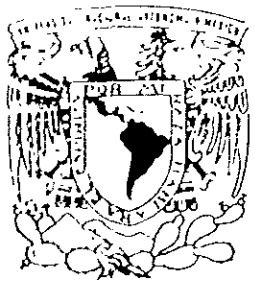


01170

UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO



DIVISION DE ESTUDIOS DE POSGRADO
FACULTAD DE INGENIERIA

MANTENIMIENTO PARA EL SISTEMA DE CONTROL
DEL PROCESO DE SOFTWARE

T E S I S
QUE PARA OBTENER EL GRADO DE:
MAESTRO EN INGENIERIA
CON OPCION EN ELECTRICA
P R E S E N T A :
RITA CAROLINA RODRIGUEZ MARTINEZ

DIRECTOR DE TESIS: GUADALUPE IBARGÜENGOITIA



MEXICÓ, D. F.

285421

2000



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

A J.H.S.

Tú que me aclaras todo, tú que me iluminas los Caminos para que alcance mis metas; yo quiero agradecerte en esta pequeña frase todo lo que me has brindado, y confirmar una vez más que no deseo apartarme de ti

A mis padres y a mis hermanos

A quienes quiero, admiro y respeto, les doy gracias por brindarme su apoyo y comprensión

A mi pequeña traviesa Nahani

A mis pequeños bebes que aunque ya no estén conmigo, siempre vivirán en mí

Cevalli, Vodka y Coyolli

Ing. Carolina Flores Illescas
M. C. Guadalupe Ibarquengoitia
Dra. Hanna Oktaba
Ing. Mario Rodríguez Manzanera

A quienes les agradezco profundamente la
paciencia, ayuda, orientación y asesoramiento
que me brindaron para la elaboración del
presente estudio

A todas aquellas personas que colaboraron de
alguna manera para el desarrollo de este trabajo

Gracias!

A ti

INDICE

Introducción

I.	Mantenimiento del Software	1
1.1	Concepto de Mantenimiento	4
1.2	Tipos de Mantenimiento	4
1.3	Actividades para el mantenimiento	5
1.4	Concepto de la Administración de la Configuración del Software (ACS)	6
1.4.1	Actividades de la ACS	7
1.4.2	Control de la Configuración	7
II.	La Calidad obtenida a través del Proceso Personal de Software	9
II.1	La importancia de la calidad	9
II.2	¿Cómo obtener la calidad a través del Proceso Personal de Software?	10
II.2.1	El propósito del Proceso Personal de Software	11
II.2.2	Concepto de Proceso de Software	12
II.2.3	La estrategia del Proceso Personal de Software	13
II.3	¿Por qué es bueno tener calidad?	13
III	Metodología para el Mantenimiento de un Sistema	15
III.1	Metodología Propuesta	15
III.1.1	Elaborar un plan de trabajo	15
III.1.2	Controlar los cambios	16
III.1.3	Construir una nueva versión o un nuevo sistema	20
III.1.4	Probar una nueva versión o un nuevo sistema	20
IV	Descripción del Sistema a Mantener	23
IV.1	Antecedentes del Sistema	23
IV.2	Objetivo General	24
IV.3	Descripción General del SCPS	24
IV.3.1	Arquitectura del SCPS	24
IV.3.2	Descripción General del Menú Principal y Funcionamiento	28
IV.3.3	Diseño de la Base de Datos	29
IV.4	Funciones del SCPS	31
IV.5	Problemas Detectados	36
V.	Mantenimiento Aplicado al Sistema para el Control del Proceso de Software	42
V.1	Cambios Propuestos	42
V.2	Aplicación de la Metodología Propuesta	43
V.2.1	Elaboración del plan de trabajo	44
V.2.2	Control de cambios	46
V.2.3	Construcción de la nueva versión o del nuevo sistema	50
V.2.4	Probar la nueva versión o el nuevo sistema	54
VI.	Resultados.....	57

Conclusiones	61
Apéndice A	63
Formatos de Registro de Defectos por Pantalla	
Apéndice B	68
Formas para las Propuestas de Cambio	
Bibliografía	123

INTRODUCCIÓN

En la actualidad existen varios sistemas de software desarrollados por ingenieros de software; pero ¿cuántos de estos sistemas se consideran sistemas fáciles de mantener?

El mantener los sistemas de una empresa funcionando es por lo general difícil y cada vez se requiere en mayor grado, porque la mayoría de los sistemas están sujetos a cambios casi constantemente.

Entre las causas posibles que pueden provocar la necesidad de dar mantenimiento pueden señalarse las siguientes

De sistemas:

- Sistemas que producen resultados incorrectos.
- Sistemas que provocan el desbordamiento de los stacks de los sistemas operativos.
- Sistemas que dejan a la máquina estática o en estado de espera continua.
- Sistemas que tiran al procesador.
- Sistemas en donde se requiere mayor precisión en los resultados.

Externos:

- Nuevas necesidades de la empresa
- Necesidad de ajuste de variables en su precisión, tamaño, etc.
- Customización. Necesidad de adecuar los parámetros internos del sistema a nuevas necesidades de la empresa

Las causas que provocaron la necesidad de dar mantenimiento a los sistemas pueden a su vez ocasionar problemas, tales como:

De sistemas:

- Que la documentación del sistema original automáticamente requiera de ser actualizada
- Que el código generado por los nuevos programas o parches o versiones pueden volver lento al sistema u ocupar más espacio, requerir más buffers o afectar los tiempos de entrada y salida
- La herramienta utilizada para el mantenimiento puede heredar sus limitaciones al desarrollo producido durante el mantenimiento, de tal forma que pueda requerirse ya sea el cambio de la herramienta totalmente o la adquisición o desarrollo de un nuevo sistema.

Externos:

- Todo mantenimiento implica la distracción de personal dedicado a desarrollo para efectuar las labores de análisis, detección y corrección de fallas lo cual representa en forma directa un impacto sobre los presupuestos de las diferentes áreas del sistema dentro de la empresa.

- La necesidad de llevar a cabo la contratación de expertos externos para la solución del problema, implica a su vez en ocasiones costos adicionales.
- La necesidad de adquirir una herramienta especializada para dar el mantenimiento.

En algunas ocasiones, cuando se llega a presentar una falla en un sistema. Esta se corrige inmediatamente lo que se llama "realizar una corrección o modificación al vuelo", inicialmente se detecta la causa que originó el problema, se hace el parche correspondiente y se corrige el error. Posiblemente la falla no se registro en una bitácora, ni se observó si afectaba en alguna otra sección del sistema. Lo único que interesaba en ese momento al encargado del sistema o a la institución era solucionar el problema inmediatamente. Sin embargo, aunque la falla finalmente es resuelta ésta se pudo corregir de una manera ordenada contando con la información adecuada y con base a realizar una serie de pasos que le permitirán al encargado de resolverla ver de dónde se originó, que programas se pueden ver involucrados, cuál es la mejor manera de solucionarlo y finalmente valorar el cambio y sus implicaciones

Con base a lo anterior, se puede asegurar que mantener un sistema es un proceso difícil, sobretodo cuando se proponen a la vez mejoras durante la solución

De aquí que es necesario que se proponga una metodología que permita de una manera aceptable y bajo el seguimiento de formas de registro dar mantenimiento al sistema, basándose principalmente en la elaboración de un plan de trabajo.

En el primer capítulo de este documento, se describe el concepto y los tipos de mantenimiento, se señalan las actividades que se efectúan usualmente durante el mantenimiento enfocándose prioritariamente a las actividades de la Administración de la Configuración de Software (ACS), dado que es un concepto que permite el desarrollo y mantenimiento de productos de software de calidad y brevemente se introducen los conceptos de control de configuración

En el segundo capítulo, se introduce el tema de la calidad enfocada principalmente para obtener productos de calidad a través del Proceso Personal de Software, se menciona la importancia de la calidad, los propósitos y estrategias del PSP y el por qué es importante obtener calidad, cuando se quiere dar mantenimiento a un sistema existente.

En el tercer capítulo, se presenta la metodología que deberá aplicarse para dar mantenimiento a cualquier tipo de sistema. Aquí se detallan cada uno de los pasos que deberán seguirse desde que se propone un cambio a este, hasta que se obtiene el resultado de su implementación, debiéndose desde luego haber elaborado un plan de trabajo con anticipación, el cual sirve como una guía para realizar cualquier tipo de cambio al sistema. Se muestran y se explican cada una de las formas que deberá llenarse inicialmente para proponer un cambio.

En el cuarto capítulo, se describe el sistema a mantener (Sistema para el Control del Proceso de Software, versión 1.0), el cual fue desarrollado por un grupo de alumnos de la maestría en Ciencias de la Computación. Se mencionan sus antecedentes, objetivos y arquitectura, así como, las funciones de cada uno de los módulos que lo integran. Se describen y muestran los defectos encontrados al aplicar la metodología y su documentación.

En el quinto capítulo, se aplica la metodología propuesta al “Sistema para el Control del Proceso de Software”. Se presentan de manera general las propuestas de solución que mejorarán la calidad y funcionamiento del sistema, elaborando un plan de trabajo. Se presentan, además los formatos de solicitud, aprobación, resultado de la implementación y liberación del cambio. Formatos a través de los cuales la Mesa de Control de Cambios (MCC) se encarga de dar la continuidad debida a los cambios propuestos.

Los cambios aceptados por la MCC, serán implementados por el ingeniero de software, y probados por la autoridad responsable del control de la calidad del sistema, quién se encargará de verificar los resultados obtenidos. Se menciona en que lenguaje de programación se desarrolló la nueva versión de SCPS y el equipo que se utilizó para su desarrollo incluyéndose información relativa al diseño de la base de datos. Por último, se presentan los cuestionarios que se le aplicaron a los usuarios para conocer su opinión acerca de la funcionalidad alcanzada y sobre el diseño de la interfaz humana utilizada en esta versión del sistema (v. 2.0).

En el sexto capítulo, se exponen los resultados obtenidos durante la implementación y pruebas de los cambios propuestos, finalizando con las conclusiones que se derivan de la crítica o discusión de estos

Por último, se presentan las conclusiones del desarrollo de esta tesis y se incluye a continuación dos apéndices que complementan de manera global la presentación de los resultados y los formatos utilizados

En el apéndice A, se muestran los resultados que se obtuvieron a través de los formatos de registro de defectos después de revisar detalladamente la funcionalidad y diseño de la interfaz humana de SCPS, versión 1.0

En el apéndice B, se muestran las formas de cada una de las solicitudes de cambio, aprobación, implementación, liberación y los resultados que se obtuvieron después de implementar los cambios presentados a la Mesa de Control de Cambio y liberados por el encargado de la Calidad.

I. Mantenimiento del Software

El mantenimiento de un sistema implica la generación de una nueva versión, porque permite efectuar cambios más precisos al sistema original a medida que se añaden nuevos requisitos, se corrigen y/o eliminan anomalías detectadas en versiones anteriores, aquellos defectos que podrían surgir al momento de corregir otros defectos conforme se llevan a cabo los cambios propuestos para mejorar la funcionalidad del sistema original, observando si la integridad del sistema resiste los cambios efectuados.

I.1 Concepto de Mantenimiento

Una vez que se entrega el sistema a producción, sus desarrolladores y usuarios finales ya tendrán probablemente una serie de mejoras o modificaciones que les gustaría realizar en las versiones siguientes. El mantenimiento de un sistema implica gestionar una lista de nuevas tareas. Así el mantenimiento es, en cierto sentido, la evolución continua de un sistema, sus productos son similares a los de la fase de desarrollo.

De acuerdo a lo anterior el estándar [IEEE Std 610 12-90] define el concepto de mantenimiento como:

"La modificación de un producto de software, en fecha posterior a su liberación, con el propósito de corregir fallas, mejorar el rendimiento u otras características y/o para adoptar dicho producto a nuevos requerimientos tales como: sistema operativo, interfaz, etc "

Durante la etapa de mantenimiento, a los programadores se les pedirá realizar mejoras continuas a un sistema existente. Estos programadores a menudo, no son los creadores originales del sistema, sino un grupo de personas que tendrán que documentarse en todo lo relacionado con este.

A medida que más usuarios utilizan el sistema, se descubrirán nuevos errores y formas de uso que no pudieron ser detectadas con anterioridad.

El mantenimiento de software consiste de una serie de actividades, requeridas para conservar un sistema en forma operacional y servible después de ser aceptado y situado dentro de la producción.

I.2 Tipos de Mantenimiento

El mantenimiento de software comúnmente se clasifica en cuatro tipos que son:

1. *Mantenimiento Perfectivo*

Altera el comportamiento funcional del sistema mejorándolo, como resultado de un cambio de los objetivos o requerimientos de dicho sistema.

2. *Mantenimiento Adaptativo*

Produce un cambio en el sistema en respuesta a una modificación de las especificaciones de entrada y salida o en las especificaciones de software o del hardware.

3. *Mantenimiento Correctivo*

Detecta y corrige defectos del sistema que provocan salidas incorrectas o terminaciones anormales

4. *Mantenimiento Preventivo*

Actualiza el sistema para anticiparle problemas futuros, esta actualización mejora la calidad del software y la documentación, así como otros factores de calidad. Las modificaciones no afectan el comportamiento funcional del sistema

I.3 Actividades de Mantenimiento.

El mantenimiento involucra actividades que son ligeramente diferentes de las requeridas durante el desarrollo del sistema. Si se ha hecho un buen trabajo en el diseño y la documentación de la arquitectura original, la incorporación de una nueva funcionalidad o la modificación de algún comportamiento existente se realizará fácilmente.

Además de las actividades usuales como análisis, diseño, etc., el mantenimiento implica una actividad de planificación que consiste en:

- Asignar prioridad a las peticiones de mejoras básicas o informes de errores que denotan problemas del sistema.
- Estimar el costo de volver a desarrollar el sistema reemplazándolo en comparación al costo de mantenerlo.
- Establecer una colección significativa de estos cambios tratarlos como puntos funcionales para la siguiente versión.
- Añadir, si los recursos lo permiten pequeñas mejoras, y más localizadas para la siguiente versión.
- Gestionar administrativamente la aceptación de las modificaciones, cambios o propuestas para ser incluidas como parte de la evolución de una versión posterior.

Existen muchas actividades que deben realizarse constantemente para llevar a cabo un mantenimiento. Sin embargo, para el propósito de esta tesis nos enfocaremos en algunas actividades propias de la "Administración de la Configuración del Software (ACS)", porque es un elemento importante para garantizar la calidad del software, que se enfoca en el manejo de los cambios, a través de las actividades de: *identificar, controlar, informar y auditar la configuración del producto* de tal forma que el sistema no se des controle y se obtengan productos de calidad.

