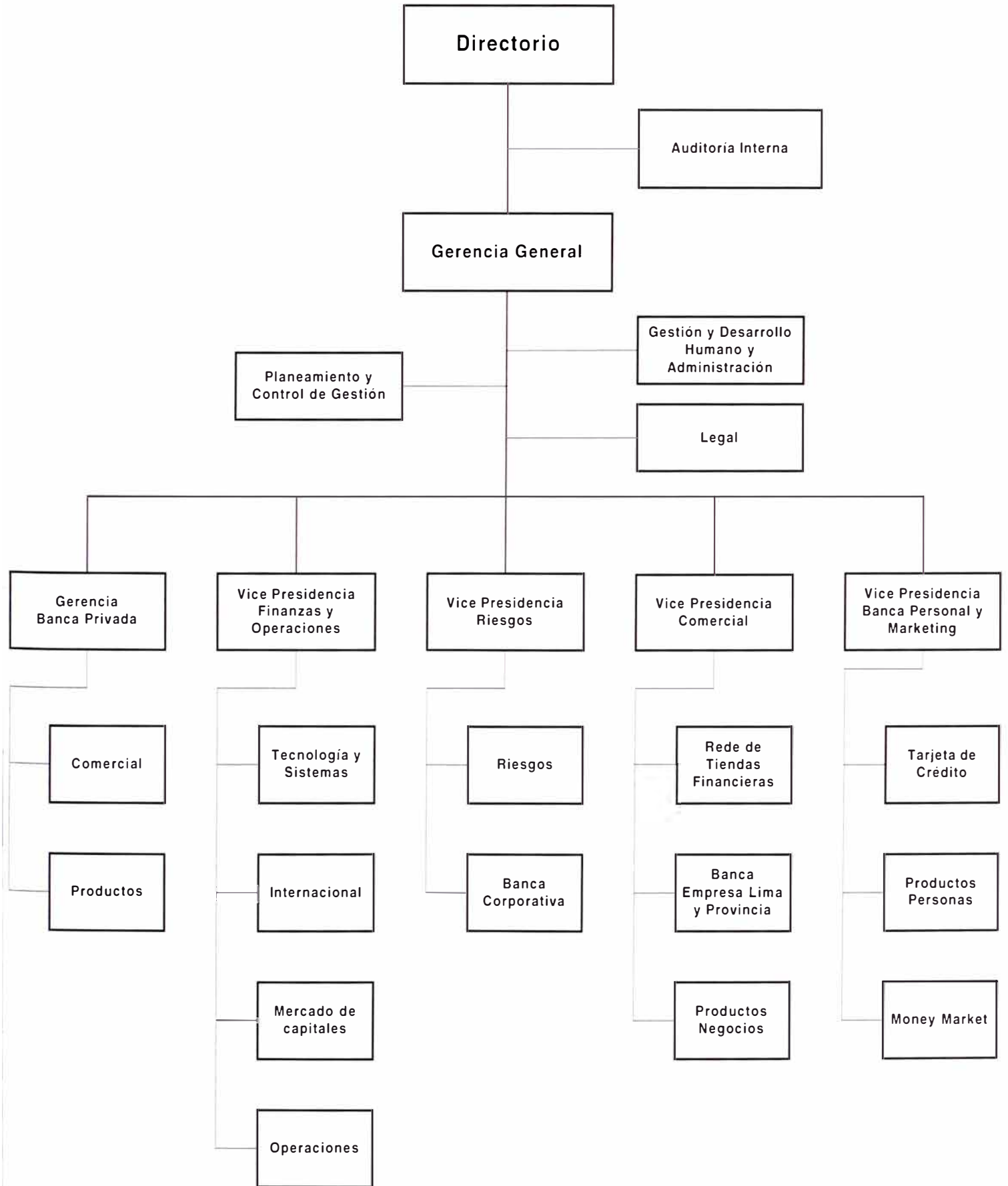
La organización del banco se presenta en el organigrama del Gráfico I.

##### **1.3.4.2. Organización de la División de Sistemas**

La *División de Sistemas* tiene la misión es apoyar al Banco en el cumplimiento de su *Plan Estratégico* brindando servicios en tecnología de información. Para ello, cuenta con herramientas tecnológicas de avanzada y apropiadas a la realidad del sistema financiero del país, las cuales son administradas por equipos humanos motivados, para contribuir a la plena satisfacción de los clientes con un costo/beneficio favorable.

La organización que se presenta en el organigrama del Gráfico II.

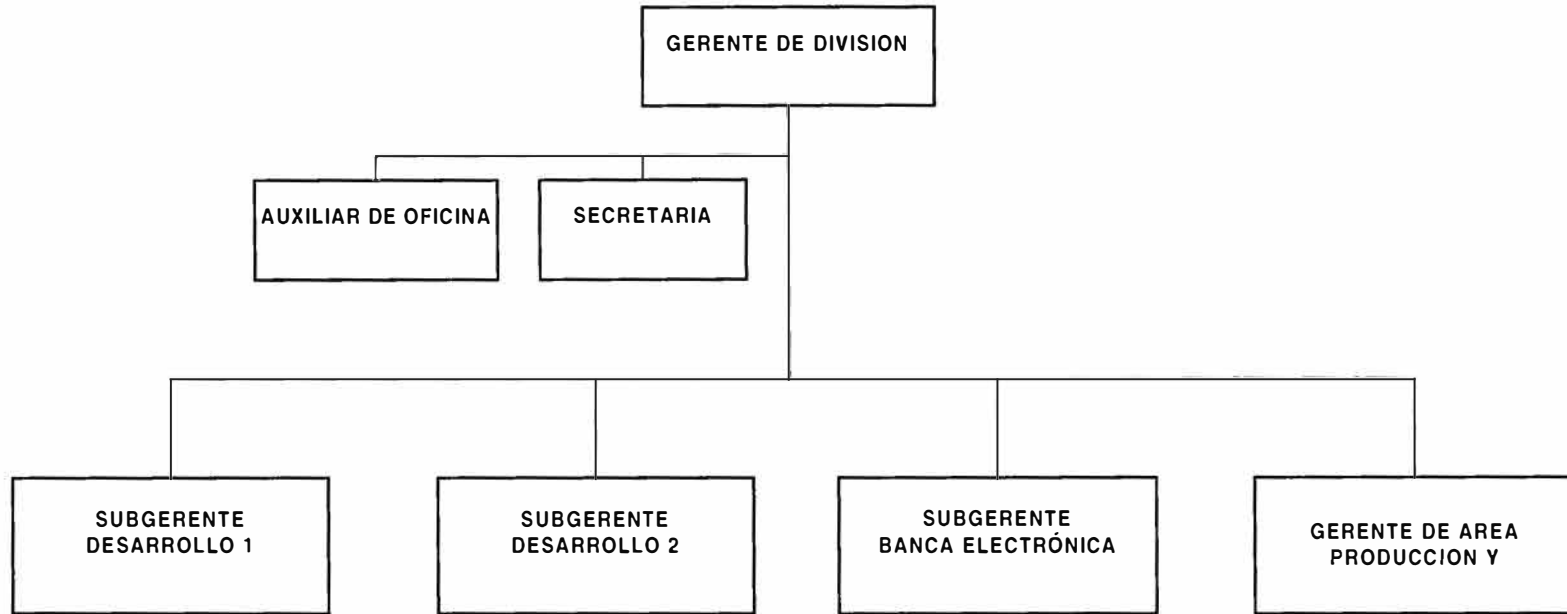
### ORGANIGRAMA DEL BANCO





# DIVISION DE SISTEMAS

Grafico II



## **CAPITULO II**

### **MARCO TEÓRICO**

#### **2.1. ASEGURAMIENTO DE LA CALIDAD EN LA ETAPA DE PRUEBAS**

El control de calidad (SQA) y la prueba de los sistemas se ha transformado en una actividad crítica, especialmente en ambientes en los que, como el financiero, un error puede tener graves consecuencias para la organización.

La actividad de pruebas de sistemas se caracteriza por su complejidad en dichos entornos, donde se trata con varias capas de equipos no homogéneos, con sistemas operativos, bases de datos, sistemas de comunicaciones y herramientas de programación variados y heterogéneos, en los cuales una transacción recorre los diferentes ambientes on-line y batch, expandiendo sus efectos a través de muchos programas y archivos. La organización compartida de los recursos, conlleva a que cada especialista desconozca, a menudo, las consecuencias que sus decisiones pueden tener en el resto del sistema. Adicionado a ello, un error que desde el punto de vista de codificación puede ser relativamente simple de corregir, resulta muy difícil de detectar y puede tener efectos gravísimos en la organización, tanto en lo interno, como en su imagen con los clientes: el no cobro de una comisión del sistema de cuentas corrientes puede causar pérdidas económicas

importantes, posiblemente irrecuperables; el cobro erróneo de una comisión que no corresponde causa un daño no menor en la imagen de la entidad.

Simultáneamente el equipo de desarrollo de software sufre continuamente presiones para cumplir fechas que a menudo son determinadas más por las necesidades del negocio que por consideraciones técnicas de factibilidad de cumplimiento, dejando de lado el hecho que un trabajo estimado para 9 meses no se hace en 1 mes con 9 personas. Como consecuencia de ello, el proceso de pruebas se realiza en forma más o menos desordenada, teniendo como objetivo el tratar de que 'las cosas funcionen'.

De allí surge el requerimiento de contar con personal especializado en tareas de pruebas, comprometidos con la calidad de los productos y que se dediquen a ello sin la presión de la coyuntura.

Son factores decisivos en el éxito de éstas tareas:

- La experiencia y formación de los recursos humanos, tanto en ingeniería de sistemas como en el 'universo' que es objeto del aplicativo
- La utilización de una metodología que ayude a sistematizarla, apuntando a ejecutar pruebas con una alta probabilidad de detectar errores
- La administración controlada del proceso de instalación de nuevas versiones
- El empleo de herramientas automáticas o semiautomáticas para cubrir un amplio rango de situaciones ante cada versión

Y aunque no suele tenerse en cuenta, un elemento psicológico:

'Que las personas a realizar las pruebas estén comprometidos prioritariamente con la detección d errores, y no con la defensa del producto'.

## **2.2. CONCEPTOS DE PRUEBAS**

En esta sección se mostrará los conceptos que son la base para entender la problemática de las pruebas.

### **2.2.1. FACTORES QUE DETERMINAN CANTIDAD DE PRUEBAS**

Dadas las complejidades en las aplicaciones actuales y de sus entornos, obtener un nivel de confianza de 100% de pruebas es extremadamente costoso, si no imposible de conseguir. En la práctica, determinar la cantidad adecuada de pruebas para una aplicación, conllevará a sopesar los cuatro puntos siguientes:

- Costo de pruebas, cuanto más amplio es el nivel de prueba más costosos son los recursos utilizados
- Costo de fallos, es una combinación de costo de corrección de errores de software, corrección de errores de operaciones y daños indirectos
- Probabilidad de fallos, relacionados a la probabilidad que ocurra un evento y probabilidad de cometer un error en la modificación
- Tolerancia a riesgos, cada aplicación tendrá su propia tolerancia a riesgos que estará por encima de cálculos más precisos

Es importante advertir que el nivel óptimo de prueba no puede decidirse sin datos ni análisis. Ni tampoco puede decidirse sin la asistencia de las áreas de negocio que forman la aplicación.

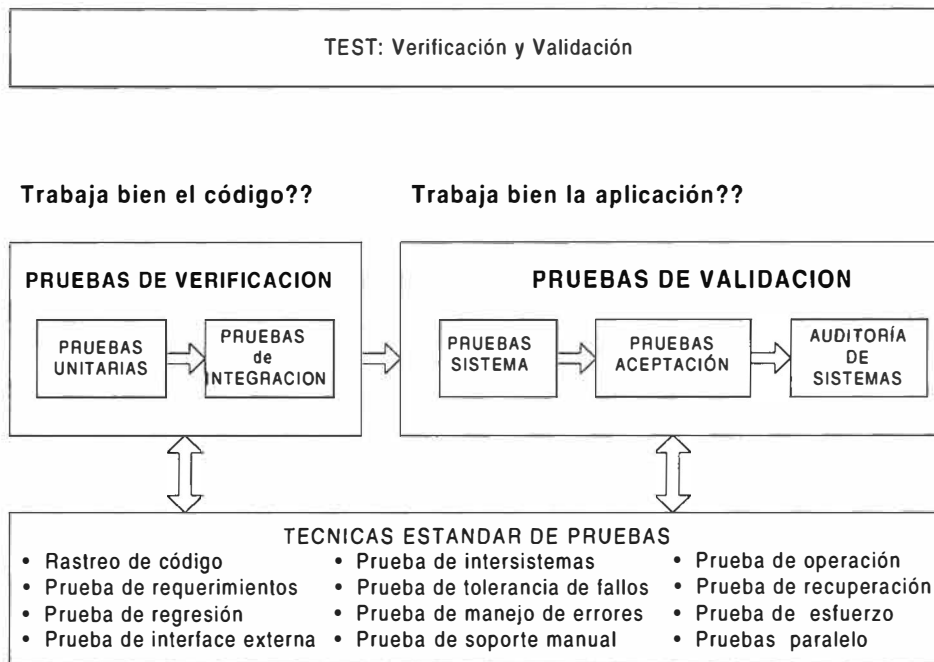
Sin una planificación por anticipado, los acuerdos se efectuarán precipitadamente exponiendo potencialmente en serios riesgos a la empresa en algunas de sus aplicaciones críticas.