El objetivo principal de ACS es controlar completamente las modificaciones que se realizan durante el proceso de mantenimiento

I.4 Concepto de la Administración de la Configuración de Software (ACS)

La Administración de la Configuración de Software (ACS) es de gran importancia para el desarrollo y mantenimiento de productos de software de calidad. Su importancia comienza a finales de la década de los cincuenta y principios de los sesenta con la administración de la configuración del hardware, lo que da origen a la especialización para el software.

La Administración de la Configuración de Software, ACS (Software Configuration Management) tiene como objetivos:

- Administrar un sistema de software, desde el inicio de su desarrollo hasta que sale de uso por obsoleto.
- Supervisar la construcción del sistema y el mantenimiento preventivo que se aplicará cuando el sistema esté en funcionamiento

{Bersoff} lo define como:

‘La disciplina de identificar la configuración de un sistema en puntos discretos del tiempo, con el propósito de controlar sistemáticamente los cambios que se sucedan en esta a través del ciclo de vida del sistema’.

La Administración de la Configuración propone los siguientes objetivos:

1. Garantizar una versión correcta del sistema.
2. Administrar los cambios que se presenten en el sistema de manera que la configuración no afecten a otros módulos, programas o sistemas.
3. Coordinar la evolución del producto a través de diferentes versiones.
4. Asignar un nombre adecuado a cada uno de los componentes que surjan durante el desarrollo del software en base a un estándar seleccionado durante la planeación del desarrollo
5. Informar los cambios en el estado de la configuración a cada uno de los miembros del proyecto y usuarios del sistema.
6. Asegurar que se entregue la versión correcta al cliente.
7. Administrar y manejar correctamente las bibliotecas de las diferentes versiones del sistema.
8. Administrar los productos que se adquieren de diferentes proveedores de software para que no afecten al sistema

I.4.1 Actividades de la ACS

La Administración de la Configuración de Software no estipula un método de diseño ni un modelo del ciclo de vida predeterminado, ni define como debe ser juzgada la calidad de los componentes. Por el contrario, provee una base sólida para llevar a cabo todas las actividades de la Ingeniería de Software. La ACS se enfoca en la integridad del desarrollo e indicios que se registran, comunican y controlan durante el desarrollo y mantenimiento del sistema. Esto se logra a través de cuatro actividades principales, que son:

1. Identificación de la configuración
2. Control de la configuración
3. Informe de la configuración
4. Auditoría de la configuración

De las actividades de la Administración de la Configuración de Software solo haremos referencia al Control de la Configuración, debido a que al propósito central de este trabajo es de proponer una metodología que permita controlar, registrar y corregir los cambios detectados durante las etapas de *mantenimiento preventivo y correctivo*.

I.4.2 Control de la Configuración

La Administración de la Configuración de Software somete a un sistema al control de sus componentes que han sido o están siendo producidos. Con el control de la configuración se establecen los procedimientos necesarios para proponer, evaluar, revisar, aprobar e implementar los cambios de los componentes para cada línea de base. En donde una línea de base define un módulo a evaluar tratando de analizar sus componentes sustantivas.

Tomando en consideración que la implementación de un cambio propuesto no necesariamente constituye una solución al problema, todo cambio propuesto a un componente de software deberá ser controlado a través de un procedimiento bien definido.

El organismo responsable de aprobar los cambios a un sistema dentro de un ambiente de desarrollo o mantenimiento es conocido como la Mesa de Control de Cambios (MCC), (Configuration Control Board). La Mesa de Control de Cambios tiene la facultad de evaluar las propuestas y aprobar la implementación de los cambios propuestos al sistema.

[MIL STD 480] lo define como:

“La evaluación, coordinación, aprobación e implementación de todos los cambios a la configuración de un componente después de un establecimiento formal de su identificación”

Los cambios que se pueden efectuar en los componentes de software son de diversos tipos:

- Variantes
- Revisiones
- Versiones

Variantes

Una variante de un componente de software realiza la misma función para situaciones ligeramente diferentes. Está pensada para ser una parte intercambiable.

[Babi 86] dice que una configuración común incluye solo una variante de cualquier componente de software. A diferencia de las revisiones, una nueva variante no sustituye a una anterior. Múltiples variantes pueden coexistir como alternativas. Una variante de un componente de software evoluciona y por lo tanto existirán varias revisiones de una variante.

Revisiones

El propósito de una revisión se da cuando se realiza un cambio; los componentes de software que se están controlando a través de la ACS, solo podrán ser modificados como producto de una revisión. A cada revisión le antecede otra, a excepción del primer componente del que estamos hablando, es decir al que le dio origen a las demás. Las revisiones reflejan un mejor desempeño y funcionalidad.

[Babi 86], define las revisiones como:

“Dos versiones de un componente de software que pueden diferir porque una es revisión de la otra. Una revisión es una nueva versión pensada en sustituir a la anterior.”

Versiones

En el desarrollo de un sistema, sus componentes sufren transformaciones, que empiezan a diferir en pequeños pedazos de código con relación a su antecesor. Estos mismos componentes al sufrir transformaciones generan nuevas instancias para ese sistema.

Una definición de versión, por [Somm 92] es:

“Una instancia del sistema que difiere de otras. Estas diferencias pueden incluir una nueva funcionalidad u operar en una configuración de hardware diferente.”

Una vez que hemos visto los principales conceptos relacionados con el mantenimiento, pasaremos al siguiente capítulo en el que enfocaremos nuestra atención en uno de los temas de mayor interés en el proceso de mantenimiento, el cual se refiere a la calidad propia obtenida en el mantenimiento de un sistema.

II. La Calidad obtenida a través del Proceso Personal de Software

En este capítulo, se trata primeramente la importancia de la calidad en los productos con base al método del Proceso Personal de Software (PSP), el cual está diseñado para ayudar a los ingenieros de software a hacer buenos trabajos. Se presenta de forma general una descripción del PSP, dado que el sistema que se describe en el capítulo siguiente se desarrolló para apoyar este proceso.

II.1 La importancia de la calidad

Primeramente se define el concepto de calidad.

Calidad: *Es la forma en la que nuestros productos y servicios son percibidos por nuestros clientes.*
[Quin 99]

[ISO 8402:1994], lo define como:

El conjunto de características de una entidad (por ejemplo un producto o servicio) que le confieren la aptitud para satisfacer las necesidades establecidas y las implícitas.

Para el ingeniero de software, la calidad de los programas que produce tendrá una importancia crítica tanto para los clientes como para los usuarios. Puesto que el software desarrollado se usa en muy diversos tipos de aplicaciones, los defectos de este pueden causar daños económicos e incluso daños físicos.

Cuando los programas de software que uno escribe se usan para hacer trabajo financiero o procesamiento de texto, sus defectos pueden ser molestos o incluso costosos, pero probablemente nadie perderá la vida ni quedará lisiado. Cuando el software es parte de un sistema que vuela aviones, maneja autos, controla el tráfico aéreo, coordina fábricas o determina el funcionamiento de plantas de energía, los defectos pueden tener consecuencias indescriptibles y peligrosas.

Hay gente que ha muerto a causa de defectos de software [Leve 95].

Es entonces importante que todos los ingenieros de software reconozcan que su trabajo puede tener serios impactos en la salud, la seguridad y el bienestar.

Es conveniente hacer notar que cualquier defecto en una pequeña parte de un programa grande puede potencialmente causar serios problemas. Si bien puede parecer poco probable, un pequeño defecto en cualquier parte de un sistema grande puede potencialmente ser desastroso.

Por lo tanto, para contar con sistemas realmente seguros, debemos aspirar a producir software sin defectos y a través de ello estaremos obteniendo productos de calidad.

A pesar de que se estudian, se revisan y se prueban meticulosamente los aspectos complejos del diseño, los aspectos más simples por lo regular se descuidan. Como resultado, la causa más común de problemas con sistemas de software de gran escala radica en simples distracciones. Estas distracciones son generalmente defectos que pudieran considerarse "errores tontos" los cuales son cometidos individualmente por ingenieros de software. Aunque típicamente la mayoría de estos defectos se detectan al integrar y probar el sistema, algunos de ellos pasan inadvertidos durante la etapa de pruebas presentándose hasta que el software se encuentra en producción, cuando producen mayores daños.

El problema es que los ingenieros de software frecuentemente confunden "simple" con "fácil". Sienten que sus defectos simples serán fáciles de encontrar. Frecuentemente se sorprenden al saber que defectos triviales como: usar un nombre equivocado de parámetro, fijar incorrectamente una condición o terminar equivocadamente un ciclo, pueda sobrevivir la etapa de pruebas y causar serios problemas durante su utilización. Estas fallas son las que causan por lo general problemas por los que se gastan millones y millones de pesos para encontrarlos y corregirlos.

La calidad de los programas grandes depende de la calidad de los programas, rutinas, módulos o subprogramas con que se construyen. Así, para producir programas grandes de alta calidad, cada ingeniero de software que desarrolla una o más de sus partes modulares debe hacer un trabajo de alta calidad. Esto significa que todos estos ingenieros deben estar personalmente comprometidos con el concepto de calidad.

Cuando los desarrolladores están comprometidos de esta manera, siguen y administran los defectos con tal cuidado que pocos posiblemente se encontrarán más tarde en la etapa de integración, pruebas o bien ya puestos en producción detectados por los clientes que usaran estos programas.

Cuando los desarrolladores están personalmente comprometidos con el concepto de calidad, están orgullosos de sus logros. Saben cuánto código han producido y cuántos defectos han encontrado otras personas en sus productos. Por el contrario, cuando no están personalmente comprometidos con la calidad, la tratan como si no fuera importante. Le dirán que probando se encontrarán la mayoría de los defectos y que los pocos que quedan se pueden fácilmente encontrar y corregir. No es sorprendente que cuando se sienten de esta manera, es difícil que produzcan productos de calidad.

II.2 ¿Cómo obtener calidad a través del Proceso Personal de Software?

Como se sabe, es caro probar el software, incluso para programas pequeños. También se sabe que se puede ahorrar tiempo encontrando los defectos al inicio de su desarrollo. Revisando los programas antes de compilarlos se puede reducir el número de defectos encontrados durante las pruebas, por lo tanto, las pruebas se harán en menor tiempo. Además encontrando los defectos al inicio se obtienen productos de mayor calidad.

Muchos programadores sienten que no deberían preocuparse de los defectos hasta empezar a compilar y probar sus programas. Se apuran entonces a diseñar y codificarlos para empezar a probar. Con el PSP, los ingenieros de software se enfocan en producir programas limpios y sin defectos desde el principio de su desarrollo, de esta manera ellos podrán obtener como resultado productos de calidad.

La razón de señalar esta estrategia es que, una vez que un producto está mal hecho, no hay herramientas para pulirlo. Una manera para demostrar esto es considerar algunos datos sobre la relación entre los defectos encontrados durante la compilación y en las pruebas, contra los encontrados por los usuarios estando el sistema en producción. Otro enfoque sería coleccionar datos sobre nuestros propios desarrollos y auto-criticarnos.

El Proceso Personal de Software (PSP) se diseñó para ayudar a los programadores e ingenieros de software a hacer un buen trabajo. Les enseña cómo aplicar métodos avanzados de ingeniería para realizar sus tareas diarias de desarrollo. Proveyéndolos con métodos detallados de estimación y planeación, les enseña como hacer un seguimiento de su desempeño con respecto a estos planes y les explica cómo definir procesos que guíen su trabajo.

De acuerdo a lo anterior, podremos decir que el Proceso Personal de Software se define como:

“Un proceso de auto-mejoramiento sobre la base de la autocritica diseñado para ayudar a controlar, administrar, y mejorar la manera de trabajar, es decir, es un marco estructurado de formas, guías, y procedimientos para desarrollar software”

Usando adecuadamente el proceso de PSP, se provee al programador o al ingeniero de software con los datos históricos que se requieren para hacer y cumplir con compromisos de una mejor manera; también, hace que los elementos rutinarios de su trabajo sean más predecibles y eficientes.

II.2.1 El propósito del Proceso Personal de Software

El único propósito del PSP es ayudar al ingeniero de software a ser mejor su trabajo, al contar con una herramienta poderosa que puede usarse de muchas maneras, por ejemplo: para administrar su trabajo, evaluar sus talentos, y/o crear nuevas habilidades. Puede ser una ayuda para planear mejor su trabajo, seguir su desempeño en forma precisa y medir la calidad de sus productos.

El PSP puede ayudarnos a hacer mejor nuestro trabajo tal como diseñar programas, desarrollar requerimientos, escribir documentación o dar mantenimiento a un sistema existente.

En lugar de usar un enfoque para cada tarea, los ingenieros de software necesitan un conjunto elaborado de herramientas y métodos; así como, el conocimiento para usarlos apropiadamente. El PSP nos brinda los datos y las técnicas de diseño para determinar cuáles tecnologías adoptar y cuáles métodos funcionarán mejor para nosotros. Nos da un marco para entender por qué cometemos defectos y cómo encontrarlos de la mejor manera. Nosotros podemos determinar la calidad de nuestros productos con base a nuestras propias revisiones, los tipos de defectos que no detectamos y los métodos que nos son más efectivos.

El PSP no es una respuesta mágica a todos los problemas de la ingeniería de software. Aunque puede sugerir dónde y cómo mejorar nuestro trabajo, cada quién debe hacer las mejoras que crea convenientes.

Para que quede más clara la idea de PSP, es necesario definir el proceso de software

II.2.2 Concepto de Proceso de Software

El *proceso del software* es la secuencia de pasos requeridos para desarrollar o mantener un software. Una *definición aceptada de proceso de software* es considerarlo como una secuencia de pasos. Cuando se presenta bien diseñada, la definición guía a los ingenieros de software conforme trabajan.

Más específicamente, el proceso de software fija el marco técnico y de administración para aplicar métodos, herramientas y gente a la tarea de desarrollo y mantenimiento del software, mientras que la definición del proceso identifica los papeles y las tareas específicas. La definición además, establece medidas y provee criterios de entrada y de salida para cada paso importante. Una definición diseñada efectivamente, ayuda a que cada parte del trabajo sea asignada adecuadamente y su estado sea seguro y supervisado. También provee un mecanismo ordenado de aprendizaje. Conforme se encuentran mejores métodos, se incorporan en las definiciones oficiales de procesos de la organización. De esta manera, un "proceso definido" permite que en cada nuevo proyecto se empleen las experiencias obtenidas por sí mismo, como las de sus predecesores.

Deming llama a los "procesos definidos" definiciones operacionales, algo acerca de lo que todo mundo puede comunicar y tratar de alcanzar [Demi 82]. Los procesos definidos proveen los siguientes beneficios [Kelln 88]

- Permiten la comunicación sobre procesos de manera efectiva entre los usuarios, los desarrolladores, los administradores, los clientes y los investigadores.
- Incrementan la comprensión de la administración, proveen una base precisa para la automatización de los procesos y facilitan la movilidad del personal
- Facilitan el reuso de los procesos. El desarrollo de los procesos es tardado y costoso. Pocos equipos de trabajo pueden darse el lujo de definir completamente la manera en que van a trabajar, pueden ahorrar tiempo y recursos usando los elementos estándar reusables proporcionados por un proceso definido
- Soportan la evolución del proceso de aprendizaje de procesos a través de una manera efectiva y un fundamento sólido para el mejoramiento de los mismos.
- Ayudan a la administración de los procesos. La administración efectiva requiere planes claros, así como, una manera precisa y cuantificada de medir su estado.

Un proceso bien definido también es más fácil de mejorar. Por ejemplo, pueden aparecer problemas si no se siguen algunos pasos o si el proceso mismo es defectuoso, o puede ser que la definición no sea clara, la comunicación pobre o bien que los ingenieros no estén motivados para usarla adecuadamente. Se pueden hacer correcciones para atender a las causas identificadas. El proceso, su definición e infraestructura que la soporta evolucionan con la experiencia.

II.2.3 La estrategia del Proceso Personal de Software

La estrategia del Proceso Personal de Software consiste en:

- Identificar los métodos y las prácticas que pueden usar los individuos (usuarios) de los sistemas grandes de software
- Definir el subconjunto de estos métodos y prácticas que puedan aplicarse al desarrollo de programas pequeños
- Estructurar estos métodos y prácticas de manera que puedan introducirse gradualmente en el proceso de desarrollo de software.
- Proveer ejercicios adecuados para practicar estos métodos en un ambiente educacional.

II.3 ¿Por qué es bueno tener calidad?

Muchos ingenieros de software y/o programadores ven los defectos del software como "bugs" (insectos). Los tratan como algo que surgió de algún lado, no como algo de lo que son responsables. Esto está equivocado. Son los ingenieros y/o programadores los que provocan la existencia de los defectos al olvidarse de corregir pequeñas deficiencias que se dejan pendientes y no se resuelven o bien se dejan para su arreglo posterior. Y claro cuando un programa tiene un defecto, ¡es defectuoso! El programador que desarrollo ese programa produjo un producto defectuoso. Cuando los ingenieros están comprometidos con la calidad, les importa esto. Y cuando les importa, son más cuidadosos en su trabajo. Cuando son más cuidadosos, producen mejores productos y de mejor calidad.

El primer paso al producir software de calidad es entonces, decidir que la calidad es importante.

El segundo paso es establecer la meta de producir programas sin defectos.

Para tener alguna posibilidad de lograr esta meta, debemos medir la calidad de nuestros programas y llevar a cabo los pasos para mejorar esa calidad. Debemos de señalar aquí que no necesariamente un programa sera de calidad con solamente no tener defectos, ya que pudieran intervenir otros conceptos como: desempeño logrado por el programa, eficiencia, eficacia y efectividad.

Como humanos de vez en cuando inyectamos defectos. El reto es administrar nuestra fiabilidad, no obstante esto se convierte en una lucha interminable.

No podemos simplemente aprender unos cuantos trucos sobre el manejo de la calidad y olvidarnos. Los productos de software del futuro serán progresivamente más elaborados y complejos. Por lo tanto, los problemas que enfrentaremos cambiarán constantemente.

Es bueno tener calidad para producir productos con la mínima cantidad de defectos, esto se puede lograr si desde que se inicia un proyecto el ingeniero de software empieza a darle mantenimiento, lo cual permitirá su facilidad de análisis, su interoperación, la facilidad de pruebas, la seguridad de acceso y su estabilidad. Es decir, desde la etapa de análisis, diseño, implementación y pruebas.

La cuestión no es si uno puede crear software sin defectos, sino si a uno le importa lo suficiente para continuar haciéndolo.

Dado lo anterior y considerando que el punto inicial de cualquier desarrollo deberá contemplar a su vez un proceso *continuo de mantenimiento*, es de gran importancia el visualizar una metodología que se pueda aplicar para el mantenimiento de un sistema, lo cual es el propósito central del siguiente capítulo.

III. Metodología para el Mantenimiento de un Sistema

En este capítulo se propone una metodología para dar mantenimiento y que se puede aplicar a cualquier sistema. La metodología que se propone combina las actividades de la Administración de la Configuración de Software [Ibar 99] y de las actividades que la conforman [Capr 96] junto con algunas ideas originales.

III.1 Metodología Propuesta

Esta metodología se encuentra compuesta por las siguientes actividades

1. Elaborar un plan de trabajo
2. Controlar los cambios
3. Construir una nueva versión o un nuevo sistema
4. Probar una nueva versión o un nuevo sistema

A continuación trataremos cada uno de estos puntos con suficiente profundidad para proveer al lector con las bases necesarias de conocimiento sobre el proceso de mantenimiento.

III.1.1 Elaborar un plan de trabajo

Para aplicar la metodología que en este capítulo se propone, se debe primeramente elaborar un plan de trabajo en el cual se definirán las actividades a realizar, dando una descripción breve de estas e incluyendo qué recursos se requieren para llevarlas a cabo, es decir quién(es) es el responsable de efectuarla(s); cuál es la fecha en que se planea iniciar la primera actividad y cuál será la fecha estimada para la terminación de la última. La figura III 1 muestra la forma del Plan de Trabajo para registrar las actividades a desarrollar

Plan de Trabajo					
Actividad	Descripción	Recursos Requeridos	Responsable	Fecha de inicio	Fecha estimada de terminación

Figura III 1 Plan de Trabajo

Una vez que ha quedado establecido el plan de trabajo por los programadores en conjunto con los usuarios, se le entrega al administrador del proyecto del software, quién junto con el grupo de aseguramiento de la calidad y otros grupos involucrados, lo revisarán para determinar su *aprobación, modificación o rechazo*.

III.1.2 Controlar los cambios

Un componente puede requerir un cambio por diferentes razones, tales como: corrección de defectos, mejoras, aumento de funcionalidad, adaptación a un nuevo ambiente, cambio de plataforma, cambio de versión de los sistemas operativos, cambio de versión de compiladores, interpretes, etc. Cada cambio debe seguir un proceso de control.

El proceso de control de cambios se inicia cuando el usuario o aquel que detecta la necesidad de un cambio en el software presenta una solicitud como la que se muestra en la figura III 2 y cuyos detalles se presentan a continuación

- 1. *Solicitud de Cambio*, esta forma es utilizada para solicitar la información básica que se verá involucrada durante la continuidad del proceso. Para lo cual contiene: un número consecutivo de control de solicitudes de cambio, la fecha en la que presentó tal solicitud, el nombre de la persona que propone el cambio, el nombre y versión del sistema que se desea modificar, nombre del componente y versión del cambio, una pequeña descripción del porque se requiere efectuar dicho cambio, etc. La Mesa de Control de Cambios considerará el contenido de este formato para evaluar si el cambio propuesto o requerido procede o no.

En caso de que se desee efectuar una modificación a un componente, se deberá llenar y presentar una solicitud a la Mesa de Control quién se encargará de evaluarla.

Solicitud de Cambio	
Solicitud N° _____	Fecha de solicitud: _____ Hora: _____
Propuesto por (Nombre de la persona que solicita el cambio)	
Sistema/Proyecto: (Nombre del sistema que se quiere modificar)	Versión: (No Correspondiente actual)
Componente a ser cambiado. (Nombre del componte a modificar)	Versión: (No Correspondiente actual)
Descripción del cambio: (Qué es lo que se quiere cambiar en esencia)	
Razones para el cambio: (Porqué se quiere realizar el cambio)	
_____ Firma del solicitante	

Figura III 2 Forma para la solicitud de un cambio

- 2 Una vez que la solicitud fue llenada por algún miembro del grupo, o bien por la persona que detectó la necesidad de modificar un componente se presenta a la Mesa de Control de Cambios, quien se encargará de evaluarla y decidir si procede o no. A continuación, se calculará el costo estimado en horas en que deberá efectuarse el cambio. El resultado de la evaluación se reflejará en la forma *Aprobación de Cambio*, la cual se muestra en la figura III.3

Aprobación de Cambio	
	Fecha de recepción _____ Hora: _____
Solicitud N° _____	Aprobada () Rechazada ()
Autorizado por. (Nombre de la persona que autoriza el cambio)	
Valoración del cambio _____	
Prioridad del cambio () Crítico () Muy importante () Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas _____	
Comentarios:	
Fecha de decisión: _____	
_____ Firma de la MCC	

Figura III 3 Forma para la aprobación de un cambio

Cuando la Mesa de Control de Cambio termine de evaluar la solicitud, procederá a llenar la forma *Aprobación de Cambio*. Si la solicitud es rechazada, se anotará en el espacio de "Comentarios" el motivo por el cual no procedió, y por lo tanto se abandona. En caso contrario, si la solicitud fue aceptada, se hará el llenado correspondiente indicando el nombre de la autoridad que valoró, evaluó y lo cuantificó, la fecha en que se decidió realizar el cambio y una valorización del cambio consistente en cuanto y cómo repercutirá en las demás componentes del sistema. la prioridad del cambio y su costo estimado en cuanto al número de horas hombres que se utilizarán; por último, la autoridad responsable (MCC) firmará la forma para proceder a su implementación.

- 3 Entrega de la forma autorizada al programador El formato de aprobación de cambio debidamente autorizado por la MCC se entrega al Ingeniero de software, quién se encargará de implementar el cambio, probarlo y observar el impacto que pueda tener con los otros componentes. Esta función sólo podrá ser realizada por el Ingeniero de software, porque es quien conoce la infraestructura del software y podrá determinar con claridad el tiempo que se llevará para hacer la implementación del cambio Para ello y concluido dicho proceso tendrá que llenar una forma *Resultado de la Implementación de Cambio*, en la cual se anotará el número de la solicitud de cambio, la fecha en que se recibió la aprobación, el nombre de la persona que se encargó de efectuar el cambio, el número total de cambios implementados, el número de versiones (en caso de que hallan sido más de una), la fecha en que se inició y se terminó el proceso de mantenimiento, una descripción breve del cambio o cambios que se efectuaron, ver figura III 4.

Resultado de la Implementación de Cambio	
Solicitud N° _____	Fecha de recepción: _____ Hora: _____
Ingeniero de software: (Nombre de la persona que efectuo el cambio)	
N° total de cambios implementados en los componentes: _____	Versión(es) _____
Descripción de los cambios.	
Fecha de inicio: _____	Fecha de terminación: _____
_____ Firma del Ingeniero de Software	

Figura III 4 Forma para implementar los resultados de un cambio

- 1 Cuando el Ingeniero de software llene la forma de implementación, la entregará al Departamento de Seguridad y al Departamento de Control de la Calidad, quiénes se encargarán de evaluar los cambios implementados, el resultado de la evaluación o prueba en el ambiente de mantenimiento lo reportarán en la forma *Liberación de Cambio*, mostrada en la figura III 5. Los datos que la forma solicita son número de la solicitud de cambio, fecha en que recibió la forma de implementación, el nombre de la persona que liberó o probó el cambio, comentarios (en caso de que un cambio fallará esté se estará revisando hasta que se libere, se anotará el problema en el espacio de comentarios), por último el probador firmará la solicitud que liberará el cambio a producción.

Una vez liberado el cambio, se registrará en la forma *Liberación de cambio*. Esta forma se le devolverá a la MCC. Quién se encargará de registrarla en la biblioteca y le dará seguimiento a la siguiente solicitud, en caso de haberla.

Liberación de Cambio	
Solicitud N° _____	Fecha de recepción: _____ Hora: _____
Autoridad responsable: (Nombre de la persona que libera el cambio)	
Comentarios	
Fecha de liberación _____	
Firma del Depto. de Seguridad y Calidad (Autoridad responsable de liberar el cambio)	

Figura III 5 Forma para la liberación de un cambio

La figura III.6 muestra el procedimiento que se sigue para solicitar un cambio, implementarlo, probarlo y por último liberarlo bajo la supervisión y autorización de la Mesa de Control de Cambios (MCC).

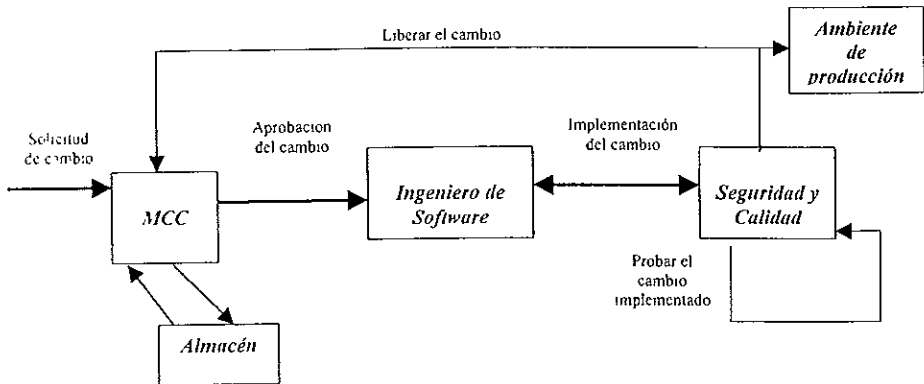


Figura III 6 Relacion de actividades, roles y productos [Ibargüengoitia]

- 5 Por último, se juntan todos los componentes que conforman la nueva versión, se complementa la documentación y se les entrega a los usuarios o clientes que solicitaron la modificación o cambio.

La nueva versión se origina de haber efectuado los cambios necesarios a los componentes que integran el sistema

$$\text{Config}_n + \text{cambios} = \text{Config}_{n+1}$$

III.1.3 Construir una nueva versión o un nuevo sistema

Para construir una nueva versión el ingeniero de software tendrá que evaluar los cambios solicitados, para verificar si estos cambios sólo afectan una parte del sistema original, tales como agregar un campo a una tabla de la base de datos, cambiar el nombre de una variable, ampliar un campo de texto, validar un campo de captura, hacer más rápido el proceso de extraer información de la base de datos, etc; si estos cambios se pueden corregir con solo actualizar la versión actual efectuando los cambios pertinentes podremos hablar de una nueva versión. Si no es así, estaremos por construir un nuevo sistema. Si los cambios no son soportados por el lenguaje de programación que se utilizó en la versión original, es mejor buscar otro lenguaje o una herramienta más potente que permita hacer las mejoras de una manera más accesible, reutilizando código, importando bibliotecas, manejando una interfaz gráfica para el diseño de ventanas, entonces estaremos hablando efectivamente de construir un nuevo sistema.