Un análisis meticuloso permite seleccionar qué riesgos son aceptables y cuales son intolerables. Los recursos de prueba pueden encaminarse entonces a validar que los riesgos intolerables se hayan remediado. Finalmente, la cuestión de cuantas pruebas se necesitan es una decisión de negocio que deben tomar los encargados de una determinada aplicación.

### **2.2.2. NIVELES DE PRUEBA**

Cuando la mayoría de gente habla sobre el ciclo de vida de Pruebas, se están refiriendo a una progresión a través de los diferentes niveles de prueba. La aplicación típica se mueve secuencialmente a través de cada nivel durante el proceso de certificación. En cada nivel las pruebas varían según la parte de cada sistema probado, las herramientas utilizadas y el individuo que realiza la prueba.

En el nivel de aplicación, existen dos grandes categorías de pruebas: las pruebas de verificación, que se enfocan en los cambios de código individuales y las pruebas de validación, enfocadas en la funcionalidad de la aplicación. La relación entre éstas pruebas aparece en el siguiente gráfico:



### 2.2.2.1. PRUEBAS DE VERIFICACIÓN

Aquí se encuentran las Pruebas unitarias y Pruebas de integración.

Hay dos niveles de pruebas de verificación. Están relacionados en el enfoque y utilización de las herramientas para verificar la corrección de los cambios de códigos en lugar de enfocarse en la funcionalidad de las aplicaciones. Al dirigir la integridad de las modificaciones individuales en cada nivel de código su ejecución corresponde a los programadores y tienden a utilizar técnicas de prueba glassbox. Se centran en verificar que los componentes individuales (programas, JCL's, sentencias de control, archivos, bases de datos, etc.) cumplan los planes de diseño estructural y funcional. Ejemplo:

¿Ejecuta correctamente la lógica del programa una función particular siguiendo reglas de negocio definidas?

¿Detecta y maneja la lógica del programa ciertas condiciones de error?

¿Se ajusta el diseño físico del programa a los convenios de nomenclatura requeridos y los estándares de diseño estructurados?

En las técnicas de prueba de verificación , los resultados de las pruebas se revisan en forma manual por el programador que realiza la prueba.

**Las pruebas unitarias** comprueban la corrección de los cambios en el nivel del programa o módulo. Cualquier módulo actualizado durante el esfuerzo de corrección debe probarse de forma unitaria. La prueba unitaria implica el diseño de casos de prueba que validan que la lógica interna del programa funciona correctamente y que las entradas al programa producen salidas válidas. Estas pruebas generalmente las realiza el programador que ha modificado el módulo. Las pruebas se dirigen utilizando debuggers, simuladores de ejecución y herramientas de prueba interactivas. El programador crea pequeños archivos de datos de prueba para debuggers o selecciona puntos de decisión en un simulador de ejecución. La validación es realizada manualmente por el programador.

**Las pruebas de integración** comprueban las interfaces entre conjuntos de módulos. Durante la prueba de integración, los módulos individuales se combinan en grupos cada vez más extensos (por función, funciones relacionadas, subsistema, etc.). Los datos de prueba utilizados para las pruebas de integración, se focalizan en aquellos problemas ocasionados por versiones mal emparejadas o interfaces de módulos incorrectas (en lugar de intentar lograr una cobertura de prueba completa).

Las pruebas de integración pueden utilizar herramientas y técnicas para pruebas unitarias y/o pequeños grupos de integración, o bien, técnicas de pruebas de sistemas para grupos más extensos.

#### **2.2.2.2. PRUEBAS DE VALIDACIÓN**

Aquí se encuentran las Pruebas de Sistema, Pruebas de Aceptación y Revisiones de sistemas

Estas pruebas se clasifican como pruebas de validación y se ejecutan a un nivel de aplicación. A diferencia de las pruebas de verificación, son con frecuencia, responsabilidad de un equipo de prueba separado al de la modificación o grupo de control de calidad. Las pruebas de sistema, de aceptación y de revisiones de sistemas se enfocan en 'validar' que toda la aplicación, interfaces y entorno, funcionen adecuadamente dentro de las necesidades del negocio. Por ejemplo:

¿Produce la aplicación resultados correctos y/o salidas según las necesidades del negocio?

¿Se completa adecuadamente la transacción de negocios que implica distintas aplicaciones? (por ejemplo un mecanismo de ATM comunica a través de un procesador Tandem que a su vez recibe permiso para lanzar fondos desde la aplicación DDA, ejecutada en el mainframe?)

¿Se procesan debidamente, siguiendo los formatos estándar, los archivos de intercambio de datos?



¿Se recupera correctamente, con total integridad, el entorno de aplicación y procesamiento en caso de desastre (ejemplo: falla en el hardware o en el software)?

**Las pruebas de sistemas** implican casos de prueba diseñados para validar que una aplicación y sus componentes de soporte de hardware/software estén procesando adecuadamente datos de negocios y transacciones. Las pruebas de sistema requieren el uso de técnicas de regresión de prueba para validar que las funciones de negocio cumplen unos requisitos definidos. Las pruebas de sistema cubren una aplicación entera y cubren además todas las facetas de la funcionalidad de aplicaciones, además de estar diseñadas para reflejar la operación de producción de las aplicaciones. Estas pruebas pueden dirigirse utilizando fechas actuales o futuras y pueden utilizar bien la técnica de prueba glassbox o bien la técnica blackbox.

**Las pruebas de aceptación** (Prueba de usuario) implican que los usuarios finales realicen la prueba final de una aplicación antes de su lanzamiento a producción. La cuestión es si una aplicación cumple o no las necesidades del usuario y si produce resultados correctos.

**Las revisiones de sistemas** validan el desarrollo funcional en curso de una aplicación una vez que se ha llevado a Producción. Revisa los resultados diarios de operación del sistema para asegurar que la aplicación continúa operando correctamente.

### **2.2.3. CATEGORÍAS DE PRUEBAS**

Las categorías de pruebas se refieren a los tipos de prueba ejecutadas utilizando las técnicas descritas por los niveles de prueba. Por ejemplo, una prueba de regresión puede ejecutarse en todos los niveles del ciclo de vida de la prueba. Hay varios tipos de prueba que se muestran a continuación:

#### **Prueba de Regresión**

Están diseñadas para asegurar que la funcionalidad existente no ha sido afectada por algún cambio. Las pruebas de regresión se ejecutan utilizando la fecha actual y los resultados de las pruebas se validan mediante comparaciones de ejecuciones paralelas de versiones sin actualizar.

#### **Pruebas de fechas límite**

Se utilizan para chequear condiciones especiales y eventos que surgirán en el proceso de una cierta fecha (fin de mes, fin de año, años bisiestos, etc.)

#### **Pruebas de interfaces externas**

Comprueba las conexiones que existen entre aplicaciones. Estas conexiones pueden darse a través de almacenamientos de datos como archivos o conexiones de comunicación. Las aplicaciones que tienen interfaces deben probarse juntas para asegurar que las interfaces funcionan correcta y consistentemente y siguen la misma estrategia de implementación.

#### **Pruebas de tolerancia a fallos**

Comprueban la capacidad que tiene una aplicación para manejar datos erróneos desde otras aplicaciones. Las aplicaciones muy críticas que se basan en datos creados externamente pueden necesitar protección contra

datos no actualizados o incorrectos que se introducen por fuentes externas. Esta prueba utiliza datos de prueba creados que anticipan los tipos de errores que pueden darse, de forma que pueda probarse la capacidad de la aplicación para manejar dichos errores.

## **CAPITULO III**

### **PROCESO DE TOMA DE DECISIONES**

#### **3.1. PLANTEAMIENTO DEL PROBLEMA**

Los problemas que se presentaban en el área de sistemas y principalmente relacionados con la etapa de pruebas fueron:

1. Los aplicativos existentes no estaban acordes a los requerimientos del mercado (software pobre, antiguo, inestable y con mucho mantenimiento). Los cambios generaban nuevos errores y mantenimientos simultáneos sobre el mismo aplicativo.
2. La información generada por sistemas era inoportuna, incompleta y de difícil acceso. Se generaba una misma información de diferentes fuentes con diferentes resultados.
3. Incapacidad de respuesta a los requerimientos de los usuarios. Los sistemas eran demasiado rígidos para implementar los cambios.
4. Los problemas por errores en programas eran frecuentes, tanto en batch como en línea y por ende el porcentaje de recursos asignados a mantenimiento era alto. Diariamente había un promedio de 4 a 5 personas que daban soporte nocturno.

5. Los usuarios no estaban identificados con las pruebas ni con todos los cambios realizados, sin embargo ellos eran responsables de autorizar los pases a Producción.
6. No existía documentación de las pruebas y muchas veces eran insuficientes o incompletas. Por ejemplo muchas veces no se realizaban pruebas de regresión o errores, o los usuarios 'confiaban' en las pruebas de los analistas de sistemas.
7. No se contaba con una metodología única de desarrollo de software ni con un equipo multidisciplinario para la elaboración de los proyectos.
8. No se contaba con metodología de pruebas ni ambientes adecuados de prueba. Solo se tenía ambiente de pruebas unitarias y ambiente de producción. No existía ambiente de pruebas para usuarios
9. No se contaban con las herramientas automatizadas para optimizar la etapa de pruebas.
10. No existía un procedimiento de pase a Producción estandarizado, que permitiera una adecuada administración de los componentes. Determinados grupos de aplicaciones tenían diferentes librerías, diferentes procedimientos de pase, en algunos casos no existían los fuentes de los programas y en otros los fuentes de producción estaban en librerías de prueba.
11. Existía una sobrecarga de trabajo descontrolado y desorganizado, lo cual incidía en el incumplimiento de plazos.

### 3.2. ALTERNATIVAS DE SOLUCIÓN

#### 1. Implementación de Reingeniería y Mejoramiento de Procesos

Que estaba comprendido dentro del Proceso de Modernización del banco y comprendería los siguientes proyectos simultáneos:

- Búsqueda y selección de un nuevo software integrado bancario para reemplazar las aplicaciones vigentes en ese momento.
- Mejoramiento de procesos de sistemas, que comprendía la formulación de Políticas y Procedimientos para Sistemas así como de una Metodología de Desarrollo de Software Aplicativo con las cuales se define el modo de trabajar para elevar la calidad de los productos tecnológicos del Banco.
- Elaboración del Plan de Sistemas, que consistía en el uso de una Metodología para llevar a cabo el primer proceso de Planeamiento Tecnológico en el Banco.

#### 2. Implementación de Modelo de Capacidad de la Madurez (CMM)

CMM es una guía para las organizaciones en la selección de estrategias de mejora, determinando la madurez del proceso actual e identificando los puntos importantes que se deben aplicar para así mejorar tanto el proceso como la calidad del software.

La madurez de un proceso implica que la capacidad del proceso de software ha crecido. Específicamente debe ser: definido, documentado, entrenado, practicado, soportado, mantenido, controlado, verificado, validado, medido y capaz de mejorar.

### 3. Implementación de ISO9000

ISO9000 es una serie de estándares internacionales para sistemas de calidad. Especifica las recomendaciones y requerimientos para el diseño y la valoración de un sistema de gestión, con el propósito de asegurar que los proveedores proporcionen productos y servicios que satisfagan los requerimientos especificados.

El preparar la organización para que cumpla con los estándares ISO9000 implica 3 aspectos principales:

- Decir lo que hay que hacer (escribir procedimientos)
- Hacer lo que se dijo que se iba a hacer (procesos)
- Mostrar evidencia de que se está haciendo lo que se dijo que se iba a hacer (registros)

Además ISO9000 obliga a las empresas establecer un mecanismo de mejoramiento continuo de la calidad, esto se logra mediante un ciclo de mejoramiento en el cual se realizan periódicamente auditorías internas y revisiones gerenciales sobre las cuales se toman acciones correctivas.

### **3.3. METODOLOGÍA DE SOLUCIÓN**

Todas las empresas deben de considerar la calidad como su preocupación central, es decir estar orientadas hacia la satisfacción de las necesidades de sus clientes.

En el banco, hacia el año 1996 se elaboró el Plan estratégico del banco, en el cual, entre otras muchas cosas se determinó la necesidad de iniciar un

profundo **Proceso de Reingeniería para todo el banco**, iniciadas con las de Recursos Humanos, Negocios, Operaciones y Sistemas.

Para el proceso de Reingeniería del área de Sistemas se realizaron las siguientes actividades:

- Conformación del equipo de Reingeniería
- Contratación de consultores externos
- Capacitación a los miembros del equipo de Reingeniería

Además el equipo de Reingeniería tuvo como encargo las siguientes 3 actividades principales: Búsqueda y Selección del Software Integrado, la elaboración del Plan de Sistemas y la Modernización de Sistemas.