Antes de implementar los cambios aprobados por la Mesa de Control de Cambio, se habrá elaborado el diseño del nuevo componente, en caso de que estemos hablando de la construcción de un nuevo sistema, por ejemplo: el diseño de la interfaz gráfica de la información que el sistema proporciona y solicita al usuario, de los mensajes de ayuda, de validación o de error, y de la ayuda proporcionada por el sistema para que el usuario pueda consultarla cuando lo crea conveniente. También se tendrá que estandarizar el nombre de los objetos usados en la interfaz humana con el nombre usado en la base de datos, el código fuente, etc.

Por último, se deberá considerar que una vez que los cambios fueron implementados, se tendrá que revisar y probar que los cambios no repercutan en ningún componente del sistema y que estos realmente hagan lo que se espera que realicen, por ejemplo: que los mensajes se desplieguen apropiadamente y de acuerdo a la información que se está manejando, que los datos se almacenen y se extraigan de la base de datos sin ningún problema, es decir, se verifica que hay concordancia en el diseño, en los datos que se manejan y en su funcionalidad. Una vez que se probaron los cambios implementados, estos se liberan.

III.1.4 Probar una nueva versión o un nuevo sistema

En el punto anterior, se menciona que los cambios implementados se prueban para posteriormente liberarlos a producción. Este punto consiste, en que una vez que el sistema es liberado por el ingeniero de software, la nueva versión se someterá nuevamente a pruebas con la ayuda del usuario que no esté familiarizado con el sistema para detectar posibles fallas que no fueron detectadas por el ingeniero de software

El usuario se encargará de utilizar el sistema para que nos brinde su opinión sobre la interfaz humana, la información solicitada, la funcionalidad, la ayuda, los mensajes, etc. Para ello, se le hará entrega de unos cuestionarios. Un ejemplo de los cuestionarios aplicados a los usuarios se presentan en la figura III.7 y III 8, los cuales se diseñaron especialmente para probar el Sistema con el usuario para el Control del Proceso de Software.

En ellos los usuarios podrán plasmar sus opiniones, las cuales serán de gran ayuda para decidir si la nueva versión está lista para su liberación.

Si la nueva versión del sistema se libera, entonces se procede a actualizar la documentación conforme a lo implementado. Los documentos que podrían ser afectados por los cambios, son:

- Manual del usuario
- La documentación del análisis
Tal como: El diccionario de clases, los diagramas de clases, etc.
- La documentación del diseño
Tal como: El diseño detallado, clases de la interfaz humana, etc.
- La guía de pruebas

Lista de actividades

**Para la evaluación de la interfaz humana del
"Sistema para el Control del Proceso de Software"**

Por favor realice la lista de tareas. Siéntase con la libertad de preguntar cualquier duda antes o durante la ejecución de las mismas. El objetivo es evaluar el diseño de la interfaz y su desempeño al llevar a cabo las siguientes tareas:

- a) Crear un proyecto nuevo
- b) Crear una actividad
- c) Medir el tiempo en realizar la actividad que dio de alta, por ejemplo revisar los correos de Internet.
- d) Cambiar su password
- e) Consultar la actividad medida
- f) Registrar un defecto de una actividad, por ejemplo que no hay comunicación de la red para revisar los correos de Internet.
- g) Imprimir un reporte de proyecto
- h) Registrar el tiempo que le llevo arreglar que la conexión de la red estuviera trabajando correctamente para revisar sus correos de Internet.

¡Muchas gracias por su cooperación!

Figura III 7 Cuestionario N°1

Cuestionario N° 1

**Evaluación de la interfaz humana del
"Sistema para el Control del Proceso de Software"**

Por favor contesta el siguiente cuestionario de acuerdo a sus experiencias en el uso de la interfaz. Las respuestas proporcionadas servirán para cubrir errores de diseño y mejorar la interacción usuario – computadora. En la columna de comentarios puedes agregar cualquier información que consideres importante.

a) Marque el recuadro correspondiente a su opinión para cada pregunta:

Pregunta	SI	NO	Comentarios
¿Los terminos usados en la interfaz son familiares para usted?			
¿Considera adecuada la forma en que se presenta la información?			
¿Disfruto del uso de la interfaz?			
¿Considera complicada la navegación entre las diferentes vistas?			

b) Califique la interfaz de acuerdo a la escala presentada, donde 5 y 1 significan la mayor y menor puntuación respectivamente. Puede usar valores intermedios para expresar opiniones intermedias

	5	4	3	2	1	0	
Facil de usar							Dificil
Simple de aprender							Complicado
Funcionalidad clara							Confusa
Eficiente para desplegar información							Deficiente
Respuesta rápida							Lenta
Método de navegación eficiente							Deficiente
Apariencia estética adecuada (Colores, tipo letra etc)							Inadecuada

- c) ¿Qué es lo que mas le gustó de la interfaz?
- d) ¿Qué es lo que más le desagradó de la interfaz?
- e) ¿Cuáles son las características de la interfaz que encontró confusas o difíciles de entender?
- f) ¿Qué cambios cree que son necesarios?
- g) Agregue cualquier comentario que crea útil para mejorar la interfaz

¡Muchas gracias por contestar este cuestionario!

Figura III 8 Cuestionario N°2

Una vez planteada la metodología que servirá de base para darle mantenimiento a un sistema, en el siguiente capítulo se describirá el sistema que se seleccionó para aplicarla y posteriormente detectar las posibles fallas que no fueron identificadas por los programadores que se encargaron del desarrollo del sistema

IV. Descripción del Sistema a Mantener

En este capítulo se describen los antecedentes generales del sistema a mantener, en este caso se seleccionó el Sistema para el Control del Proceso de Software, versión 1.0 el cual fue desarrollado por algunos alumnos de la Maestría en Ciencias de la Computación y que al no ser concluido requería de una reingeniería de software. A continuación se habla de su objetivo principal y de los objetivos secundarios, de su arquitectura tanto lógica como física. Se hace una descripción general del menú principal, de las funciones que efectúan cada una de las opciones que integran sus módulos. Por último, se muestran los defectos encontrados en cada una de las pantallas que integran dicho sistema.

IV.1 Antecedentes del Sistema

El Grupo de Ingeniería de Software Orientado a Objetos (GISOO), integrado por los alumnos de la Maestría en Ciencias de la Computación desarrollaron en 1997 un proyecto llamado "*Sistema para el Control de Proceso de Software (SCPS)*".

El SCPS se desarrolló como un ejemplo de un software para dar apoyo al Proceso Personal de Software (PSP). Este proceso tiene como objetivo ayudar a los ingenieros de software a controlar, administrar y hacer mejoras en su manera de trabajar. Se basa en el uso de diferentes formas de registrar tiempos, defectos y correcciones; proporcionando los datos históricos que se requieren al hacer nuevos compromisos o hacer que las rutinas que integran su trabajo sean más previsibles y eficientes

Humphrey creador de PSP [Hump 97], lo define como "*el proceso diseñado para ayudar a los ingenieros de software a hacer un buen trabajo*" Para lograrlo, es necesario que los ingenieros de software conozcan que actividades realizan y cuales son sus habilidades en dichas actividades; por ejemplo

- Saber el tiempo (promedio) que se tarda en desarrollar una actividad. Para ello se puede tomar nota de los tiempos en que se inicia y termina; así como, descontar las interrupciones que tuvo la actividad durante su ejecución.
- Medir el tamaño de los productos desarrollados en algún tipo de unidad (por ejemplo: páginas, listados de programas, líneas de código, casos de usos, etc.).
- Cuántos defectos (en promedio) ocurrieron en cada producto.

Es importante, que durante el desarrollo de los datos para conocer sus actividades con exactitud ellos lleven todos los registros de toda la información para que ésta pueda ser analizada posteriormente:

- Evaluar y mejorar sus habilidades en cada actividad;
- Administrar y planear mejor sus tiempos;
- Estimar tiempos en actividades similares;
- Hacer compromisos más responsables, etc.

De acuerdo a lo anterior, surgió la idea de crear el Sistema para el Control de Proceso de Software (SCPS), el cual tiene como finalidad automatizar el método del Proceso Personal de Software.

IV.2 Objetivo General del SCPS

El objetivo principal del sistema es calcular y registrar el tiempo efectivo, en minutos, que se utiliza en ejecutar la(s) actividad(es) que integran un proyecto, desde que inicia hasta que se termina.

Otros objetivos son:

- Determinar donde se gasta la mayor parte del tiempo y como se distribuye cuando se realiza una actividad. El tiempo usado es un simple registro del tiempo de reloj que se utiliza en cada actividad del proyecto
- Determinar el tamaño de un producto en algún tipo de unidad (ejemplo: casos de uso, líneas de códigos, métodos, número de programas, páginas, etc.)
- Permitir llevar un registro del tiempo usado en detectar y corregir los defectos encontrados durante la realización de la(s) actividad(es) de un proyecto y sumárselo al tiempo obtenido en cada actividad, para determinar el tiempo efectivo total del proyecto.

Para poder calcular el tiempo efectivo se requiere disciplina (y ética profesional) por parte de la persona que lo lleva a cabo, porque si el proyecto está enfocado al desarrollo de un sistema, es importante que los administradores obtengan datos confiables para poder calcular costos, calidad y productividad del proyecto. [Okta 99]

IV.3 Descripción General del SCPS

El Sistema para el Control de Proceso de Software provee las formas correspondientes para reunir los datos de cómo se comporta un ingeniero de software al realizar sus actividades que se le asignan en un proyecto. Para reunir estos datos, primero deberá entender los tiempos que se tardan en cada una de las actividades. Posteriormente se integran para observar como mejorar: su productividad, los tiempos efectivos en las actividades; la calidad del trabajo a fin de planear mejoras.

IV.3.1 Arquitectura del SCPS

El Diseño del Sistema se realizó siguiendo una metodología que es conjunción de las ideas de programación orientada a objetos. Por lo que a continuación, se definen las características de sistema bajo las cuales se llevo a cabo la programación del SCPS como un producto orientado a objetos utilizando las herramientas con las que contaban los diseñadores:

- Hay una documentación completa en cada una de las fases del proceso de desarrollo que incluye diagramas de casos de uso, de clases, de secuencia, de arquitectura, etc.

- Para la documentación los desarrolladores usaron una herramienta de software llamada Rational ROSE.
- El sistema corre en Windows 95.
- Se utilizó el manejador de base de datos Access para Windows 95
- Para la generación de ventanas se utilizó una herramienta llamada PARTS for Java ObjectShare versión 2.0 (permite diseñar el ambiente gráfico, es decir los componentes que integran cada una de las pantallas).
- Para la codificación se utilizó el lenguaje de programación JDK1.1.4 (Java Development Kit)

En cuanto al hardware se refiere se usó:

- ◆ Una PC Pentium
- ◆ Monitor a color con 32 MB en RAM.
- ◆ Disco duro de 1 2 GB

Se Adaptó la metodología de Booch para describir la arquitectura del SCPS, en dos capas: Arquitectura lógica y Arquitectura física

1 *Arquitectura Lógica* muestra el agrupamiento de clases en categorías

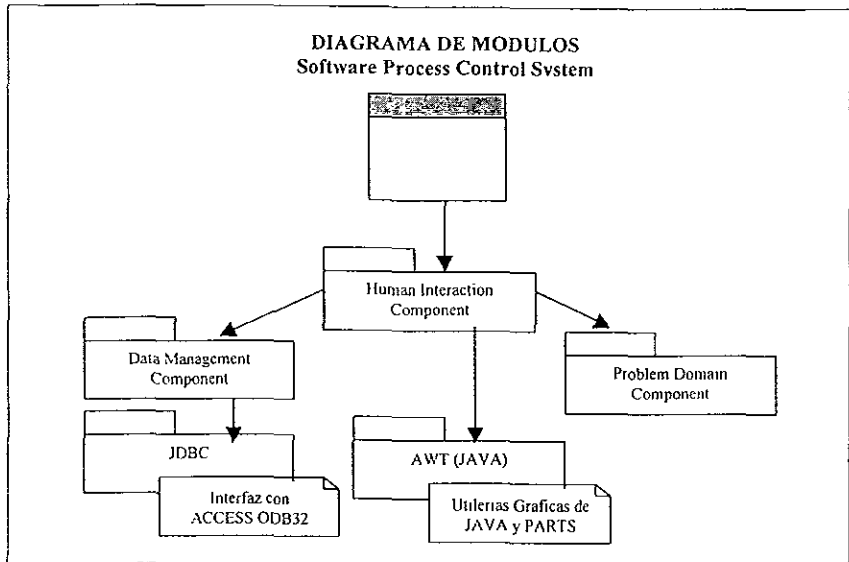


Figura IV 1 Diagrama de Módulos de la Arquitectura Lógica

En el diagrama de la figura IV.1, se puede observar que la arquitectura lógica cuenta con tres principales componentes de clases: *Interacción humana*, *Dominio del problema* y *Administración de datos*.

Cada componente engloba la declaración o definición de clases, objetos y otros detalles del lenguaje; la dependencia que es la única relación que puede darse entre dos módulos (es de compilación), se representa por una línea que apunta al módulo con relación a la que existe la dependencia.

2 *Arquitectura Física* muestra el agrupamiento de módulos en subsistemas.