### **3.4. TOMA DE DECISIONES**

Durante el año 1995, el banco tomó decisiones trascendentales sobre la que sería su filosofía tecnológica para los años venideros.

La primera decisión tomada fue que la solución principal que se adoptara para las aplicaciones transaccionales y bancarias tradicionales en el mainframe, debería estar complementada e interconectada con otras aplicaciones cliente/servidor, por ser ésta la tendencia moderna de más alta productividad en el desarrollo de aplicaciones. Se escogió como plataforma preferida para las nuevas aplicaciones cliente/servidor, el Back Office de Microsoft por la capacidad de conexión muy poderosa con las aplicaciones residentes en el mainframe y en cualquier otra plataforma.



La segunda decisión importante, fue decidir la compra de un paquete integrado de software bancario que reemplazó a sus aplicaciones existentes, descartando tanto la posibilidad de mejorarlos como la idea de migrar progresivamente a un ambiente cliente/servidor.

Paralelamente, el Banco se comprometía en un proceso de Reingeniería por áreas, iniciadas con las de Recursos Humanos, Negocios, Operaciones y Sistemas.

Como parte de la Reingeniería de Sistemas y después de un proceso de selección muy riguroso con consultores, la tercera decisión adoptada y sin duda la más trascendental, fue la contratación con la empresa Alltel Information Services (Systematics) para adquirir una solución integral, tanto de software transaccional y de aplicaciones bancarias tradicionales, como de varios módulos cliente/servidor complementarios de éstos, y sobre todo aquellos en los que residen varios sistemas de información gerencial.

En el área de Sistemas se decidió por un Proceso de Reingeniería y Mejora de Procesos, básicamente porque ya existía un lineamiento general a nivel del banco. Sin embargo en ese momento de las posibles alternativas, fue la más adecuada debido a lo siguiente:

- El banco es una institución cuya actividad principal no es el desarrollo de software y habiendo pasado de ser una entidad pública a privada, se encontraba con mucho desorden, procesos ineficientes, información mal generada y no usada, etc. El primer paso era ordenar y tener procesos bien definidos.

- CMM también pudo ser una alternativa viable, sin embargo en el banco se tenía prácticamente en simultaneo la elaboración del Plan de sistemas y la implementación del nuevo software integrado, para lo cual se necesitaba contar con la metodología de desarrollo de software y con todas las herramientas necesarias para lograr una implementación exitosa.
- Certificar bajo la norma ISO9000 es una tarea ardua y costosa pero además no nos garantiza una metodología de desarrollo, ni un rediseño de procesos, lo cual era necesario en el estado que se encontraba el área. Además ahora podemos ver que en el tiempo ha surgido la nueva norma ISO9000-2000, la cual busca reducir la complejidad de los documentos y poner más énfasis en Mejoramiento continuo, participación gerencial, administración del negocio basado en información, satisfacción del cliente, mejoramiento de los procesos y mejor aplicación a todos los tipos de organización y negocios.

La decisión del proceso de Reingeniería y Mejoramiento de procesos trajo en el transcurso del tiempo buenos resultados: Se ha logrado tener procesos ordenados, definidos, documentados y conocidos por el personal del sistemas y usuarios. Sin embargo, a partir de este punto se puede pensar en implementar CMM, que solo se orienta a la producción del software y en un futuro certificar con ISO9000, que incluye toda la organización.

### **3.5. ESTRATEGIAS ADOPTADAS**

En el banco se implementó la Metodología de Pruebas basada en la Reingeniería y se complementó con la Metodología de trabajo de la empresa proveedora del nuevo software integrado del banco Systematics (Ambientes de prueba).

#### **3.5.1. REQUERIMIENTOS**

Como la metodología de Pruebas forma parte de la Metodología de Desarrollo de Software, a continuación se presentan los requerimientos para definir el proceso:

- Optimizar el proceso de desarrollo de software
- Obtener un nivel de calidad satisfactoria para los productos y servicios del banco
- Satisfacer las expectativas de los usuarios
- Proporcionar a los usuarios del banco que se constituyen en los clientes de la División, los sistemas de información que requieren para desarrollar sus negocios con la mayor flexibilidad, productividad y oportunidad, siendo capaces de entender y potenciar sus reales necesidades de información y adecuarse a las prioridades de estos usuarios.
- Dotar al Banco de la tecnología de información más avanzada y apropiada para el negocio de la banca, tanto en software como en hardware y telecomunicaciones, con un claro criterio de costo/beneficio.

- Crear y mantener en todo el personal que labora en la División, una cultura de servicio al usuario con una adecuada planificación conjunta y dinámica.
- Mantenerse al día de los constantes avances tecnológicos y concretarlos en propuestas viables a los usuarios para que mejoren el desempeño de sus funciones.
- Adelantar a la competencia bancaria en el país, proponiendo a los usuarios, soluciones tecnológicas para hacer frente a sus problemas o capitalizar oportunidades del entorno.
- Mantener informados a los usuarios de la *División de Sistemas*, de los reales niveles de desempeño creando una cultura de veracidad y mutua confianza.

### **3.5.2. MISIÓN DE LA DIVISIÓN DE SISTEMAS**

“Brindar servicios y productos de tecnología de información con Calidad, Oportunidad y Flexibilidad utilizando los recursos eficientemente para apoyar al banco en el cumplimiento de sus objetivos”

### **3.5.3. ESTABLECIMIENTO DE UNA POLITICA DE CALIDAD**

La política de calidad de la División de Sistemas es satisfacer los requerimientos de los usuarios, brindando productos y servicios de software que permitan lograr y mantener el máximo nivel de servicio, enfocando los esfuerzos al mejoramiento continuo de los productos y procesos.

### **3.5.4. PROCESO DE DESARROLLO DE SOFTWARE**

El proceso se inicia con un requerimiento de desarrollo de software aplicativo para un producto o servicio del Banco planteado por un Usuario Responsable, y finaliza :

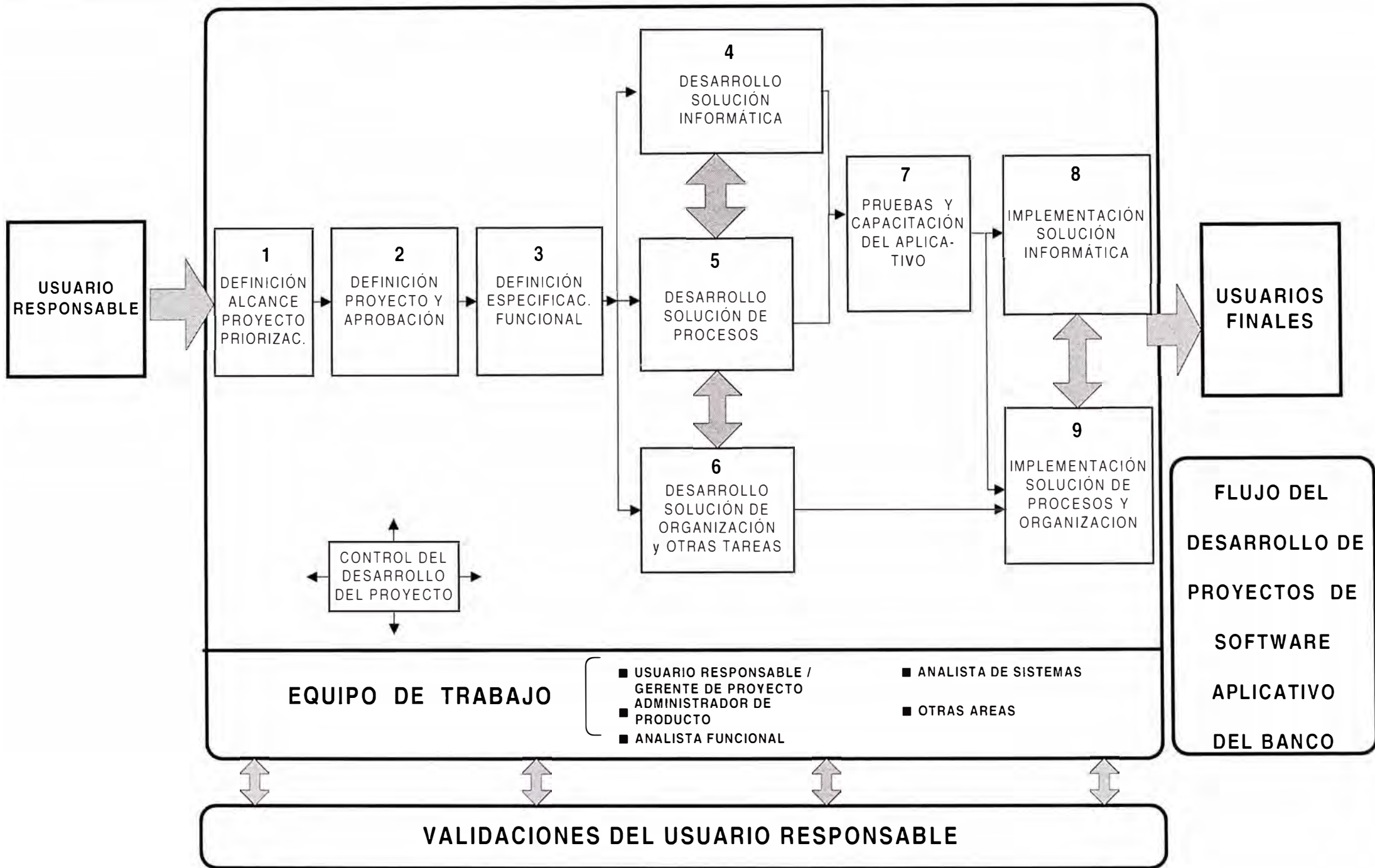
- Si el proyecto es aprobado, cuando la solución se implementa a nivel nacional
- En caso contrario, cuando el proyecto es repriorizado o desechado.

El desarrollo de un proyecto se lleva a cabo a través de nueve (9) fases y su flujo general se presenta en el Gráfico III.

#### **3.5.4.1 Políticas y Estrategias**

Como producto del proceso de Reingeniería de Sistemas, la actividad de desarrollo de software aplicativo se ha modernizado y se han determinado estrategias importantes para asegurar la calidad y oportunidad del servicio.

- Todo proceso de desarrollo y mantenimiento se realizará cumpliendo la metodología presentada por el *Manual de Desarrollo de Software Aplicativo*
- Las actividades necesarias en cada fase de desarrollo se cumplirán de conformidad a lo establecido en el documento de *Políticas y Procedimientos de la División de Sistemas*.
- Todo desarrollo realizado con terceros deberá considerar en la labor de análisis y diseño a personal del Banco, que asegure que el conocimiento sobre el producto final quede en el banco.



### **3.5.4.2 Conceptos Pilares en el desarrollo de Software**

El desarrollo de sistemas en el Banco se fundamenta en los conceptos que se presentan a continuación:

#### **Procesos**

Desarrollar un software aplicativo significa : Definirlo, Producirlo o comprarlo, Implementarlo, y Mantenerlo. Dentro de sistemas se establecieron los siguientes procesos:

- Desarrollo de Proyectos, cuando el desarrollo del aplicativo impacta al Banco y demanda un esfuerzo mayor o igual a un mes/persona
- Requerimiento menor, cuando el esfuerzo es menor a un mes/persona. Generalmente no requiere cambios en los procesos del Banco, sólo cambios menores del aplicativo. (ej.: campo nuevo en pantalla o reporte ).
- Crisis, cuando la solución debe ser inmediata. Dan solución a problemas que interrumpen la atención al cliente y paralizan la operativa del Banco (ejemplo: caída de comunicaciones, descuadre de aplicativos, transacción que rechaza cargar/abonar una cuenta, etc.).
- Desarrollo de Software departamental, cuando se trata de la construcción de aplicativos puntuales y aislados, que no impactan los aplicativos centrales del Banco, y tiene un tratamiento particular.

#### **Usuario Responsable**

Es el responsable total por el adecuado funcionamiento de uno o varios aplicativos, desde su definición y desarrollo, hasta su explotación y

mantenimiento. En esta labor cuenta con el soporte de diferentes entes, quienes tienen la responsabilidad de ejecución de ciertas actividades.

### **Cronograma de desarrollo de proyectos**

Es una lista de proyectos de desarrollo de software aplicativo propuestos y priorizados por los *Usuarios Responsables*, en coordinación con sus *Vicepresidencias*, quienes definen el orden y la fecha de atención de cada proyecto.

Se actualiza con nuevos proyectos o redefiniendo, repriorizando, desechando los ya establecidos, siendo la *División de Sistemas* responsable de su actualización y publicación trimestral. El Usuario Responsable y la División de Desarrollo Organizacional monitorean el cronograma y disponen de los recursos necesarios para iniciar el proyecto correspondiente.

### **Equipo de trabajo**

Generalmente está conformado por el Usuario Responsable (Líder del equipo), Administrador de Producto, Analista Funcional y el Analista de Sistemas. Adicionalmente, el Usuario Responsable puede designar a un Gerente de Proyecto (funcionario de la misma área u otra del Banco) para su administración, permaneciendo la responsabilidad por el proyecto en el Usuario Responsable. Asimismo, dependiendo de la naturaleza del proyecto, convoca a otras áreas del Banco (Oficina de Auditoría, Oficina Legal, Mercadeo, Planeamiento, Red de tiendas, etc.) para formar parte del Equipo de Trabajo.



Este equipo se encarga tanto del mantenimiento de aplicativos como de nuevos desarrollos.

Una explicación detallada de las responsabilidades del equipo de trabajo se muestra en el Anexo I.

### **Componentes de la solución**

Todo requerimiento de soporte tecnológico es analizado de manera integral y sinérgica por el Equipo de Trabajo. Los componentes de su solución son

- Solución de procesos
- Solución de Organización y otras tareas
- Solución Informática

Una explicación más detallada se muestra en el Anexo II.

### **3.5.5. IMPLEMENTACIÓN DE LA METODOLOGÍA DE PRUEBAS**

A continuación se presenta todos los aspectos considerados para la implementación de la Metodología de Pruebas en el banco.

#### **3.5.5.1 Estrategias para la implementación**

La implementación se realizó por etapas, la cuales se muestran a continuación.

1. Del proceso de Reingeniería se obtuvieron los siguientes productos y se realizaron diversas actividades para la implementación:
  - Elaboración de los manuales de Metodología de Desarrollo de Software Aplicativo, que incluía el tratamiento para proyectos, mantenimientos, urgencias y desarrollos departamentales

- Elaboración del manual de Políticas y Procedimientos de la División de Sistemas
  - Elaboración del documento de Funciones del área de sistemas
  - Charlas a la División de Sistemas sobre dichos manuales
  - Capacitación en manejo de herramientas para control del proyecto
  - Capacitación en el registro de actividades en el sistema
  - Charlas de la metodología a los usuarios (usuarios responsables, administrador del producto, analistas funcionales)
  - Definición de responsables por cada aplicativo (usuario responsable, administrador de producto y analista de sistemas)
2. En el proceso de Reingeniería se determinó la necesidad de crear el área de Aseguramiento de la Calidad y Servicio, la cual tuvo el encargo de crear un ambiente para pruebas de usuario, que se llamó ambiente de Pre-Producción y luego con la implementación del nuevo software integrado se implementó la Metodología de Pruebas, para lo cual se realizaron las siguientes actividades
- Se inició con la actividad de ordenamiento e inventario de todo el software del banco. Se asignaron responsables en sistemas por aplicativo y se elaboró documentación básica.
  - Se crearon 3 ambientes de prueba similares a producción (De Pruebas unitarias, De Integración, De Aceptación Usuaría), con la diferencia en la cantidad de data y algunos productos.

- Se unificaron librerías, procedimientos de compilación y se crearon herramientas para la migración entre ambientes.
  - Se adquirieron discos para la creación del ambiente y herramientas de ayuda para la creación de data de prueba y seguimiento de errores
  - Se definieron los formatos de prueba, formatos de manuales y documentación de la metodología de migración entre ambientes.
  - Se unificó el manual de estándares y se dieron charlas sobre los ambientes y estándares
  - Se coordinó la capacitación en el uso de las herramientas
3. Posteriormente con el problema del Año 2000, el área de Aseguramiento de la Calidad y Servicio pasó a formar parte del Proyecto Año 2000 y se complementaron algunos aspectos, como:
- Se adquirieron más herramientas de prueba, sobre todo las relacionadas con fechas
  - Se adquirió una herramienta de Control de cambios
  - Se adquirió más espacio en disco y se aumentó la capacidad de máquina
  - Con el análisis de impacto se realizó la documentación de todo el software existente.

### **3.5.5.2 Descripción del proceso de Pruebas**

En esta sección se muestra la Fase de Pruebas que es parte del Proceso de Desarrollo de Software Aplicativo. Aquí se ejecutan las pruebas funcionales y técnicas del aplicativo, según los planes de pruebas definidos.

El objetivo es obtener las certificaciones de que el aplicativo cumple con los resultados esperados : Certificación Funcional y de Eficiencia. Asimismo, validar el *Manual de Usuario* y manuales técnicos del aplicativo.

De acuerdo al Plan de Capacitación, se capacita a los encargados de implantar la solución a nivel nacional (usuarios finales de la misma área del Administrador de Producto o de otras áreas, Coordinadores Operativos de la División de Desarrollo Organizacional). Asimismo, se capacita a la Sección Soporte Informático de la División de Sistemas, la cual absolverá las consultas y/o problemas técnicos ocurridos durante el uso del aplicativo.

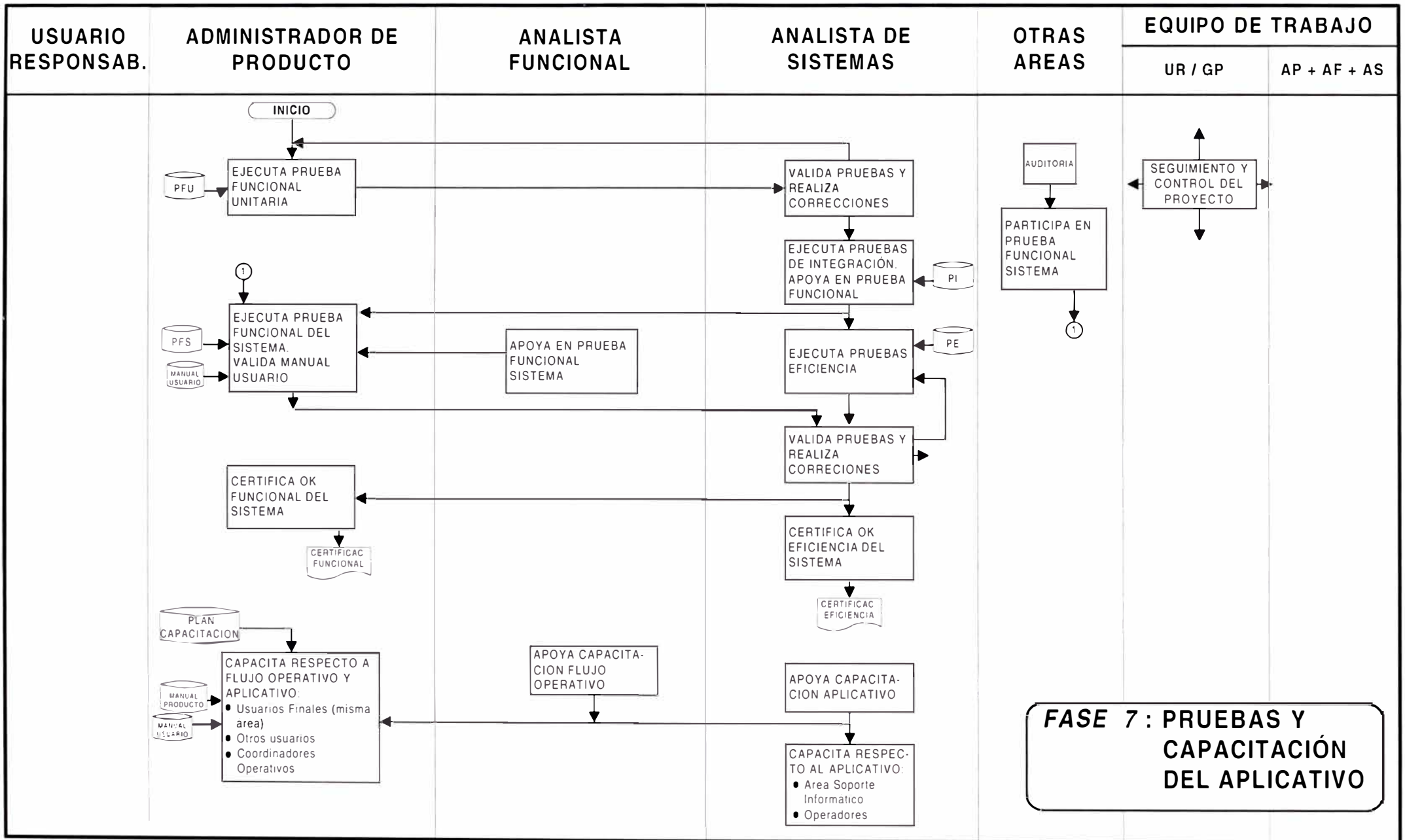
El flujo de la Fase de Pruebas y Capacitación del Aplicativo (Fase 7) se muestra en el Gráfico IV.

### **3.5.5.3 Ambientes de Prueba**

El Banco cuenta con un adecuado ambiente de certificación implementado y dedicado exclusivamente para este fin. En este ambiente se replican los siguientes escenarios:

- Equipamiento y software tipo de la red de tiendas
- Estaciones de trabajo de acceso a los sistemas Host de proceso centralizado
- Plataformas de servidores y PC's
- Soluciones departamentales en PC
- Cajeros automáticos
- Saldoactivos
- Estaciones de cliente y servidor (software instalado en clientes empresa)

Gráfico IV



El ambiente está formado por 3 ambientes de pruebas:

1. Ambiente de Pruebas Unitarias, se realizan las pruebas de código por programa, pruebas de archivos, procesos sin data, errores, etc. Es ejecutada por el programador. También se realiza inspección del código.
2. Ambiente de Pruebas de Integración, se realizan las pruebas y la interacción con otros sistemas. Se realizan pruebas cruzadas, es decir las pruebas no necesariamente las realiza quien hizo la programación. Se incluyen pruebas en paralelo, de regresión, de esfuerzo, etc. La migración al ambiente de Aceptación Usuaría es la prueba de Instalación y debe hacerse exactamente igual a como se realizará el pase a Producción.
3. Ambiente de Pruebas de Aceptación Usuaría, es un ambiente similar a Producción, donde los únicos autorizados a operarlo son los usuarios. Tiene un proceso batch propio que se ejecuta todos los días igual que Producción, pero con hora de inicio más temprano y se le da soporte igual que en Producción.

En este ambiente también se realizan las pruebas de volumen, tomando tiempos de proceso y realizando el análisis de uso de recursos, carga, concurrencia, etc. Estas pruebas se hacen en conjunto con personal de Ingeniería de sistemas. Luego de certificar en este ambiente se migra al Ambiente de Producción.

Adicionalmente se cuenta con:

- Ambiente de Capacitación, similar al ambiente de aceptación usuaria, usado para capacitar principalmente al personal de tiendas.

- Ambiente de Dress Rehearsal, implementado y usado para proyectos grandes, tipo las experiencias del Proyecto Año 2000 o la Implementación del software integrado Systematics.

Es un ambiente exactamente igual a Producción y se realizan pruebas de volumen, carga del sistema, se toman tiempos de ejecución, etc.

#### **3.5.5.4 Estrategia de pruebas**

La estrategia y metodología básica para enfrentar las pruebas de aceptación usuaria se centra en el concepto de Eventos de Negocio.

Desde el punto de vista de negocios, existen servicios apoyados por diversas aplicaciones en una variedad de plataformas. Cada aplicación de negocios está concebida para apoyar diferentes funciones específicas, las cuales en su procesamiento generan lo que se llama un Evento de negocios, es decir desde su ingreso por alguna aplicación, su información es procesada por un conjunto de aplicaciones que modifican su estado de acuerdo a los datos originales y eventos que se producen en el ámbito de su vigencia. Es decir, un evento de negocio es un conjunto de actividades y/o tareas todas ellas apoyadas por uno o más sistemas computacionales, que reciben como entrada uno o más insumos (solicitudes) y generan como salida un producto para un cliente interno o externo.

Todo evento de negocio tiene un inicio y un término bien definidos. Su inicio se define en función a alguna solicitud que inicia un cliente externo o interno y su término es un resultado concreto para este cliente u otro.

En el anexo III se muestra un ejemplo de definición de eventos de Negocio.

### 3.5.5.5 Herramientas

El banco cuenta con herramientas de prueba que facilitan la labor de creación de data de prueba, debugging, pruebas de esfuerzo, identificación de errores, etc. Estas herramientas automatizadas utilizadas en las diferentes etapas de las pruebas son detalladas a continuación y son de gran utilidad en el proceso de pruebas del banco.

- Data Ager, es una herramienta que facilita el envejecimiento o rejuvenecimiento de los datos de los archivos de una o varias estructuras de datos, usado para la simulación de las pruebas con fechas.
- Xpediter/Xchange, es una herramienta que permite simular selectivamente distintas fechas del computador sin modificar las fechas de la partición lógica del Sistema, sean estas modificaciones para procesos On-line o Batch, procesos agrupados o unitarios.
- File Aid , es una herramienta poderosa que permite modificar y comparar el contenido de los archivos, de cualquier tipo de estructura.
- Play Back, es una herramienta que permite capturar las transacciones en línea y volverlas a ejecutar tantas veces como sea necesario en forma automatizada, con capacidad de modificación de la trama y de comparación del input y el output, por lo que resulta muy útil en la creación y re-utilización del Baseline. Muy útil en la prueba de esfuerzo del sistema pues permite manejar la velocidad y la cantidad de transacciones en el proceso de re ejecución.