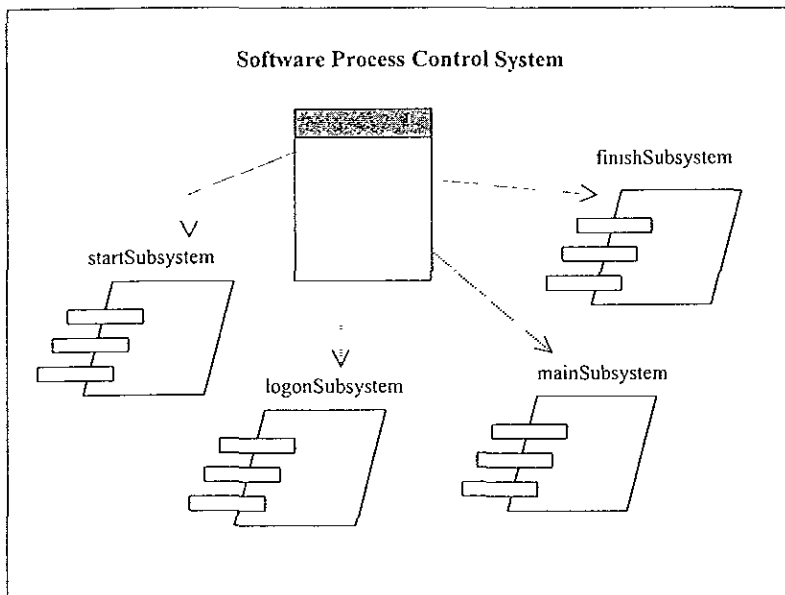


Figura IV 2 Diagrama de Módulos del Programa Principal (Arquitectura Física)

En este diagrama se observa que el programa principal se compone de cuatro subsistemas:

- **startSubsystem**: Despliega la pantalla *Presentación*, la cual muestra el logotipo del sistema, mientras se efectúa la conexión con la base de datos.
- **logonSubsystem**: Despliega la pantalla *Sesión*, la cual solicita login y password al usuario, si los datos son correctos, la pantalla se cierra permitiendo la entrada al menú principal, si los datos no son correctos permite dos intentos más antes de ejecutar el finishSubsystem.

- MainSubsystem. Despliega la pantalla *Menú Principal*, permite que el usuario seleccione una opción que se ejecuta mientras el usuario no cambie de opción, o bien seleccione la *opción salir del sistema*.
- finishSubsystem: Despliega la pantalla *Terminación*, la cual muestra el logotipo del sistema y un texto de despedida, cierra la pantalla y el programa termina. (Esto realmente no lo hace el sistema, solo muestra la pantalla *Opción*, la cual se muestra en la figura IV 3)

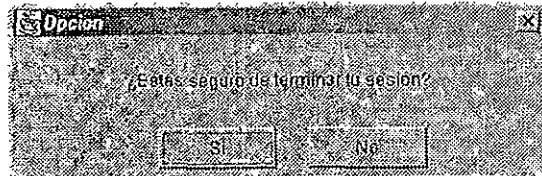


Figura IV 3 Pantalla Opción

Para terminar de describir la Arquitectura física del sistema, se presenta de manera general en el Diagrama de Módulos del Menú Principal que corresponden a las opciones del menú.

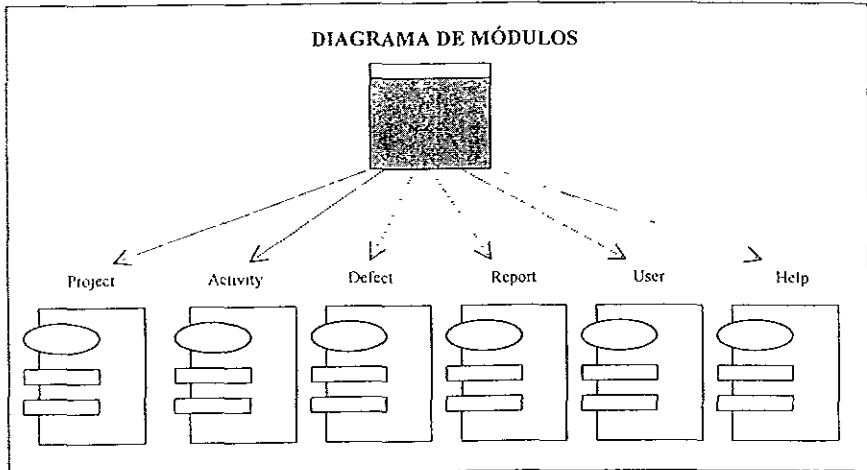


Figura IV 4 Diagrama de Módulos del Menú Principal

De acuerdo a la figura IV.4 el control de flujo de ejecución del menú se hizo siguiendo el *patrón command*, en donde cada elección de las opciones del menú se instancia a un objeto de la *clase command* respectivo y se ejecuta el método *execute*, el cual se encarga de construir la ventana correspondiente y de mostrarla. A su vez, para cada botón seleccionado dentro de una ventana se ejecuta el método *execute*, con los datos que obtuvo de la ventana que lo invocó.

El patrón de diseño se utilizó para implementar la clase *Command* permite que el Menú Principal se desacople de las operaciones que le corresponden a cada *"item"*, delegando esta tarea a los *comandos* de la *Clase Abstract Command*. Con este patrón se tiene la ventaja de poder agregar o eliminar *"items"* del Menú Principal sin afectar a los demás, agregando o eliminando su correspondiente clase *Command*.

IV.3.2 Descripción General del Menú Principal y Funcionamiento

El Menú Principal (mainSubsystem) se encuentra integrado por seis módulos:

- Proyecto
- Actividad
- Defecto
- Reporte
- Usuario
- Ayuda

Podríamos decir que la funcionalidad real del sistema recae en los cuatro primeros módulos. Estos módulos son dependientes uno de otro, porque primero es necesario crear un proyecto, identificar las actividades que lo integran y al ejecutar cada una de ellas, se contabiliza un registro de los tiempos. En caso de detectar un defecto, éste se registra en el *Módulo Defecto* con la opción *Corregir* o en el *Módulo Actividad* en la opción *Ejecutar Actividad*.

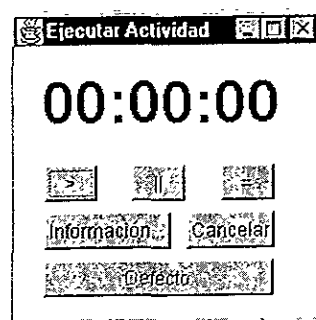


Figura IV 5 Pantalla Ejecutar Actividad

En ambas pantallas se muestra un botón llamado **Defecto**, ver figura IV.5 al oprimir este botón invoca la pantalla *Defecto Nuevo*. Una vez que se corrige, se registra el tiempo total que se llevó en corregir el defecto. El tiempo total, se suma al tiempo efectivo de la actividad que detectó dicho defecto.

Finalmente, si se desea puede generar un reporte en pantalla o en impresión de proyecto (despliega un listado general de todas las actividades que lo integran y el tiempo que se tardó en hacer cada una de ellas, etc), de actividad (despliega un listado de las actividades que integran un proyecto seleccionado), defecto (despliega un listado de todos los defectos que se encontraron en una actividad, si estos fueron o no corregidos) o por fecha.

Para crear los reportes se usó el *Patrón de Diseño Composite*, el cual facilitó el proceso de elaboración de los diferentes reportes que el sistema proporciona. Este patrón utiliza una clase abstracta cuyas clases concretas que le heredan se utilizan para construir el objeto a través de una clase *Composite*, la cual puede utilizarse recursivamente para construir a su vez objetos compuestos.

Los otros dos módulos *Usuario* y *Ayuda* son independientes, porque el *Módulo Usuario* solamente será utilizado por el usuario normal, cuando desee cambiar su contraseña, o bien por el usuario administrador, cuando elimine o cree una cuenta. El *Módulo Ayuda*, sirve de apoyo, cuando el usuario tiene alguna duda relacionada con un concepto y funcionalidad del sistema (este módulo no fue terminado).

La figura IV.6. muestra el diagrama de módulos y las opciones que los componen. Cada uno de estos módulos realiza una función específica, la cual se explicará en el punto de *Funciones del SCPS*, pero enfocadas principalmente al Diseño de Pantallas.

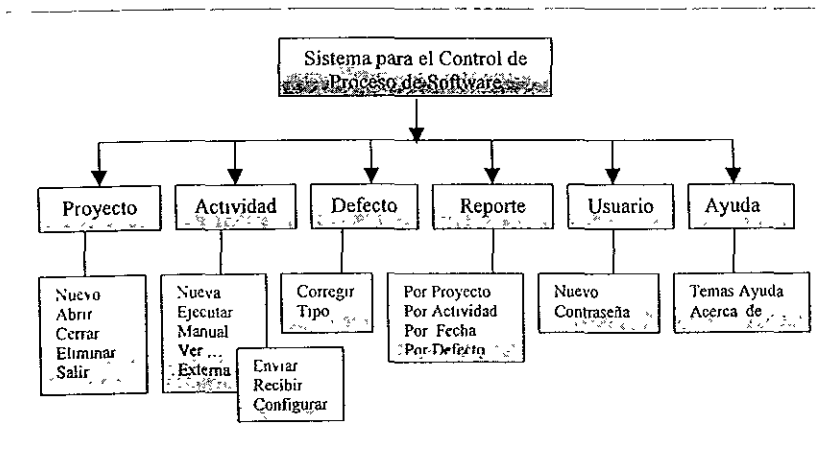


Figura IV.6 Diagrama de Módulos del SCPS (versión 1.0)

IV.3.3 Diseño de la Base de Datos

Para el Diseño de la base de datos se siguió la metodología de Coad/Yourdon, la cual menciona al componente de datos como el proveedor de la infraestructura para el almacenamiento y recuperación de objetos desde un sistema administrador de datos. Para aislar el impacto del esquema de administración de datos se decidió usar un sistema administrador de base de datos relacional.

Lo primero que se hizo, fue hacer un mapeo de clases de objetos persistentes a relaciones, para ello se apoyaron en la metodología OMT, que dice que por cada clase del dominio del problema identificada como persistente se genere una relación, se identifique si es un dato nulo, identificar el dominio al que pertenece. si tiene llave primaria y cuál es, y los atributos a través de los cuales se pueden hacer accesos frecuentes. La figura IV 7 ejemplifica el mapeo de la tabla *Project*, se aplica para las otras tablas.

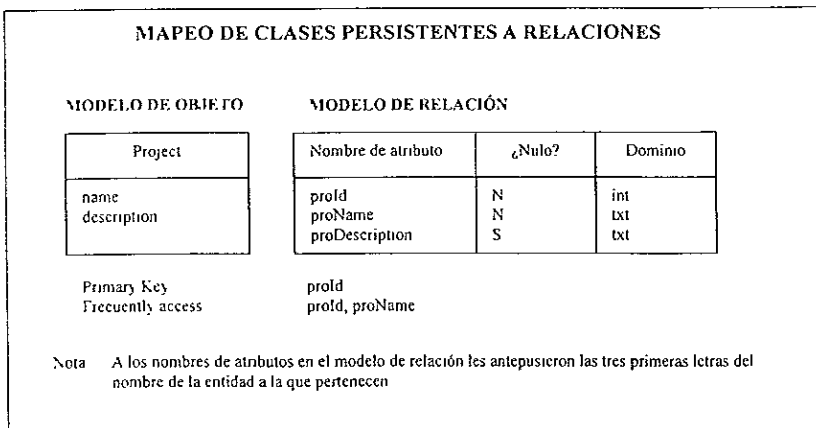


Figura IV 7 Mapeo de la tabla *Project*

La base de datos que se usó es "gisoo97.mdb", se elaboró en access97. Esta base se encuentra integrada por las siguientes tablas:

- Project
- Activity
- ActivityLine
- Defect
- DefectType
- User

La figura IV 8 muestra el Modelo Entidad Relación del Sistema para el Control del Proceso de Software, versión 1.0. En esta figura se observan las entidades con sus atributos y las relaciones con su multiplicidad.

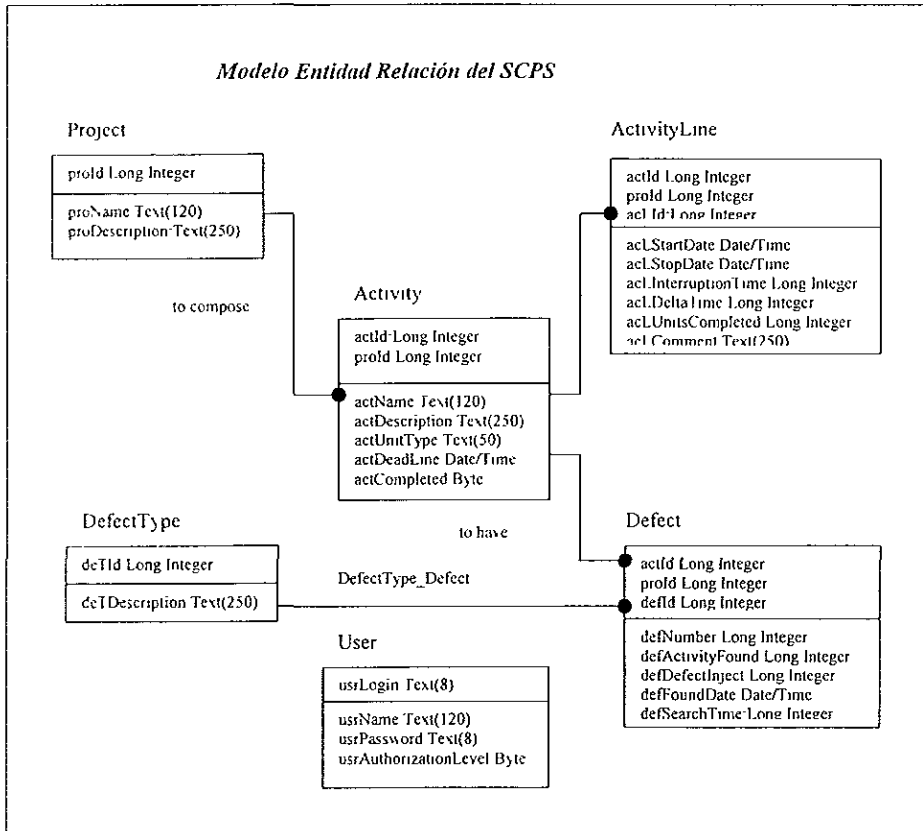


Figura IV 8 Modelo Entidad Relación

IV.4 Funciones del SCPS

El Sistema para el Control de Proceso de Software está dirigido a todas aquellas personas que desean mejorar su manera de trabajar, reducir tiempos en actividades similares y planear mejoras. Este sistema se puede utilizar en la administración y desarrollo de grandes proyectos, por ejemplo: el desarrollo de un sistema de información, el cual tendrá como mínimas seis actividades las cuales podrían ser análisis, diseño, implementación, pruebas, liberación y mantenimiento, éstas a su vez podrían estar divididas en subactividades y así sucesivamente. Cada actividad podría ser realizada por una persona que usaría el SCPS. Un líder de proyecto (responsable), llevará el control y la administración del proyecto. También se puede usar en la administración de pequeños proyectos o actividades (individuales), que a la mejor solo tendrá una actividad, por ejemplo: la realización de la tarea de cálculo integral, la cual tendrá esa actividad como tarea.