- Xpediter, es un “debugger” que permite aumentar significativamente la productividad de los analistas de Sistemas, al facilitar la depuración de los programas y así evitar riesgos en los pases a producción.
- Abend Aid, es una herramienta que detecta los errores a tiempo de ejecución y facilita su detección, pudiendo complementarse con el xpediter.

#### **3.5.5.6 Estándares**

La División de Sistemas cuenta con estándares que norman los diferentes procesos de Desarrollo de software y Pruebas.

- Metodología de desarrollo de software aplicativo
- Manual de políticas y procedimientos de sistemas
- Estándares de nomenclatura de componentes de aplicativo
- Procedimiento de migración entre ambientes
- Estándares para la documentación técnica de un nuevo aplicativo

#### **3.5.5.7 Preparación y Ejecución de la pruebas**

Previo al proceso de Pruebas de Aceptación Usuaría y de acuerdo al cronograma del proyecto, el equipo de trabajo del proyecto prepara el ambiente de acuerdo al Plan de Pruebas establecido. Realiza las siguientes actividades:

- Preparación de los eventos del negocio, casos de prueba y Test scripts
- Revisión de los eventos del negocio, casos de prueba y Test scripts
- Preparación de los datos de prueba
- Preparación del ambiente de prueba

- Coordinación con los usuarios de otros sistemas involucrados en el evento de prueba
- Coordinación de accesos al sistema y reportes.

#### **3.5.5.8 Registros**

Dentro del proceso de pruebas, se cuentan con los siguientes formatos que sirven para registrar todo lo relacionado a las pruebas. El usuario que realiza las pruebas es el encargado de elaborarlos y se encuentra familiarizado con estos formatos, sin embargo cuenta con el soporte y apoyo de las áreas de Sistemas y Desarrollo Organizacional.

- Formato para registrar los eventos de negocio (Anexo IV)
- Formato para registrar el inventario de casos de prueba por evento de negocio (Anexo V)
- Formato para registrar los test scripts por caso de prueba (Anexo VI)
- Formato para registrar los problemas durante las pruebas (Anexo VII)
- Formato para registrar el Informe del proceso batch (Anexo VIII)
- Formato de Certificación, actualmente no se usa, la conformidad y certificación se realiza vía e-mail del usuario responsable de las pruebas.

#### **3.5.5.9 Documentación**

- Manual de usuario
- Manual del producto y/o instructivos, elaborado por el área de Desarrollo Organizacional

- Manual del proceso, elaborado por sistemas y requisito indispensable para el pase a Producción
- Plan general de pruebas

### **3.5.5.10 Estadísticas**

#### Ambiente de Producción y Aceptación Usuaría

- Proceso Batch

Se realizan estadísticas de los errores que causan cancelaciones en el proceso nocturno. Estos errores son registrados por aplicativo e informados al analista responsable. Personal de producción hace el seguimiento y control hasta que se solucione por completo el problema

- Procesos On-line

Se realizan estadísticas por cualquier cancelación (abend) en el ambiente interactivo (CICS) que causan contenciones en el sistema, caídas o dejan inoperativa determinada parte del sistema. Estos errores son monitoreados y registrados por el área de Ingeniería de Sistemas e informados al analista responsable para su solución. Al igual que en el proceso batch, se hace seguimiento hasta que el problema se solucione por completo.

#### En ambiente de Aceptación usuaria

- Estadísticas de porcentaje de avance de pruebas

Se registra el porcentaje de avance de las pruebas tomando como unidad de medida el caso de prueba y se reporta el retraso o adelanto de acuerdo al cronograma de pruebas.

- Estadísticas de errores en las pruebas

Se registran los errores encontrados durante la ejecución de las pruebas, y son clasificados en diferentes niveles de acuerdo a la severidad del problema, La escala de niveles de gravedad está en función del impacto de la aparición de ese error durante la operación del sistema, independientemente de la dificultad de su corrección o de las consecuencias sobre el diseño.

<b>Nivel</b>	<b>Severidad</b>	<b>Tipo error</b>	<b>Descripción</b>
A	Interrupción	Invalidantes	Un elemento significativo de la prueba está totalmente bloqueado y no se puede seguir con la prueba. Imposibilidad de operar funciones básicas.
B	Alta	Graves	Un elemento significativo de la prueba está bloqueado, pero se puede seguir con el resto de los elementos con algunas restricciones.
C	Media	Comunes	La prueba no está bloqueada, pero hay ciertas restricciones para terminar con uno o más casos de prueba (Ej. Reporte mal ordenado)
D	Baja	Leves, detalles	La prueba puede continuar, pero se requiere la corrección de la falla (Ej. Error cosmético en pantalla o reporte)

También se registra el estado y el tiempo que se ha utilizado en resolverlos

Un error puede tener los siguientes estados:

<b>Estado</b>	<b>Descripción</b>
Pendiente	El error ha sido identificado en pruebas y está en proceso de revisión y corrección.
En Prueba	El error ha sido corregido y está disponible para pruebas. El grupo de pruebas realiza la prueba y si tiene éxito pasa a estado Atendido, de lo contrario pasa a estado Verificado con error
Verificado con Error	La corrección del error no ha sido exitosa y el error persiste
Atendido	La corrección del error ha sido probada exitosamente y aceptada por el grupo de pruebas

A partir de este registro (ver formato en anexo VII) se obtienen las estadísticas de Distribución de problemas por criticidad y Problemas Pendientes por antigüedad, que se muestra en el Anexo IX.

### **3.5.6. MEJORAS EN EL PROCESO DE PRUEBAS**

Dentro de la División de Sistemas se sigue un proceso de pruebas exhaustivo, se cuenta con metodología, estándares, documentación para proyectos, etc. Sin embargo, al no existir un área de Calidad, los controles son realizados por el equipo de trabajo del proyecto y hay pocas mediciones (no son formales o se realizan para algún proyecto específico).

Dentro de las estrategias a implementar para lograr un mejoramiento continuo de la calidad en el desarrollo de software y en la etapa de pruebas, se plantean lo siguiente:

1. Establecimiento de un área de Aseguramiento de la Calidad, como entidad que provea:
  - Consultoría y soporte a las funciones con rol de defensor y apoyo, no fiscalizador
  - Capacitación y apoyo en procedimientos, estándares y documentación
  - Asistencia y participación en las revisiones técnicas formales
  - Implementación de Listas de chequeo, tanto para el Plan de pruebas y el Ciclo de vida de Desarrollo del software.
2. Establecimiento formal de Métricas

Las métricas proporcionarán una vista objetiva del avance del proceso de pruebas . Los objetivos para la obtención y reporte de métricas de prueba son:

- Proporcionar métricas operacionales para la administración táctica del proceso de pruebas
- Proporcionar métricas de tendencias que permitan a la administración realizar decisiones mejor informadas acerca de asignación de recursos y tiempos para las actividades de pruebas futuras.

Las métricas se calcularán periódicamente para registrar la situación de avance de las pruebas y poder estimar el esfuerzo restante. La unidad de definición de métricas es el caso de pruebas, y para cada caso de prueba se debe definir una estimación de tiempo y luego registrar los tiempos reales a medida que se ejecutan. Para cada caso de prueba hay que registrar las horas reales utilizadas y hacer una estimación del tiempo restante.

Con el informe de avance se actualizarán los tiempos reales de las pruebas ya ejecutadas y se corregirán las estimaciones informadas para las pruebas no ejecutadas aún.

En el anexo X se muestran en detalle las métricas planteadas para su implementación

Adicionalmente, y aunque no se está detallando, es importante considerar las métricas de Productividad, como por ejemplo:

- Requerimientos atendidos, por proyecto, por tipo de requerimiento y por analista responsable.
  - Requerimientos pendientes por proyecto, por tipo de requerimiento y por tiempo de antigüedad
  - Errores por proyecto y por analista responsable
3. Documentación de las modificaciones por programa, para poder tener un control del número de cambios, tipo de cambios, errores, etc.
  4. Implementación de herramienta de control de cambios

Las necesidades informáticas actuales hacen que la Gestión de cambios sea una plataforma que controle el Ciclo de Vida de la modificación y que su implantación en la instalación se pueda hacer de manera rápida, progresiva y no traumática. Dicha gestión debe estar enfocada al cambio y no al componente, ya que un cambio puede contener múltiples y diferentes componentes los cuales están íntimamente relacionados y es imprescindible que sean vistos como una única entidad de cambio a lo largo del ciclo de vida de la modificación.

Changeman es un producto que proporciona un control total sobre todos los componentes lógicos que integran los procesos de cambio de cualquier aplicación, así como el paso de dichos componentes entre los ambientes definidos en las direcciones de Sistemas de Información.

El banco adquirió esta herramienta, después de ver su funcionamiento en grandes bancos mexicanos; sin embargo nunca se llegó a implementar.

Entre los principales valores que aporta esta herramienta, se pueden destacar:

- Es un gestor de tercera generación, enfocado a procesos de trabajo y multiplataforma
- Trabaja con expedientes de cambio
- Rapidez y facilidad para su implementación
- Automatización de procesos
- Arquitectura y seguridad
- Control e implementación de desarrollos concurrentes
- Eliminación de papel en solicitud y aprobación
- Garantía absoluta de los fuentes y ejecutables
- Control completo sobre la información histórica y en vuelo, además del estado de los expedientes de cambio (Amplio número de informes)
- Automatización del ciclo de vida de las modificaciones



## **CAPITULO IV**

### **EVALUACIÓN DE RESULTADOS**

A continuación se presentan los beneficios obtenidos con la implementación de la metodología de pruebas para el proceso de desarrollo de Software.

1. Reducción del tiempo y costo de desarrollo de software, por:
  - Reducción en las discrepancias entre especificaciones funcionales y diseño técnico
  - Reducción de errores de codificación por mejoramiento de los planes de prueba unitarias y de sistemas
  - Mayor control de recursos utilizados planificando y haciendo seguimiento de cada proyecto

Aunque no se cuenta con datos cuantificables, una experiencia concreta es la Fusión que se trabajó por la adquisición del Banco Latino, la cual se realizó en solo 3 meses.

2. Aumento de la calidad de los productos de software implementados (Disminución de errores y/o caídas de los sistemas), por:
  - Los cambios y mejoras de los sistemas reflejan en forma más precisa los requerimientos del negocio.

- Se trabaja el inventario de requerimientos y priorizaciones por aplicativo y si es necesario la implementación por etapas.
  - Los pases a Producción son más seguros, luego del proceso de pruebas. Los usuarios están informados de cualquier cambio, incluso los cambios técnicos.
3. Se ha logrado una cultura organizacional en lo que respecta a Desarrollo de software y pruebas, lo cual permite lograr un aseguramiento de la calidad. Esto debido a:
- Énfasis en el uso de la metodología de desarrollo de software. Están claramente definidos los procesos para cada tipo de desarrollo (proyecto, mantenimiento, crisis y departamentales).
  - Equipos de trabajo multidisciplinario. El equipo es conformado por las áreas de: Negocios, Desarrollo Organizacional, Sistemas y Operaciones.
  - Definición de los roles de cada una de las áreas que conforman el equipo y compromiso con sus funciones
4. Se ha logrado obtener un grado de madurez en el desarrollo de software, principalmente en la etapa de pruebas, ya que se cuenta con procesos definidos, documentados y continuamente mejorados que permiten definir los tiempos de implementación de manera más exacta y garantizar la calidad del producto. El cambio se realizó con la implementación del software integrado hacia 1997 y se consolidó con el Proyecto 2000 y la fusión con el Banco Latino.
5. Finalmente se puede mencionar algunas experiencias, en relación a los productos de Interbank y su metodología de desarrollo de software:

- En las diversas visitas a bancos mexicanos y chilenos por el problema del Año 2000, y diversas opiniones de consultores extranjeros, mencionaban que ningún banco en el mundo realizaba un pase a Producción de todo el software bancario, pues era un proceso de mucho riesgo. Sin embargo gracias a los procesos de prueba con que cuenta Interbank, se pudo realizar un cambio de todo el software en dos oportunidades: La implementación del software Systematics y el proyecto Año 2000.
- El proyecto de Fusión con el Banco Latino, en la cual se migraron todos los sistemas del banco Latino a los sistemas de Interbank, se realizó en solo 3 meses
- Interbank ha logrado el reconocimiento por los productos y servicios novedosos que ha puesto a disposición de los clientes, en cuyos desarrollos el área de sistemas ha tenido una activa participación.

El año 2001 se obtuvo el premio "Creatividad Empresarial" en la categoría Servicios Bancarios, por la Tarjeta Virtual Netactiva 24, la primera y única tarjeta virtual segura de compras por Internet.

- Se han recibido reconocimiento por parte de los clientes-empresa por determinados servicios, los cuales son comparados y preferidos a la competencia. Esto se puede medir por el incremento de clientes y las estadísticas del servicio.