En sí, el SCPS se puede aplicar a cualquier tipo de proyecto o actividad, sin importar si es complejo. lo importante es que proporcione los tiempos que se necesitan para llevarlo a cabo, identificar las principales actividades que integran el plan del proyecto, registrar los tiempos dedicados a cada actividad, registrar el tamaño del producto y registrar los tiempos en localizar y corregir defectos, para obtener como resultado final los tiempos efectivos del proyecto.

Una vez que conocemos la funcionalidad general del sistema, describiremos las funciones principales de cada uno de los módulos que integran el sistema, como se observó en la figura III.6

Módulo Proyecto

1. **Nuevo.** Permite registrar un nuevo proyecto en la base de datos. Los datos solicitados son:
 - Nombre del proyecto
 - Descripción del proyecto
2. **Abrir.** Permite seleccionar, de una lista existente, el proyecto que se desea abrir de la base de datos y trabajar sobre las opciones de los módulos de actividad, defecto o reporte
3. **Cerrar.** Permite cerrar el proyecto que está presente en memoria, sin cerrar la sesión del sistema
4. **Eliminar.** Permite eliminar de la base de datos el proyecto seleccionado, incluyendo sus actividades, registros de tiempos y de defectos.
5. **Salir.** Permite salir del sistema y regresar el control al sistema operativo.

Módulo Actividad

1. **Nueva.** Permite registrar una nueva actividad en la base de datos de un proyecto previamente seleccionado. Los datos a registrar son:
 - Nombre de la actividad
 - Unidad que se va a utilizar para medir el avance de la actividad
 - Fecha de término de la actividad
 - Descripción de la actividad

Estos datos no quedan presentes en memoria. Pero son almacenados en la base de datos, solo si son aceptados.

2. **Ejecutar.** Permite cronometrar automáticamente el tiempo empleado en ejecutar una actividad en la base de datos, es decir, al iniciar la ejecución de la actividad registra la fecha y hora de inicio, si durante la ejecución hay interrupciones se puede registrar el tiempo de cada una de ellas. Finalmente cuando termina y se obtiene el producto esperado, se calcula el tiempo efectivo de la actividad.

3. **Ejecutar en el botón defecto.** Permite registrar un defecto cuando la actividad está en pausa, para ello abre una ventana llamada defecto nuevo. La cual pide los siguientes datos:

- Actividad que detectó el defecto
- Actividad que inyectó el defecto
- Actividad que corregirá el defecto
- Defecto que se introdujo
- Fecha de detección del defecto
- Hora de detección del defecto
- Tiempo de búsqueda del defecto
- Tipo de defecto
- Descripción del defecto

Si los datos son aceptados quedarán registrados los datos del nuevo defecto.

4. **Manual.** Permite capturar manualmente los registros de tiempos para las actividades que no se dieron de alta con él cronometro, los datos solicitados son:

- *Número de unidades producidas para la actividad*
- Fecha de inicio de la actividad
- Fecha de término de la actividad
- Hora de inicio de la actividad
- Hora de término de la actividad
- Tiempo efectivo de la actividad
- Tiempo de interrupción de la actividad
- Descripción de la actividad
- Terminada (sí es el caso)

5. **Ver.** Muestra los detalles de una actividad, incluyendo totales de tiempo y unidades producidas, así como todos sus registros de tiempo (pantalla sólo de lectura).

6. **Externo.** Permite enviar al dispositivo externo las actividades del proyecto que se desean controlar por el cronómetro (esta función no se implementó)

7. **Recibir.** Permite recibir en el sistema actividades provenientes del dispositivo externo. (esta función no se implementó)

8. **Configurar.** Permite seleccionar los parámetros de comunicación entre el sistema y el dispositivo externo. los datos son: (esta función no se implementó)

- Puerto
- Bits por segundos
- Bits por datos
- Paridad
- Bits por parada
- Tiempo de interrupción

Módulo Defecto

1. **Corregir.** Permite registrar automáticamente en la base de datos el tiempo de corrección de un defecto, es decir cuando se detecta un defecto, se crea un registro para este, el cual contendrá los datos siguientes:
 - Actividad que detectó el defecto
 - Actividad que inyectó el defecto
 - Actividad que corregirá el defecto
 - Defecto que se introdujo
 - Fecha de detección del defecto
 - Hora de detección del defecto
 - Tiempo de búsqueda del defecto
 - Tipo de defecto
 - Descripción del defecto

2. **Tipo.** Permite registrar en la base de datos un nuevo tipo de defecto. Los datos registrados son:
 - Tipo de defecto
 - Descripción del defecto

Módulo Reporte

Este módulo permite generar cuatro tipos de reportes, que son:

1. **Por Proyecto.** Genera un reporte de un proyecto existente en la base de datos, el cual contiene:
 - Nombre del proyecto
 - Fecha de impresión actual
 - Descripción del mismo
 - Nombre de todas las actividades que pertenecen al proyecto
 - Fecha en que se terminó cada una de las actividades, así como su tiempo efectivo y de interrupción.

2. **Por Actividad.** Genera un reporte con todos los detalles de una actividad seleccionada de un proyecto existente en la base de datos, el reporte es generado con los datos:
 - Nombre del proyecto
 - Fecha de impresión
 - Nombre y descripción de la actividad
 - Unidades producidas
 - Tiempo efectivo
 - Tiempo de interrupción
 - Estado actual

3. **Por Fecha.** Genera un reporte de todas las actividades de un proyecto existente en la base de datos, que inicia en un rango de fechas dadas, los datos que se generan son:
 - Periodo de fechas
 - Nombre y descripción del proyecto
 - Lista de actividades con: nombre, fecha de inicio, fecha de terminación e indicador de terminación(Esta opción no fue implementada)

4. **Por Defecto.** Genera un reporte de todos los defectos corregidos y sin corregir de un proyecto existente en la base de datos, los datos que se generan son:
 - Nombre y descripción del proyecto
 - Defectos detectados con nombre
 - Actividad en que se inyectó
 - Fecha de detección
 - Tipo de defecto
 - Descripción

Módulo Usuario

1. **Nuevo.** El administrador del sistema registrará un nuevo usuario a la base de datos. Los datos solicitados son:
 - Nombre del usuario
 - Clave del usuario
 - Contraseña del usuario

2. **Contraseña.** Permite tanto al usuario como al administrador cambiar su contraseña. Este cambio modifica la base de datos. Los datos que solicita son:
 - Contraseña
 - Confirmación de la contraseña

Módulo Ayuda

1. **Acerca de.** Muestra la ventana de presentación del sistema, con los datos principales del mismo y el nombre de los integrantes del equipo de trabajo de GISOO.

2. **Temas de Ayuda.** Presenta un índice de los temas de ayuda que el usuario puede elegir para apoyarse en la realización de su proyecto (esta opción no fue terminada).

Existen otros dos módulos que no forman parte del menú principal. Pero que se describe su función, porque forman parte del sistema.

Pantalla Presentación

1 Muestra una pantalla de bienvenida que contiene:

- Nombre del sistema
- Logotipo
- Versión
- Mensaje de Bienvenida

Pantalla Sesión

1 Solicita al usuario o administrador que introduzca su clave y contraseña. Verifica que los datos capturados sean correctos en la base de datos

IV.5 Problemas Detectados

El desarrollo de esta tesis está enfocada principalmente en la fase de mantenimiento del SCPS, pero mas específicamente en el "Diseño de Pantallas", lo cual significa que solamente se describirán los problemas detectados en esta parte.

Se partió de un reporte de defectos registrados desde el 6 de mayo al 10 de junio de 1998. Este reporte es una revisión de la interfaz del usuario y funcionalidad general del SCPS versión 1.0. Como el reporte es extenso y detallado, se elaboró un resumen general de dicho reporte que muestra a grandes rasgos la situación en la que se encontraba el Sistema. (Figura IV.9)

Del cuadro observamos que del total de defectos reportados (92) solo se corrigieron y validaron 35 de ellos, esto significa que la funcionalidad general del sistema y la interfaz humana se encuentran operando bien un 38.05%, mientras que el 61.95% no se ha corregido ni validado. (Figura IV.9)

Resumen de defectos registrados en la interfaz de usuario y funcionalidad general del SCPS (versión 1.0)										
No. Defecto	Tipo de Defecto	Serevidad			Prioridad			Estado		
		A	C	G	B	M	A	R	C	V
12	Documentación	12	-	-	12	-	-	12	3	3
21	Sintaxis	15	3	3	13	5	3	21	14	14
5	Constructor, paquete	1	-	4	1	2	2	5	2	2
8	Asignación	7	-	1	4	4	-	8	-	-
9	Interfaz	4	-	5	5	3	1	9	5	5
7	Verificación	1	3	3	2	2	3	7	3	3
24	Función	6	14	4	8	1	15	24	7	7
4	Sistema	-	1	3	1	1	2	4	1	1
2	Ambiente	1	-	1	1	1	-	2	-	-
Total	92	47	24	21	47	19	26	92	35	35

Nota	En Serevidad	En Prioridad	En Estado
	A= Advertencia	B = Baja	R = Reportado
	G = Grave	M = Media	C = Corregido
	C = Critico	A = Alta	V = Validado

Figura IV 9 Resumen General de Defectos

Partiendo de este cuadro, analizaremos cada una de las pantallas del sistema revisando la funcionalidad e interfaz de usuario, (los defectos detectados se registrarán en un formato diferente). La figura IV.10 muestra el diseño de formato de registro de defectos propuesto en este trabajo para registrar las anomalías detectadas para cada pantalla.

Registro de Defectos por Pantalla	
Registro: Fecha de revisión. Módulo: (nombre del módulo al cual pertenece la pantalla a revisar) Pantalla: (nombre de la pantalla a revisar) Revisor: (nombre o iniciales de la persona que esta haciendo la revisión)	
<i>Descripción:</i>	<i>Defectos encontrados.</i>
<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> Imagen o representación de la pantalla. (Señalando los errores a corregir) </div>	

Figura IV 10 Formato de Registro de Defectos por Pantalla

Este formato se encuentra dividido en tres partes, la primera parte solicita los datos generales de la pantalla, la segunda parte pide una descripción breve y los defectos que se detectaron y la última una representación de la pantalla que está siendo revisada, señalando los puntos a los cuales se está haciendo referencia en la parte de defectos encontrados.

El SCPS cuenta aproximadamente con 31 pantallas, se ejemplificarán en este punto tres de ellas (ver figuras IV.11, IV.12 y IV.13), las demás se encuentran en el Apéndice A “Problemas Detectados en Pantallas”

Dada la descripción y funcionalidad del sistema propuesto, en el cual se identificaron defectos que no fueron detectados por el personal encargado de efectuar las pruebas durante el desarrollo de este sistema. En el siguiente capítulo, se aplicará la metodología al sistema original para ver de que manera se puede corregir las fallas detectadas y observar las mejoras que pudieran darse para incrementar la calidad y funcionalidad del sistema.

Ejemplo N° 1 Pantalla de Sesión

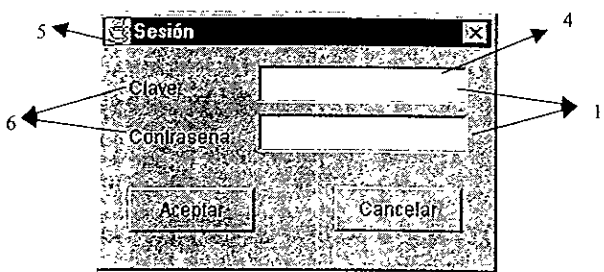
Registro de Defectos por Pantalla	
Registro: 1 Módulo: Pantalla: Sesión Revisor: Rita Carolina Rodríguez Martínez	Fecha de revisión: 02/07/99
Descripción Solicita la clave y contraseña del usuario. Si los datos son correctos, la pantalla se cierra y abre la <i>Pantalla Principal</i> . Si los datos son incorrectos, despliega una ventana de <i>mensaje de error</i> y regresa a esta pantalla para que el usuario reintente de nuevo (solo acepta tres intentos). Si ninguno es correcto cierra el sistema. Permite moverse de un campo a otro con la tecla TAB o con el botón de mouse.	Defectos encontrados <ol style="list-style-type: none"> 1 No limpia los campos, cuando el usuario se equivoca y regresa nuevamente a esta pantalla. 2 No responde a la tecla ENTER 3. El mensaje de error no es claro 4 El campo clave acepta mayúsculas o minúsculas, ejemplo: Si la clave declara como HAPARICI, y se introduce como haparici, el sistema lo acepta (campo no validado) 5 El logotipo no es apropiado 6. Los nombres de los campos son sinónimos.
	

Figura IV 11 Defectos Encontrados en la pantalla Sesión

Ejemplo N° 2. Pantalla Ejecutar Actividad

Registro de Defectos por Pantalla	
Registro: 2 Módulo: Actividad Pantalla: Ejecutar Actividad Revisor: Rita Carolina Rodríguez M.	Fecha de revisión: 02/07/99
<p>Descripción</p> <p>Primero se despliega una ventana con la lista de actividades existentes en el proyecto. Al seleccionar una actividad y oprimir el botón <i>Aceptar</i> oculta la ventana <i>Principal</i>, abre la ventana <i>Ejecutar Actividad</i> y la centra en la pantalla del monitor.</p> <p>El botón <i>></i> inicializa el cronómetro y empieza a registrar el tiempo hasta que se oprime el botón <i> </i>, el cual detiene el registro del tiempo, registrando el tiempo de interrupción.</p> <p>El botón <i>=</i> termina la actividad y abre la ventana <i>Actualizar Actividad</i>.</p> <p>El botón <i>información</i>, muestra una ventana sólo de lectura con el nombre de la actividad y del proyecto que se está ejecutando.</p> <p>El botón <i>Defecto</i> abre la ventana <i>Defecto Nuevo</i>.</p> <p>El botón <i>Cancelar</i> cierra esta pantalla y muestra la pantalla <i>Principal</i>.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1 El logotipo no es el correcto 2 Innecesario el botón <i>Información</i> y por lo tanto la ventana que llama 3. La ventana se despliega en la parte central de la pantalla del monitor. 4. No responde a las teclas Ctrl+E, cuando es invocada desde la pantalla <i>Principal</i>. 5. Si la pantalla se cierra con el botón X de la barra de título, no regresa a la pantalla <i>Principal</i>, por lo tanto se tienen que oprimir las teclas Ctrl+C para salir de sesión (se pierde el control del sistema). 6 Aparecen los botones minimizar y restaurar en la barra de título, lo cual deforma el diseño de la pantalla

Figura IV 12 Defectos encontrados en la pantalla Ejecutar Actividad

Ejemplo N° 3. Pantalla Defecto Nuevo

Registro de Defectos por Pantalla	
Registro 3 Módulo: Actividad Pantalla: Defecto Nuevo Revisor Rita Carolina Rodríguez M	Fecha de revisión: 02/07/99
<p style="text-align: center;">Descripción</p> <p>Esta pantalla es invocada por el botón Defecto de la ventana <i>Ejecutar Actividad</i>. Permite registrar un defecto cuando la actividad esta en pausa, permite moverse de un campo a otro solo con la tecla TAB o con el botón del mouse. Una vez introducidos los datos correctos, se oprime el botón <i>Aceptar</i>, los datos proporcionados quedaran registrados en la base de datos, la ventana se cierra y regresa el control a la ventana <i>Ejecutar Actividad</i> para continuar con la realización de la actividad que lo invocó.</p>	<p style="text-align: center;">Defectos encontrados:</p> <ol style="list-style-type: none"> 1 El logotipo no es el correcto. 2 No se especifica el formato para los campos de <i>Fecha y Hora de Detección</i>, por lo tanto no se validan 3 Los campos de lectura no deberían ser de texto. O bien, el contenido aparecer en otro color. 4 La información solicitada en la pantalla no es clara (faltan ayudas). 5 La pantalla no responde al botón <i>Aceptar</i>, es decir no hace nada. 6 El campo <i>Tiempo de Búsqueda</i> no especifica formato (mins., hrs. o segs.) 7 No acepta la tecla ENTER 8 Los nombres de algunos campos no están acentuados.
<p>The screenshot shows a window titled 'Defecto Nuevo - [Sistema de Proceso de Software]'. It contains several input fields and buttons. Callouts are as follows: 1 points to the window title bar; 2 points to the 'Fecha de Detección' field; 3 points to the 'Actividad que Inveio' dropdown; 4 points to the 'Actividad que Corregira' field; 5 points to the 'Aceptar' button; 6 points to the 'Tiempo de Búsqueda' field; 7 points to the 'Cancelar' button; 8 points to the 'Actividad que Inveio' dropdown. Other fields include 'Actividad que Defecto' (ANALISIS), 'Defecto que Introdujo' (NINGUNA), 'Hora de Detección', and 'Tipo de Defecto' (Sintaxis).</p>	

Figura IV.13 Defectos encontrados de la pantalla Defecto Nuevo

V. Mantenimiento Aplicado al Sistema para el Control del Proceso de Software

En este capítulo, como caso práctico, se aplica la metodología para dar mantenimiento a un sistema existente, siguiendo las actividades propuestas en el capítulo tres. La finalidad de este capítulo es entonces corregir y/o eliminar los defectos detectados durante la revisión detallada que se hizo a la interfaz humana del SCPS, versión 1.0 y a la documentación del sistema. Los defectos encontrados en el capítulo anterior se registraron y se presentan en el *apéndice A*, en los *Formatos de Registro*. Para corregir estos defectos se hacen propuestas de solución y se elabora un plan de trabajo que permiten tomar decisiones las cuales de ser aceptadas mejoraran la funcionalidad y darán mayor calidad al sistema siguiendo la metodología propuesta.

V.1 Cambios Propuestos

En este punto se plantea una propuesta de cambio global que deberá realizar enfocada principalmente al "*Diseño de Pantallas*" para mejorar la interfaz del usuario y la funcionalidad del Sistema en lo referente a complementar y clarificar la información necesaria para que el usuario realice mejor su trabajo. En V.2 se propone cambios más específicos para hacer más amigable el diseño de la interfaz de cada una de las pantallas que integran el sistema y así mejorar tanto la funcionalidad de este, como brindar ayuda al usuario en su manejo.

Será aquí donde se podrá observar si la Mesa de Control de Cambios (MCC) realizó su trabajo al haber aceptado y corregido al 100% los cambios propuestos. Y si asentó la documentación a cerca de aquellos que no pudo realizar durante la realización de esta versión. Información que deberá ser utilizada por los desarrollares en futuras versiones.

De acuerdo al Registro de Defectos puestos en bitácora y al análisis aplicado a cada una de las pantallas que integran el SCPS (versión 1.0), curiosamente se identificaron defectos que a simple vista podrían corregirse sin ningún problema. Sin embargo, para corregirlos y depurarlos, era necesario proponer cambios que implicaban un análisis más profundo del sistema y para ello es recomendable y necesario realizar un plan de trabajo para revisar la documentación del sistema actual. El estudio y análisis de esta información como un todo, será de gran ayuda para tener un panorama más amplio del funcionamiento y desempeño del sistema y tener un punto de partida

La propuesta de cambio global, es la siguiente:

1. Diseñar un icono que ejemplifique al Sistema de Control del Proceso de Software, es decir una imagen que al ser observada por el usuario identifique inmediatamente la actividad que este puede realizar con el sistema.
2. Desaparecer el logotipo de Java (la taza de café) que aparece en la barra de título en todas las pantallas, a fin de no distraer al usuario de la actividad que esta realizando. Utilizar el espacio del logotipo reemplazándolo por el icono propuesto en el punto anterior
3. Uniformizar y clarificar los mensajes que el sistema despliega cuando falta un dato o cuando ocurre un error. Tratando de comunicar lo mas explícitamente posible el contenido del mensaje

- 4 Uniformizar el tamaño de las pantallas de captura, consulta, reporte y ayuda. Estas, deberán ocupar el área de trabajo de la pantalla *Principal*. Dado que esta pantalla permanece abierta la mayor parte del tiempo, se decidió incluir en la barra de título el nombre del proyecto en el cual se encuentra el usuario trabajando. Con esto se evita además de repetir en cada pantalla el nombre del proyecto el causar un overhead al sistema teniendo cada vez que se despliega buscar este dato.
- 5 Sustituir el nombre que aparece en cada uno de los botones de las diferentes pantallas por una imagen que represente la función que cada botón desempeña.
- 6 Habilitar las teclas TAB y ENTER en el teclado, como adición a los botones, a fin de poder pasar entre ellos (*Aceptar, guardar, cancelar, cerrar*) pudiendo seleccionarlos con una combinación de estas teclas a fin de que el usuario pueda trabajar con más libertad. Es decir, que si el usuario está acostumbrado a usar solo el teclado para moverse de un lugar a otro, lo haga, o en caso contrario utilice el mouse.
- 7 Diferenciar los campos de captura de los campos de consulta (solo lectura), es decir, ayudar la labor del usuario a través de asignar un color diferente a aquellos campos que el usuario debe utilizar para introducir la información solicitada por el sistema, pudiendo diferenciarlos de manera inmediata de aquellos que solamente resultan ser informativos o de despliegue. Se propone fondo azul oscuro y texto en letras amarillas oscuras.
- 8 Cambiar la presentación del mouse, por default el mouse aparece como una flecha, se propone que aparezca en forma de una mano, en aquellos objetos que permitan seleccionar información, seleccionar y oprimir un botón.
- 9 Eliminar totalmente de las pantallas de captura y consulta el nombre del proyecto que aparece en la barra de título. Por la misma razón que se mencionó en el punto 4.
10. Implementar las opciones que faltaron en algunos de los módulos del menú, en particular lo referente al módulo Defecto y Ayuda, en donde el sistema no era explícito y estaba incompleto, produciendo insatisfacción al usuario dado que faltaba información para que él pudiera realizar su trabajo adecuadamente.
11. Como resultado del punto anterior, rediseñar la base de datos complementando los campos faltantes a fin de hacer más explícita la información desplegada en las pantallas.

V.2 Aplicación de la Metodología Propuesta

Utilizando la metodología planteada en el capítulo tres sobre el “*Sistema para el Control del Proceso de Software*” y sin perder de vista que es considerado como un caso práctico; se aplicaron las propuestas de cambio mencionadas en el punto anterior con el propósito de comprobar si la metodología propuesta ayuda a mejorar el proceso de mantenimiento de este sistema. Para ello es necesario elaborar un plan de trabajo, el cual va a indicar las actividades que el programador deberá llevar a cabo para implementar los cambios propuestos.

V.2.1 Elaboración del plan de trabajo

Para definir el plan de trabajo, primeramente se debe concentrar la información pertinente del sistema, la cual consiste en documentación descriptiva de este, documentación relativa a las versiones anteriores, diagramas de flujo, manuales operativo y de usuario e información obtenida de forma directa por parte del usuario. A partir de su análisis, definir las actividades a llevar a cabo para corregir los defectos teniendo cuidado de priorizarlas y calendarizarlas, a fin de que unas no afecten a otras. En nuestro caso, para corregir los defectos que se identificaron en el capítulo anterior y que no se corrigieron en el producto inicial.

La figura V.1 muestra el plan de trabajo que sirvió de base para realizar la nueva versión del SCPS.

Es importante hacer notar que en el tema *Problemas Detectados* del capítulo tres forma parte del plan de trabajo como resultado del análisis de la información.

Plan de Trabajo					
Actividad	Descripción	Recursos Requeridos	Responsable	Fecha de inicio	Fecha estimada de terminación
Revisión detallada de SCPS como es Diseño Datos y Funcionalidad (Revisión de la interfaz humana)	<ul style="list-style-type: none"> • Observar si la pantalla abre y cierra correctamente • Si al cerrar y abrir cada pantalla va y regresa al lugar que la llamó. • Si los botones de salvar abren la base de datos y almacena la información sin ningún problema • Si los botones aceptan la tecla ENTER o solo el mouse • Si el tamaño de las pantallas es estandar. • Si las opciones del menú permanecen activas cuando deberían estar inactivas. • Revisar si las opciones declaradas en los módulos del menú son las apropiadas • Verificar los campos de validación • Verificar si la información solicitada en cada pantalla es entendible para el usuario y si esta información es la adecuada, o bien faltan o sobran datos • Verificar si los campos de captura se diferencian de los campos de lectura • Verificar si los mensajes de error y los mensajes de validación son explícitos. 	Sistema de Control de Proceso de Software Access Windows95	Responsable 1 Responsable 2 Responsable N	28/06/1999	02/07/1999

V. Mantenimiento Aplicado al Sistema para el Control del Proceso de Software

Revisión detallada de documentación	<ul style="list-style-type: none"> • Verificar si no hay inconsistencia en el nombre de los campos y los métodos declarados en el diccionario de clases Revisar si la documentación realizada por GISOO corresponden con el diseño y funcionalidad del sistema, es decir si el diseño refleja lo implementado • Revisar la arquitectura, y ver si el lenguaje en que se desarrollo la versión 1.0 soportaría cambios • Revisar si existen inspecciones de código. 	<p>Analisis Ver. 3.0 Diseño Ver 1.0 El manual de diseño</p>	<p>Responsable 1 Responsable 2</p>	21/06/1999	25/06/1999
Proponer cambios para la creación de nuevas pantallas y funcionalidad del sistema	<ul style="list-style-type: none"> • Describir los cambios que ayudarán a mejorar la funcionalidad del sistema <ul style="list-style-type: none"> • Proponer una interfaz gráfica nueva • Manejar mensajes explicitos • Implementar las opciones que hicieron falta en algunos de los módulos del sistema • Uniformizar el código con la base de datos • Diseñar o actualizar la base de datos 		<p>Responsable 1 Responsable 2</p>	12/07/1999	31/07/1999
Presentar los cambios propuestos a la MCC	Por medio de solicitudes presentar los cambios a la MCC, quién decidirá si las solicitudes proceden o no a su implementación		<p>Responsable 1 Responsable 2</p>	02/08/1999	06/08/1999
Implementar los cambios aprobados para la construcción de la nueva versión	<p>Llevar a cabo todos los cambios que se puedan implementar para ellos es necesario</p> <ul style="list-style-type: none"> • Diseñar las nuevas pantallas (adaptando las pantallas anteriores al nuevo diseño) • Estandarizar los nombres de las pantallas • Diseñar la nueva base de datos • Liberar los cambios implementados y aquellos que por alguna razon no fue posible su implementación, documentarlos para una versión futura 		<p>Responsable 1 Responsable 2</p>	16/08/1999	27/02/2000
Probar el nuevo sistema	Aplicar cuestionarios a personas ajenas al sistema para conocer sus opiniones sobre la interfaz humana del SCPS		<p>Responsable 1 Responsable 2</p>	20/03/2000	19/03/2000
Liberar la nueva versión			<p>Responsable 1 Responsable 2</p>	13/04/2000	11/04/2000
Documentar el sistema	Generar los nuevos diagramas (aquellos que se vieron afectados con los nuevos cambios).		<p>Responsable 1 Responsable 2</p>	06/03/2000	14/04/2000

Figura V | Plan de Trabajo

Una vez que el diseño del plan de trabajo esta concluido se entrega a los administradores del proyecto de software, en donde ellos se encargan de revisarlo, evaluarlo y finalmente autorizarlo a fin de que sea llevado a cabo en forma ordena y con base a las fechas compromiso pactadas.

V.2.2 Control de Cambios

El “*Control de Cambios*” consiste en presentar a la Mesa de Control de Cambios las propuestas de solución Estas propuestas, en nuestro caso particular estuvieron enfocadas principalmente al desarrollo de Pantallas, dado que la mayoría de los defectos que se detectaron durante la revisión detallada del sistema se vieron concentrados en el Diseño de la Interfaz del Usuario.

Las propuestas de cambios se entregaron a la MCC, ideal, a través de la forma “*Solicitud de Cambio*”, (en línea). La Mesa de Control de Cambios ideal estuvo conformada por la Dra. Hanna Oktaba, M. en C. Guadalupe Ibarguengoitia, Ing. Carolina Flores y Lic. Rita Rodríguez. La función principal de la Dra. Hanna y M. en C. Guadalupe fue la de supervisar la implementación del sistema y ayudar a la toma decisiones en caso muy especiales. La MCC entonces se encargó de evaluar y dictaminar si la solicitud procedía o se cancelaba. Si la solicitud era aprobada, se procedería a su implementación y a la prueba de los cambios implementados a fin de verificar que no repercutieran en otra parte del sistema Una vez que no fueran detectadas anomalías, se llevaría a cabo la liberación de los cambios.

El procedimiento anterior se aplicó a cada una de las solicitudes. Una vez obtenida su aceptación, se procedió a la liberación del nuevo sistema. Aquí, es conveniente hacer notar que el resultado obtenido en el caso estudiado, fue la implementación de un nuevo sistema dado que el mantenimiento del sistema nos llevo a crear una nueva implementación utilizando un diferente lenguaje y a la modificación del diseño de las pantallas del primer sistema y al rediseño de la base de datos

En esta parte solo se ejemplifica una pantalla, la primera que se uso en el capítulo III “*Registro N°1 de Defectos Encontrados*”. Se decidió usar el primer registro para darle seguimiento, tener en cuenta cuáles fueron los cambios que sufrieron, y si fue posible llevar a cabo todos los cambios o solo parte de ellos y por último probar si estos cambios mejoraron la calidad y funcionalidad del sistema

La primera solicitud que se presentó fue la "Pantalla Sesión".