## **CAPITULO V**

### **CONCLUSIONES Y RECOMENDACIONES**

1. La situación actual en el mercado de software requiere implementar un sistema que asegure la calidad de los productos. Es necesario como empresa poder satisfacer al cliente, tener un lugar en la competencia y minimizar los defectos del software. Las empresas deben preocuparse por brindar un producto de calidad y un proceso que sirva como guía durante la creación del software.
2. Es importante implementar un área de Aseguramiento de la Calidad que se encargue de velar por el cumplimiento de la metodología de desarrollo, revisiones técnicas, estrategias de pruebas, control de la documentación, y seguimiento y actualización de estándares y procedimientos. Asimismo es fundamental considerar mediciones y estadísticas para valorar la calidad de los sistemas que se construyen.
3. Es necesario la gerencia participativa para poder implementar productos de calidad. Las gerencias deben estar convencidas de la importancia de la calidad del software. Nuestra experiencia nos demuestra que el equipo de desarrollo de software sufre continuamente presiones para cumplir fechas que a menudo son determinadas más por las necesidades del negocio que por consideraciones técnicas de factibilidad de cumplimiento. Es aquí donde se empiezan a sacrificar

actividades: documentación, capacitación, revisiones y hasta pruebas, produciéndose muchas veces sistemas que funcionan bien, pero sin ser necesariamente de calidad (software difícil de mantener, desordenado, no es flexible, ineficiente, etc.) y muchas veces sólo se aplica la metodología al desarrollo (proyectos grandes) dejando de lado el mantenimiento.

4. Es importante mencionar que la Etapa de pruebas es una de las actividades principales del desarrollo de software y no es una actividad que se realiza en secuencia después de la programación. La Prueba es una actividad que está presente a lo largo de todo el ciclo de vida del desarrollo de software. Cada fase del ciclo de vida se enfoca en diferentes aspectos de prueba desde la etapa de análisis hasta la implementación en Producción. Es por eso de mucha importancia que exista una Cultura de Pruebas que sea extensiva en toda la organización y no sólo en el área de Sistemas.
5. Finalmente cabe mencionar que la Metodología de Pruebas del banco es una herramienta que puede ser implementada en el área de desarrollo de software de cualquier organización, tanto de empresas financieras como no financieras. De esta forma, logrando la cultura de pruebas en la organización y haciéndola parte del trabajo diario, se puede decir que la calidad no cuesta.

## BIBLIOGRAFÍA

1. Ingeniería de Software

Roger S. Pressman

Cuarta Edición, editorial McGraw-Hill España 1998

2. Solución Testing Año 2000

Propuesta de Solución de Empresa Selesta con uso de herramientas VIASOFT

Año 1998

3. Application Testing in the Real World

Propuesta de Solución con uso de herramientas COMPUWARE

Año 1998

4. Documentación Proyecto Systematics

Manuales para implementación en Interbank

Año 1996

5. Documentación de Bancos Extranjeros

Banco Santander - Chile

Banco de A.Edwards – Chile

Banco Crédito de Inversiones - Chile

Banco del Estado de Chile

Banco Bilbao Vizcaya BBV - México

Banco Santander – México

Bancomer - México

Setiembre 1998

## GLOSARIO DE TERMINOS

Término	Descripción
<i>Aplicativo</i>	Denominado también <i>sistema de información</i> . Se conforma por un conjunto de programas y cada programa realiza una determinada acción o función.
<i>Sistema de información</i>	Ver aplicativo
<i>On-line</i>	Se denomina así a todo intercambio de información que se efectúa en forma directa desde una determinada plataforma con el <i>Host</i> .
<i>Batch</i>	Se denomina así al procesamiento de datos por lote.
<i>Host</i>	Computador Central o <i>Mainframe</i> .
<i>Mainframe</i>	Computador Central o <i>Host</i> .
<i>Input</i>	Ingreso de datos que se realiza a través de un medio informático o plataforma. Insumo importante para todo procesamiento de datos. Todo <i>input</i> puede ser proporcionado por medios magnéticos como un <i>diskette</i> que contiene información previamente almacenada o por digitación de datos a través de un teclado de computadora.
<i>Output</i>	Salida de información, generalmente resultado del procesamiento de un aplicativo, que se realiza a través de vistas en pantalla o reportes impresos.
<i>PC, PC's</i>	Abreviatura de Personal Computer (Computador Personal).
<i>Subsistema</i>	La terminología utilizada en el Banco para referirse a un conjunto relacionado de aplicaciones.
<i>Prueba</i>	Un proceso por el cual la operación de un programa o grupo de programas es validado ejecutándolo(s) con un completo y adecuado conjunto de entradas y un conjunto de resultados esperados verificado.
<i>Plan de pruebas</i>	Una descripción de cómo están planificadas las actividades de prueba para una partición particular o grupo de particiones serán organizadas y ejecutadas.
<i>Caso de prueba</i>	Un detalle del conjunto de objetivos, datos, procedimientos, resultados esperados necesarios para validar un elemento funcional específico o característica de un subsistema.
<i>Test Scripts</i>	Un conjunto de pasos que deben ser seguidos en secuencia para ejecutar un caso de prueba particular.
<i>Abend</i>	Una falla en un programa on-line, que origina que la transacción que lo está usando paralice su operación.
<i>JCL's</i>	Es un lenguaje de control de trabajos que permite la ejecución de programas batch.
<i>Debuggers</i>	Son programas que permiten hacer seguimiento a los errores para suprimir fallos. Realizan el seguimiento instrucción por instrucción.
<i>Infopac</i>	Herramienta que permite registrar por usuario los diferentes reportes que serán distribuidos después de finalizado el proceso <i>batch</i> de los aplicativos. La distribución se realiza de manera automática a toda la red de oficinas de provincias y oficinas de Lima y Callao. Los resultados pueden ser visualizados e impresos de acuerdo a la necesidad de cada usuario.
<i>Scheduler</i>	Herramienta que permite registrar la secuencia requerida por los aplicativos para la ejecución del procesamiento <i>batch</i> en el <i>Host</i> .

## **ANEXOS**

Se mostrarán los formatos utilizados por el banco para registrar el Plan de pruebas, así como los resultados y el seguimiento de las pruebas.

En los formatos se está incluyendo un ejemplo para una determinada aplicación.

1. Anexo I : Responsabilidades generales del Equipo de trabajo
2. Anexo II : Componentes de la Solución
3. Anexo III : Ejemplo de definición de eventos del Negocio
4. Anexo IV : Formato para registrar los eventos de negocio
5. Anexo V : Formato para registrar el inventario de casos de prueba por evento de negocio
6. Anexo VI : Formato para registrar los test scripts por caso de prueba
7. Anexo VII : Formato para registrar el log de problemas durante las pruebas
8. Anexo VIII : Formato para registrar el Informe del proceso batch de Aceptación
9. Anexo IX : Estadísticas problemas durante las pruebas.
10. Anexo X : Métricas



## ANEXO I

### RESPONSABILIDADES GENERALES DEL EQUIPO DE TRABAJO

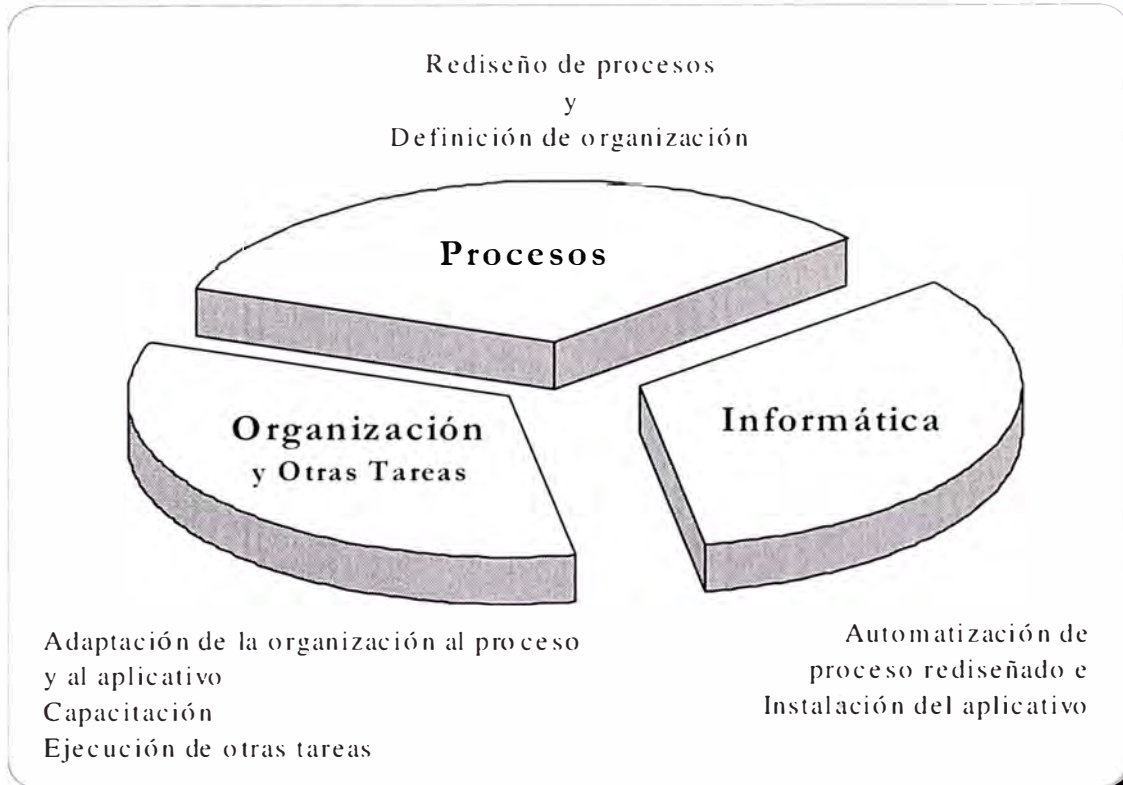
Se presentan en el siguiente cuadro y están referidas al desarrollo de software aplicativo, tanto proyectos como requerimientos menores

<b>USUARIO RESPONSABLE</b>	
<ul style="list-style-type: none"><li>◆ Prioriza sus requerimientos tecnológicos, aquellos que derivan en proyectos y aquellos que son requerimientos menores.</li><li>◆ Define la funcionalidad del aplicativo (<i>el qué</i>).</li><li>◆ Define los datos que ingresan al aplicativo y los resultados que debe producir.</li><li>◆ Aprueba el pase a producción del aplicativo.</li><li>◆ Centraliza y filtra los requerimientos de <i>usuarios finales</i> (único punto de contacto).</li><li>◆ Sustenta los beneficios de su proyecto ante el Banco.</li><li>◆ Asume los costos del desarrollo del aplicativo, así como, sus costos de explotación.</li></ul>	
<b>ADMINISTRADOR DE PRODUCTO</b>	<b>ANALISTA FUNCIONAL</b>
<ul style="list-style-type: none"><li>◆ Define la forma del aplicativo en detalle (<i>el cómo</i>).</li><li>◆ Define y ejecuta las pruebas funcionales del aplicativo.</li><li>◆ Elabora y mantiene el <i>Manual de Usuario</i> del aplicativo.</li><li>◆ Elabora y mantiene el <i>Manual del Producto y/o Servicio</i>, con el apoyo del <i>Analista Funcional</i></li><li>◆ Capacita respecto al aplicativo y al flujo operativo a nivel nacional.</li><li>◆ Implementa el aplicativo y el flujo operativo a nivel nacional.</li><li>◆ Otorga los accesos al aplicativo a <i>usuarios finales</i>.</li><li>◆ Asesora al <i>Usuario Responsable</i> en el filtro de requerimientos de sus <i>usuarios finales</i>.</li><li>◆ Durante la explotación del aplicativo, realiza los cuadros de la información.</li></ul>	<ul style="list-style-type: none"><li>◆ Diseña o rediseña el proceso del cual se deriva el aplicativo.</li><li>◆ Define los aspectos organizacionales y los recursos no tecnológicos necesarios para integrar el proceso y el aplicativo a la organización del Banco.</li><li>◆ Apoya al <i>Administrador de Producto</i> en la elaboración y mantenimiento del <i>Manual del Producto y/o Servicio</i>.</li><li>◆ Apoya al <i>Administrador de Producto</i> en la implementación del aplicativo y del flujo operativo a nivel nacional, así como en la capacitación respectiva.</li></ul>
<b>OTRAS AREAS DEL BANCO</b>	<b>ANALISTA DE SISTEMAS</b>
<ul style="list-style-type: none"><li>◆ Apoyan al <i>Usuario Responsable</i> en las diferentes fases del desarrollo de un aplicativo, según la naturaleza de éste.</li><li>◆ El área de <i>Auditoría Interna</i> verifica la existencia de controles internos en los procedimientos operativos y en el aplicativo.</li></ul>	<ul style="list-style-type: none"><li>◆ Analiza la factibilidad técnica del proyecto o requerimiento menor.</li><li>◆ Produce o adquiere el aplicativo (previa investigación en el mercado), y le da mantenimiento.</li><li>◆ Realiza :<ul style="list-style-type: none"><li>◇ las pruebas técnicas del aplicativo,</li><li>◇ los manuales técnicos, y</li><li>◇ la capacitación técnica.</li></ul></li><li>◆ Instala el software aplicativo a nivel nacional, incluyendo hardware y telecomunicaciones.</li></ul>

## ANEXO II

### COMPONENTES DE LA SOLUCIÓN

Todo requerimiento de soporte tecnológico es analizado de manera integral y sinérgica por el *Equipo de Trabajo*. Los componentes de su solución son



- **Solución de Procesos**  
Diagnostica el flujo de procesos inicial (considerando el valor agregado y controles de cada proceso) y su organización (áreas y cargos involucrados), identificando los aspectos a ser rediseñados. Define el nuevo flujo de procesos y la estructura organizacional necesaria que lo llevará a cabo.
  
- **Solución de Organización y otras tareas**  
Implementa la organización necesaria para llevar a cabo el flujo de procesos rediseñado y que integra el uso del aplicativo. Capacita a nivel nacional, respecto a los nuevos procedimientos a seguir y sobre el funcionamiento del aplicativo. Asimismo, especifica, adquiere o prepara, e instala los recursos no tecnológicos necesarios (fuerza laboral adicional, capacitación, mobiliario, infraestructura, publicidad, etc.).
  
- **Solución Informática**  
Define la tecnología necesaria para automatizar el proceso: especifica las características del aplicativo; define los aspectos que lo integrarán con otros aplicativos y con la plataforma tecnológica del Banco; desarrolla, compra o modifica algún aplicativo existente y lo instala a nivel nacional.

## ANEXO III

### EJEMPLO DE DEFINICIÓN DE EVENTOS DE NEGOCIO

A continuación se muestra, un ejemplo para definir un evento del negocio para el producto bancario Cuenta Millonaria.

Uno de sus eventos de Negocio para este producto es la Apertura de la cuenta Millonaria. Su inicio está definido por la operación de un representante financiero a través de la transacción del banco llamada 'Apertura Rápida'. A partir de ese momento se realiza el depósito inicial a través de una transacción financiera, se refleja en los totales de caja la operación y se pueden realizar las consultas en los sistemas de 'Clientes' y 'Cuentas de Ahorros' del registro correspondiente. Asimismo se debe efectuar la actividad de complementación de los datos del cliente en el sistema de 'Clientes'. Su término está definido por los reportes contables tanto de caja como de la aplicación y las consultas en el aplicativo de 'Cuentas de Ahorros' una vez efectuado el proceso batch.

Para apoyar este evento de negocio, existen varios sistemas o funcionalidades de éstos que están directa o indirectamente relacionados con este evento de Negocio, como son:

- Apertura Rápida
- Transacción financiera de depósitos
- Consultas en transacciones financieras y administrativas el mismo día
- Transacciones de extorno financiero
- Registro de datos en aplicación de Clientes
- Reportes contables
- Consultas administrativas después del proceso batch
- Transacciones de extorno administrativo

El banco ha decidido que la certificación de sus aplicaciones las realizará de acuerdo a cada uno de los eventos de negocio que se generan a partir del apoyo a las funciones operativas del banco. En este sentido se realizó un levantamiento de las funciones apoyadas por cada una de las aplicaciones.

## Anexo IV EVENTOS DEL NEGOCIO

### CERTIFICACION PROYECTO COMERCIO EXTERIOR

<b>APLICACION</b>	Nombre del Aplicativo
<b>GRUPO DE TRABAJO</b>	Nombre Usuario Responsable
<b>FASE</b>	Certificación UAT

N°	Nombre del Evento	Criticidad	Descripción	Aplicaciones Relacionadas
BKT_001	Ingreso de Carta de Crédito Exportación	1	Permite el registro en sistema de todos los datos solicitados en el crédito como beneficiario, importe, producto, vcto, cobro de gastos de apertura.	CICS, POSICION CLIENTE , INFOPAC, TARIFARIO, SYSTEMATICS
BKT_002	Consulta de Carta de crédito Exportación	2	Verificación de los datos ingresados en la apertura	CICS
BKT_003	Enmiendas de Carta de Crédito Exportación	1	Permite efectuar cualquier enmienda indicada por el banco aperturador del crédito	CICS - IMPACS
BKT_004	Ajuste de Carta de Crédito Exportación	2	Se utiliza para modificaciones de orden interno	CICS
BKT_005	Comments de Carta de Crédito	2	Se utiliza para saber la situación del documento (manejo interno del banco).	CICS
BKT_006	Pagos de Carta de Crédito.	1	Ingreso del importe que se ha negociado así como las fechas de reembolso y proceso por sistemas que permite el cobro de gastos por negociación, trámite, portes,courier, cheques,swift/tlx y abono neto según instrucciones.	CICS / IMPACS / INFOPAC / SAFE
BKT_007	Vouchers contables de C. de Crédito.	2	Permite efectuar ajustes contables o creditos/debitos pendientes	IMPACS / CICS
BKT_008	Movimiento Contable de C.Credito	2	Consulta del documento por saldo, comisiones y movimiento contable	CICS
BKT_009	Ingreso Cobranza de Exportación	1	Permite el ingreso de la Cobranza con los datos requeridos como girado, girador,importe,producto,vencimiento,forma de pago,banquero.	CICS, POSICION CLIENTE , INFOPAC, TARIFARIO, SYSTEMATICS
BKT_010	Consulta de Cobranza.	2	Verificación de los datos ingresados en el sistema.	CICS
BKT_011	Ajuste de Cobranza	2	Se utiliza para modificaciones de orden interno.	CICS
BKT_012	Comments de Cobranza	2	Se utiliza para saber la situación del documento (manejo interno del banco).	CICS
BKT_013	Pago de Cobranza.	1	Ingreso del importe que se ha negociado así como las fechas de reembolso y proceso por sistemas que permite el cobro de gastos por negociación, trámite, portes,courier, cheques,swift/tlx y abono neto según instrucciones.	CICS / IMPACS / INFOPAC / SAFE
BKT_014	Vouchers Contables de Cobranza.	2	Permite efectuar ajustes contables o creditos/debitos pendientes	CICS / IMPACS
BKT_015	Movimiento Contable de Cobranza	2	Consulta del documento por saldo, comisiones y movimiento contable	CICS
BKT_016	Ingreso Cheques en Cobranza.	1	Permite el ingreso del cheque con los datos requeridos como girador,importe,moneda,banco.	CICS, POSICION CLIENTE , INFOPAC, TARIFARIO, SYSTEMATICS
BKT_017	Consulta Cheque en Cobranza	2	Verificación de los datos ingresados en el sistema.	CICS

**Anexo V**  
**INVENTARIO DE CASOS DE PRUEBA POR EVENTO**

test script	criticidad	descripción	evento	descripción evento
BKT_001_001	1	Acción de prueba: ingreso carta de credito vista avisada con cta cte moneda nacional	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_016	1	Acción de prueba: ingreso carta de credito plazo diferido, confirmada , aladi sin cta cte moneda nacional	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_022	1	Acción de prueba: ingreso carta de credito transferible vista, confirmada, sin cta cte moneda nacional	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_029	1	Acción de prueba: ingreso carta de credito transferible vista, avisada, aladi con cta cte moneda nacional	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_035	1	Acción de prueba: ingreso carta de credito transferible plazo diferido, confirmada, aladi con cta cte moneda nacional	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_038	1	Acción de prueba: ingreso carta de credito a plazo avisada para enviar a un banco local	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_039	1	Acción de prueba: ingreso carta de credito vista confirmada para enviar a un banco local	BKT_001	Ingreso de Carta de Crédito Expo
BKT_001_041	1	Acción de prueba: ingreso carta de credito vista avisada con cta cte moneda nacional en otra moneda diferente de dolares	BKT_001	Ingreso de Carta de Crédito Expo
BKT_002_001	2	Acción de prueba: consulta carta de credito vista, avisada con cuenta corriente	BKT_002	Consulta de Carta de crédito Expo
BKT_002_016	2	Acción de prueba: consulta carta de credito plazo diferido, confirmada, aladi sin cuenta corriente	BKT_002	Consulta de Carta de crédito Expo
BKT_002_022	2	Acción de prueba: consulta carta de credito transferible vista, confirmada, sin cuenta corriente	BKT_002	Consulta de Carta de crédito Expo
BKT_002_029	2	Acción de prueba: consulta carta de credito transferible vista, avisada, aladi, con cuenta corriente	BKT_002	Consulta de Carta de crédito Expo
BKT_002_038	2	Acción de prueba: consulta de carta de credito a plazo avisada que fue enviada a un banco local	BKT_002	Consulta de Carta de crédito Expo
BKT_002_039	2	Acción de prueba: consulta de carta de credito a la vista confirmada que fue enviada a un banco local	BKT_002	Consulta de Carta de crédito Expo
BKT_003_006	1	Acción de prueba: enmienda de carta de credito avisada, vista, aladi sin cta cte	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_003_013	1	Acción de prueba: enmienda de carta de credito avisada, plazo diferido, aladi con cta cte	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_003_030	1	Acción de prueba: enmienda de carta de credito transferible, avisada, vista, aladi sin cta cte	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_003_036	1	Acción de prueba: enmienda de carta de credito transferible, confirmada, plazo diferido, aladi sin cta cte	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_003_038	1	Acción de prueba: enmienda de carta de credito a plazo avisada que fue enviada a un banco local	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_003_039	1	Acción de prueba: enmienda de carta de credito a plazo avisada que fue enviada a un banco local	BKT_003	Enmiendas de Carta de Crédito Expo
BKT_004_002	2	Acción de prueba: ajuste de carta de credito confirmada, vista con/sin cta cte	BKT_004	Ajuste de Carta de Crédito Expo
BKT_004_005	2	Acción de prueba: ajuste de carta de credito avisada, plazo diferido con/sin cta cte	BKT_004	Ajuste de Carta de Crédito Expo
BKT_004_016	2	Acción de prueba: ajuste de carta de credito transferible, avisada, plazo diferido, aladi con/sin cta cte	BKT_004	Ajuste de Carta de Crédito Expo
BKT_004_017	2	Acción de prueba: ajuste de carta de credito transferible, confirmada, vista, aladi con/sin cta cte	BKT_004	Ajuste de Carta de Crédito Expo
BKT_004_020	2	Acción de prueba: ajuste de carta de credito a la vista y a plazo confirmada que fue enviada a un banco local.	BKT_004	Ajuste de Carta de Crédito Expo

## Anexo VI

**APLICATIVO:** BANKTRADE

**TEST SCRIPT:** BKT\_001\_001

**EVEN TO:** INGRESO CARTA DE CREDITO

---

**ACCION DE PRUEBA:** INGRESO CARTA DE CREDITO VISTA AVISADA CON CTA CTE MN

**DATOS DE PRUEBA:** 1. VERIFICAR CTA CTE DEL CLIENTE.  
2. INGRESO DE INFORMACION DEL CREDITO; BANQUERO, IMPORTE, BENEFICIARIO, PRODUCTO, PAIS, REEMBOLSO ETC.

**RESULTADOS ESPERADOS:** -INGRESO DEL CREDITO EN EL CICS DE BANKTRADE CON LOS DATOS DE LA CARTA DE CREDITO RECIBIDA.  
-GENERACION DEL NUMERO CORRELATIVO  
-GENERACION DE REPORTES  
-GENERACION DE TRACERS  
-CONSULTA EN EL CICS DE BANKTRADE  
-INGRESO CONTABLE DEL CREDITO  
-EMISION SWIFT MT730

**VERIFICACION DE RESULTADOS :**

- Aplicativos: CICS-POS. DECLIENTE- INFOPACS-TARIFARIO-SYSTEMATICS.
- Pantallas : CUENTAS CORRIENTES-CONSULTA CICS
- Reportes : POR DEFINIR
- Refrendos : TRACER, NOTAS DEBITO
- Otros :

**NOTA:** Adjuntar impresión de pantallas, anotar mensajes, códigos de error y toda documentación que se obtenga de la prueba.

**PRUEBA REALIZADA POR :**

<b>NOMBRE(S):</b>	
-------------------	--

<b>FECHA:</b>		<b>HORA:</b>	
---------------	--	--------------	--

<b>OBSERVACIONES:</b>	
-----------------------	--

<b>FIRMA(S):</b>
------------------

---

---

## Anexo VII LOG DE PROBLEMAS

Consolidado de Problemas pendientes de las Pruebas Pagoactivo UAT al 28-2-2002

n°	aplicativo	criticidad	reportado		descripción	derivado	status	días	fecha
			por	fecha		a ...		pend	solución
1	RNC	3	Usuario1	01/11/2000	Pago de celular (deuda en dólares) pagada con ctacte. soles que no tiene saldo envía mensaje: "No lo podemos atender ¿Desea realizar otre operación?" en lugar de enviar un mensaje que indique: Fondos insuficiente o cuenta sin saldo	Analista1	OK	2	03/11/2000
2	PAG	1	Usuario1	01/11/2000	No se permite el pago de Mensatel. Motivo: Ajuste por que el cajero no permite ingresar letras ('K')	Analista2	OK	2	03/11/2000
3	RNC	1	Usuario1	01/11/2000	No permite el pago de SEDAPAL (empresa sin data). Motivo: Cuenta viaja en ceros	Analista1	OK	2	03/11/2000
4	PAG	3	Usuario2	01/11/2000	La glosa se trunca en 11 caracteres cuando se realizan pagos en la misma moneda. Para CV sale 'Retiro ATM CV'	Analista2	OK	2	03/11/2000
5	\$IM	3	Usuario2	01/11/2000	En la consulta de movimientos las glosas son variadas en algunos casos dice DB C/V U\$ ATM TELEF(Servicio en Dólares -CtaCte en Soles), RET C/V(Serv S/. -CtaCte en dólares), RET ELE TELEF BAS (Sevic S/. -Cta Cte Soles), TELEF BAS AT(Serv. S/. Cta Aho S/.)	Analista2	OK	2	03/11/2000
6	PAG	1	Usuario2	01/11/2000	Cuando realizo el pago de servicios sin data sale el mensaje: Ingrese el monto a pagar en dólares cuando la deuda es en soles	Analista1	OK	2	03/11/2000
7	PAG	1	Usuario2	01/11/2000	Cuando se paga Sedapal (actualmente inscrita como empresa inactiva en UAT) no me debería dejar pagar, sin embargo me envía el mensaje: Ingrese los 8 dígitos de su N.servicio (Si estuviera activa el msg. sería: Ingrese los 11 dígitos de su ref.de cobro'	Analista2	OK	2	03/11/2000
8	PAG	2	Usuario2	01/11/2000	Para pagos sin data o con maestra (Ejem. Pagoactivo, varios) sale el mensaje : Ingrese el código de la empresa del rubro a pagar. Se debe revisar dicho mensaje. Asimismo muestra una ventanita para ingresar 7 dígitos debiendo ser solo 5	Analista1	OK	3	04/11/2000
9	PAG	2	Usuario1	01/11/2000	Cuando se selecciona 1.Pago de Servicio, 2.Celular, aparecen las empresas para ese servicio y un Mensaje no descriptivo: "Seleccione la empresa del servicio que deseas pagar"	Usuario	OK	3	04/11/2000
10	RNC	2	Usuario1	01/11/2000	No se imprime el saldo disponible	Analista1	OK	2	03/11/2000
11	RNC	3	Usuario1	01/11/2000	En las pantallas de cajeros falta acento en Telefónica (considerando que Telefonía si tiene)	Analista1	OK	3	04/11/2000
12	PAG	3	Usuario1	01/11/2000	Revisar las variables de códigos en administrativo (Obs. de forma)	Usuario	OK	3	04/11/2000
13	PAG	3	Usuario1	01/11/2000	Cambiar mensaje 'Ingrese xx dígitos...' por 'Digite xx dígitos....'	Analista2	OK	3	04/11/2000
14	PAG	2	Usuario3	01/11/2000	Al realizar un pago de teléfono fijo o celular Telefónica el mensaje dice ingrese 8 dígitos y debe decir "Ingrese 7 dígitos para lima y 8 para provincias como actualmente esta en producción"	Analista2	OK	3	04/11/2000
15	PAG	2	Usuario3	01/11/2000	Cuando ingreso los 7 dígitos de celular Telefónica para realizar el pago me envía un mensaje genérico por lo que pide el ingreso de un 1 al inicio para completar los 8 dígitos	Analista2	OK	3	04/11/2000
16	PAG	3	Usuario3	01/11/2000	En el refrendo para Teléfono fijo Lima se está anteponiendo el dígito '0' (8 dígitos en lugar de 7)	Analista1	OK	1	02/11/2000
17	PAG	3	Usuario3	01/11/2000	debería aparecer un campo de ingreso de datos centrado.	Analista2	OK	1	02/11/2000
18	RNC	3	Usuario3	01/11/2000	En la opción pagoactivo cuando pide el código de deudor: 'Ingrese los xx dígitos de su cuenta' no se muestra el fondo oscuro (zona de digitacion) para el ingreso de datos	Analista1	OK	3	04/11/2000

**Anexo VIII**  
**INFORME DE PROCESO BATCH DE ACEPTACIÓN**

N°	FECHA	HORA	SCHDID	JOBID	PASO	RC	JN	DIAGNOSTICO	SOLUCION	SUP
1	11/01/2001	18:09:58	ALSG101	LSGA010H	S02PCD01	00209	1783	Codigo: 209. Tiempo de respuesta agotado	Se verifica copia a HOST y se da CP	HGL
2	11/01/2001	18:38:39	ASAT101	SATA012D	S06PFT01	00012	1980	SQLCODE : -0000000911	Se reenvia nivel del mismo paso	HGL
3	11/01/2001	19:39:29	\$DYNMVS	ICCA425H		00016	2235			HGL
4	11/01/2001	20:02:51	ASAT104	SATA133D	S10PE3IN	00016	2324	RUN ABORTED - ATTEMPTED TO PROCESS AN INCOMING FINANCIAL FILE WHICH HAS ALREADY BEEN PROCESSED.	Se da CP por indicacion del Analista responsable	HGL
5	11/01/2001	20:33:46	ATMF101	TMFA050D	S06IEBG	SA13	2259	13-	Se da CP para continuar	HGL
6	11/01/2001	21:34:04	APAG101	PAGA140D	S03P140	00016	2507	062-086,PAGA140D,S03P140 PAGP140,PAGFEXTE	Se reenvia desde inicio	HGL
7	11/01/2001	22:55:59	ATDA101	TDAA060D	S07PB210	00016	2758	ARCHIVO ACUMULADO-OLD SIN REGISTROS	Se da CP para continuar.	HGL
8	11/01/2001	23:08:47	ACJE102	CJEA075D	S02PB075	00016	2847	FECHAS DISTINTAS SE CANCELA PGM CJEB075	se comunico a Analista responsable quien indico solución y reenviar desde paso cancelado.	ARG
9	11/01/2001	23:10:23	ASAT101	SATA037D	infopac	U0666	2441	U0666	se recargaron los reportes y se dio post	ARG
11	12/01/2001	00:38:39	ASAT101	SATA039D	s24pft01	00012	3022	SQLCODE : -0000000805	se comunico al Analista responsable quien indico se de post	ARG
12	12/01/2001	01:43:50	A\$ST101	\$STA079D	S07PB301	00016	3081	ARCHIVO ABIERTO	Se cerró y repitió desde paso cancelado	ARG
13	12/01/2001	01:48:39	A\$ST101	\$STA079D	S07PB301	00016	3137	ARCHIVO EN USO	Se cerró y se dio RERUN	ARG
15	12/01/2001	04:27:40	A\$CL101	\$CLA050D	s15piebg	S222	3322	pide cartucho privado	Se da CP para continuar.	ARG
16	12/01/2001	05:37:46	ARNC102	RNCA100D	S15PFAID	00129	3341	NO LICENSE MANAGEMENT SUBSYSTEM IS ACTIVE	Se ejecutó por A12JES2	ARG
17	12/01/2001	06:49:43	ASAT102	SATA481M	S02UTIL	00012	3694	NO HAY ARCHIVO EN CARTUCHO	Se asignó área de disco y se dio RERUN	ARG
18	12/01/2001	07:15:22	ASAT102	SATA481M	S08PFT01	00012	3740	FECHAS DIFERENTES	Se comunicó a Analista responsable quien indico se deje pendiente	ARG
19	12/01/2001	07:45:43	ASAT102	SATA431M	S41B466R	00016	3762	ERROR EN OPEN ACUMMES = 39	se comunico a Analista responsable quien indico se reprocese cambiando longitud de acumulado	ARG

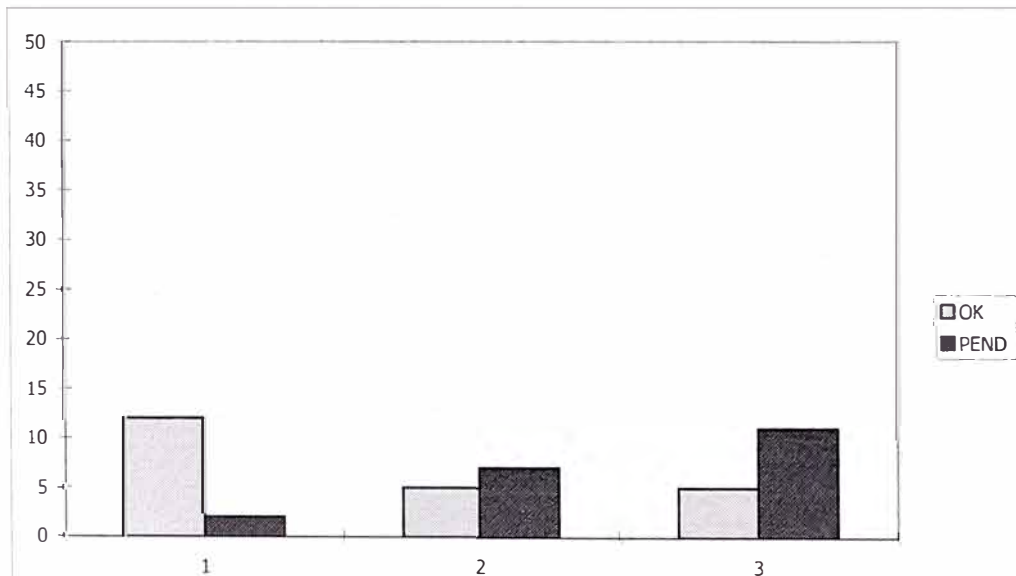


## Anexo IX ESTADISTICAS DE ERRORES DURANTE LAS PRUEBAS

Resultados al: 28/02/2002

Criticidad	OK	PEND	Total	%
1	12	2	14	33.3%
2	5	7	12	28.6%
3	5	11	16	38.1%
<b>Total</b>	<b>22</b>	<b>20</b>	<b>42</b>	

Duración	PEND	%
¡más de 1 semana!	20	100.0%
7 días	0	0.0%
6 días	0	0.0%
5 días	0	0.0%
4 días	0	0.0%
3 días	0	0.0%
2 días	0	0.0%
1 día	0	0.0%
reportados hoy	0	0.0%
	<b>20</b>	



## ANEXO X

### MÉTRICAS

#### I. MÉTRICAS OPERACIONALES

1. **Porcentaje Real de Completitud (PRC):** Esta métrica da una medida de la cantidad de pruebas planificadas completadas basadas en una estimación actual del esfuerzo restante. Se calcula de la siguiente manera:

$PRC = A / (A + B)$  donde

A = Horas reales incurridas

B = Estimación de las horas restantes para terminar la prueba

Ejemplo:

Total de horas incurridas en la prueba = 200

Estimación de las horas restantes para terminar prueba = 100

$$PRC = (200)/(200 + 100) = 67\%$$

2. **Factor de Variación de la Ejecución de las Pruebas (FVEP):** Esta métrica da una medida de cuan bien esta progresando la ejecución de las pruebas en relación a los planes. Un valor positivo indica un adelanto con respecto al plan y un valor negativo indica un retraso en relación a lo planificado.

$FVEP = A - B$  donde

A = PRC (Porcentaje Real de Completitud)

B = PPC (Porcentaje Planificado de Completitud)

Ejemplo:

A la fecha se ha incurrido en 200 horas en la prueba y se estima que se necesitan 150 horas mas para terminar.

Según el plan deberían haberse incurrido en 180 horas y deberían quedar 200 horas más para terminar.

$$PRC = (200)/(200 + 150) = 57\%$$

$$PPC = (180)/(180 + 200) = 47\%$$

$$FVEP = 57 - 47 = 10$$

En este ejemplo la ejecución de las pruebas está adelantada con respecto a lo planificado.

#### II. MÉTRICAS DE TENDENCIA

Se pretende que estas métricas ayuden a la planificación de pruebas futuras. El principal objetivo es obtener datos que permitan una distribución de tiempo y recursos efectiva en las pruebas siguientes.

1. **Defectos por programa**

Esta es una medida de los defectos identificados en las pruebas relativas al número de programas modificados. Esta métrica es medida para cada tipo de pruebas (on-line y batch) y proveerá información acerca de la tasa de errores que se puede esperar en las pruebas futuras.

DPC : Defectos por programa modificado.

NPC = número de programas modificados.

ND = Número de defectos encontrados.

Se aplica la fórmula:

$$DPC = ND/NPC$$

## 2. Tiempo promedio de reparación de defectos

Esta es una medida del esfuerzo necesario para reparar los defectos identificados en las pruebas. También esta métrica es medida para cada tipo de pruebas (en-linea y batch). Esta métrica proveerá información para la planificación de tiempo y recursos de la actividad de rectificación en pruebas futuras.

TPRD : Tiempo promedio de reparación de defectos.

TRD = tiempo informado de reparación de los defectos encontrados. Esta es la suma de los tiempos de reparación de todos los defectos encontrados.

ND = número de defectos encontrados.

Se aplica la fórmula:

$$\text{TPRD} = \text{TRD}/\text{ND}$$