Solicitud de Cambio	
Solicitud N° <u>1</u>	Fecha de solicitud: <u>20/09/1999</u> Hora: <u>09:30 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Sesión</u>	
Descripción del cambio:	
1 Los botones ACEPTAR Y CANCELAR se puedan habilitar las teclas TAB y ENTER en el teclado	
2 Representar las etiquetas de los botones <i>Aceptar</i> y <i>Cancelar</i> con una imagen que represente la función que desempeñan	
3 Cambiar la etiqueta <i>clave</i> por <i>login</i>	
4 Limpiar los campos <i>login</i> y <i>contraseña</i> cuando el control regresa a esta pantalla, y los datos que el usuario introdujo después de oprimir el botón <i>Aceptar</i> fueron erróneos	
5 Uniformizar y clarificar los mensajes que el sistema despliega cuando falta un dato o cuando ocurre un error	
6 Validar los campos <i>login</i> y <i>contraseña</i> .	
7 Cambiar el logotipo de Java que aparece en la barra de título, por un icono que ejemplifique al sistema	
8 Cambiar la presentación de la pantalla que esta sea más amigable para el usuario	
Razones para el cambio:	
<ul style="list-style-type: none"> • Los cambios que se proponen son para mejorar la funcionalidad del sistema • Permitirle al usuario la posibilidad de moverse de un lugar a otro no solo con el mouse sino también con el uso del teclado • Que el mensaje de error que el sistema despliega indique de una manera más clara que el usuario ALGO no ha sido dado de alta, en vez de indicarnos Usuario:ALGO inválido. • Para que la presentación sea más agradable se propone eliminar de la barra de título el nombre de la pantalla, usar imágenes en los botones en lugar de etiquetas, añadir un letrero en la parte central de la pantalla con letras mayúsculas y de color azul el nombre de la pantalla 	
<hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma del solicitante	

Figura V.2 Forma para la solicitud de un cambio

La solicitud de cambio previamente llenada se le entregó a la MCC, quién se encargó de analizarla y posteriormente autorizar la solicitud a través de la forma "Aprobación de Cambio", la cuál se muestra en la figura V.3

Aprobación de Cambio	
Solicitud N° <u>1</u>	Fecha de recepción: <u>22/09/1999</u> Hora: <u>09:30 a.m.</u>
Autorizada por: <u>La Mesa de Control de Cambios virtual</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada (<input type="checkbox"/>)
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio	
() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios	
Fecha de decisión <u>22/09/1999</u> <hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma de la MCC	

Figura V.3 Forma para la aprobación del cambio

Una vez que la MCC autorizó los cambios, hizo entrega de la solicitud aprobada a los Ingenieros de Software, quienes se encargaron de implementarlos, probarlos y detectar si no afectaban a otros componentes del sistema. Los resultados obtenidos se plasmaron en la forma “*Resultado de la Implementación del Cambio*”, mostrada en la figura V.4

Resultado de la Implementación de Cambio	
Solicitud N° <u>1</u>	Fecha de recepción <u>23/09/1999</u> Hora: <u>9.30 a.m.</u>
Ingeniero(s) de software: <u>Programador 1, Programador 2</u>	
N° total de cambios implementados en los componentes <u>5</u>	Versiones: <u>1</u>
Descripción de los cambios:	
<p><u>De los ocho cambios propuestos solo fue posible realizar cinco de ellos (1, 2, 3,5 y 8), los otros quedaron registrados en una bitácora para una versión posterior. Los tres cambios pendientes son:</u></p> <ul style="list-style-type: none"> • <u>No se pudo cambiar el logotipo de Java por el icono que representa al sistema en la barra de título de la pantalla</u> • <u>No se limpian los campos de login y contraseña cuando el usuario proporciona datos erróneos</u> • <u>No valida el campo de login, si este se declara con mayúsculas y se tecldea la primera letra con mayúscula y las demás con minúsculas y si la contraseña es la correcta, le permite al usuario la entrada a sesión</u> <p><u>Para llevar a cabo los cambios solicitados fue necesario importar la clase LogonCommand. Así como, hacer uso de las clases MinWindow, Scope, Database y RaiseError. Se genero la clase LogonWindow, las variables utilizadas en la clase LogonCommand se estandarizaron con el nombre de las variables declaradas en el nuevo sistema.</u></p>	
Fecha de inicio: <u>24/09/1999</u>	Fecha de terminación: <u>26/09/1999</u>
Firma del Ingeniero de Software	

Figura V.4 Forma resultado de la implementación de cambio

Una vez que se terminaron de implementar los cambios propuestos, se entregó a la autoridad responsable la forma “*Resultado de la Implementación del Cambio*”, quién se encargó de evaluar los cambios realizando una serie de pruebas para verificar si efectivamente los cambios implementados en el sistema fueron satisfactorios. Los resultados obtenidos de su análisis al aplicar las pruebas los plasma en la forma “*Liberación del Cambio*”, ver figura V.5.

Liberación de Cambio

Solicitud N° 1 Fecha de recepción 27/09/1999
Hora: 05:30 p.m.

Autoridad responsable Seguridad y Calidad

Comentarios
Se decidió liberar el cambio, pero no fue posible implementarlo en su totalidad, pero quedo registrado y documentado como parte de la carpeta Prosoft para la creación de una nueva versión o un nuevo sistema, según sea el caso.

Fecha de liberación: 28/09/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Figura V.5 Forma para la liberación de cambios

A esta forma se le anexo la nueva pantalla, en la cual se señalan los cambios que se implementaron, ver figura V.6.

Esta imagen muestra como quedo la Pantalla Sesión una vez efectuados los cambios presentados a la MCC por medio de la Solicitud de cambio N° 1

Los cambios efectuados fueron

- 1 Los botones Aceptar y Cancelar se implementaron para que acepten la tecla Enter y TAB y se cambio los nombres de los botones por imagenes que representan la función que ellos realizan
- 2 Se cambio el nombre de la etiqueta Clave por Login
- 3 Se agrego en la parte superior una etiqueta de color azul en la cual se despliega la función que desempeña esta pantalla
- 4 Se elimino el nombre de la pantalla de la barra de titulo y en su lugar aparece el nombre del grupo de desarrollo

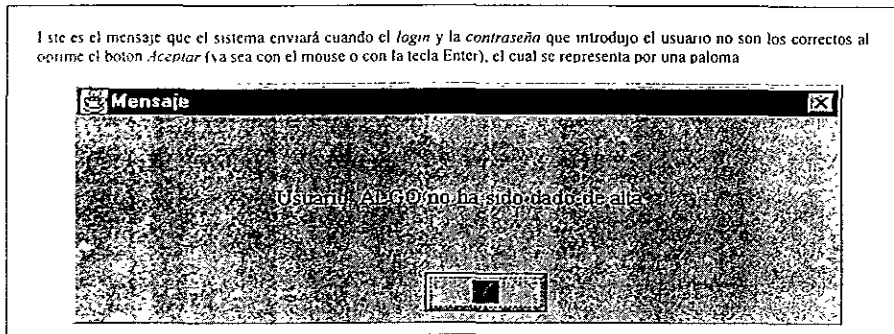


Figura V 6 Anexo de la forma liberación del cambio

En el apéndice B “Formas para las propuestas de cambio” se muestran cada una de las solicitudes por pantalla presentadas a la MCC para implementar y liberar los cambios aceptados para la construcción del nuevo sistema.

V. 2.3 Construcción de la nueva versión o del nuevo sistema

En este punto, el ingeniero de software, primeramente evaluó los cambios solicitados y aprobados, y verificó de que manera podían afectar al sistema original. Es decir, analizó si los cambios sólo afectaban una parte del sistema y si podían ser implementados con solo actualizar la versión actual del programa. Una vez que terminó de evaluar los cambios, pudo decidir, ya sea realizar una nueva versión o construir un nuevo sistema. Dado que los cambios solicitados implicaban el uso de una herramienta más potente, en la cual se pudieran importar bibliotecas, manejar interfaces gráficas para el diseño de ventanas, reutilizar código, etc. Y también, era necesario rediseñar la base de datos y diseñar las ventanas faltantes en los módulos **Defecto** y **Ayuda**, se decidió construir un nuevo sistema

Una vez tomada esta decisión, se procedió a la implementación de los cambios aprobados por la MCC, los cuales se pueden observar en la figura V.4 y en el apéndice B.

Recapitulando, para haber llegado a esta decisión fue necesario:

- a) *Realizar los cuatro primeros puntos propuestos en el Plan de trabajo*, es decir revisar y analizar la documentación del sistema y la interfaz gráfica, según se estableció en el capítulo tres, mientras que el punto cuatro “las propuestas de cambio” como se presentaron de manera general en el punto V 1 de este capítulo.
- b) *Decidir si se va a desarrollar una nueva versión del sistema o si se trata de una nueva implementación* como es el caso que estamos tratando, en el cual el sistema a mantener se encontraba escrito en JDK 1.1.4 ejecutándose bajo un sistema operativo Windows 95 y con un manejador de bases de datos Microsoft Access 3.0, decidiéndose optar por implementar un nuevo sistema bajo las siguientes especificaciones:

- Lenguaje de programación Visual Café for Java Database Developer edition 2.0 de Symantec.
- Sistema operativo Windows 98
- Manejador de base de datos Microsoft Access 97 para Windows 98

La razón de haber decidido utilizar Visual Café para Java es porque este lenguaje ofrece la compatibilidad requerida para que los cambios implementados sean mínimos, al contar con las características y facilidades que se incluyen a continuación:

- Importar a un ambiente propio el código fuente de la versión 1.0 (programas command)
- Importar las bibliotecas utilizadas en PARTS (herramienta de Java para el diseño en un ambiente gráfico)
- Reutilizar código tomando las clases, objetos y métodos del sistema original.
- Utilizar un generador de ventanas propio, lo cual facilitó el diseño de las nuevas pantallas
- La conexión con Access se logró sin ningún problema dado que se tomaron las medidas precautorias para evitar el problema del Y2K tanto en el manejador de datos como en el sistema operativo.

En cuanto al hardware utilizado para el nuevo desarrollo, no existieron requerimientos notables dado que el sistema original fue desarrollado en una PC Intel 486 con un disco duro de 1.2 GB con una memoria RAM 16 MB con un reloj de 100 Mhz; mientras que el nuevo sistema se desarrolló en un equipo con las siguientes características y especificaciones:

- Una computadora AcerPower8000 con un microprocesador Intel Inside Pentium II
- Disco duro 4.2 GB
- 32 MB RAM
- 350 Mhz

c) *Diseñar las nuevas pantallas.*

Al haber revisado los requerimientos del nuevo sistema, se procede a analizar las pantallas de las interfaces gráficas del diseño original para adaptarlas al nuevo diseño de tal manera que en nuestro caso se adaptaron a los cambios propuestos a la MCC, según se muestran en la figura V.1 y en el apéndice B. A continuación se diseñaron las nuevas pantallas que no existían en el sistema versión 1.0 y que impactaban principalmente al módulo **Defecto**, opción *Corregir*. El análisis indicaba que esta opción realmente no desempeñaba la función para la cual se suponía había sido creada en el sistema versión 1.0.

Las pantallas que no se contemplaron en la versión anterior y que fueron diseñadas en esta nueva versión son:

1. En construcción
2. Eliminar un usuario
3. Abrir defecto
4. Ejecutar un defecto
5. Registrar automáticamente un defecto
6. Registrar manualmente un defecto
7. Registro detallado del defecto

Con estas pantallas se logro implementar la funcionalidad del módulo Defecto.

d) *Estandarizar el nombre de las pantallas*

El siguiente paso, una vez diseñadas las nuevas pantallas y a fin de lograr un estándar en la nomenclatura utilizada procedimos a estandarizar los nombres de los objetos de cada una de ellas

El nombre asignado a cada objeto y pantalla fue seleccionado para tener una relación directa, fácil de recordar con el nombre de la clase, del archivo, del módulo y de la opción del módulo. De manera que si alguien desean revisar, actualizar o modificar esta versión lo pueda hacer fácil y rápidamente, porque podrá identificar por pantalla a que clase y tabla se está haciendo referencia Solo ejemplificaremos dos pantallas, pero el esquema que se utilizó es el mismo para todas las demás, ver figura V.7.

<i>Nuevo proyecto</i>		
Nombre de la pantalla		NewProjectWindow
Nombre de la clase:		NewProjectCommand
Tabla:		Project
Módulo:		Proyecto
Opción		Nuevo
Nombre que se les asignó a los campos que integran la pantalla		
NewProjectNameL		newProjectNameT
NewProjectResponsibleL		newProjectResponsibleT
NewProjectLocationL		newProjectLocationT
NewProjectStartDateL		newProjectStartDateT
NewProjectEndDateL		newProjectEndDateT
NewProjectDescriptionL		newProjectDescriptionTA
NewProjectSaveB		newProjectCancelB
<i>Nueva Actividad</i>		
Nombre de la pantalla		NewActivityWindow
Nombre de la clase:		NewActivityCommand
Tabla:		Activity
Módulo:		Actividad
Opción		Nueva
Nombre que se les asignó a los campos que integran la pantalla		
NewActivNameL		newActivNameT
NewActivResponsibleL		newActivResponsibleT
NewActivProduceToUnitL		newActivProduceToUnitT
NewActivStartDateL		newActivStartDateT
NewActivLimitDateL		newActivLimitDateT
NewActivDescriptionL		newActivDescriptionTA
NewProjectSaveB		newProjectCancelB

Figura V 7 Esquema de nombre por pantalla

c) Diseñar la nueva base de datos

El siguiente paso natural fue el relativo a la revisión de los datos para efectuar una migración transparente de ellos del sistema original al nuevo sistema. Dado a que los cambios aprobados afectaban el diseño de la base de datos de la versión anterior, se decidió diseñar una nueva base de datos llamada "PSP99". La figura V.8 muestra la base de datos para el nuevo sistema.

Los cambios que se efectuaron a la base de datos original y que están incluidos en la PSP99 se describen a continuación:

- Cambiar la nomenclatura de añadir al principio del nombre del campo las tres primeras letras del nombre de la tabla a la que pertenecen, por el nombre completo del campo, dejando la asignación de la tabla en un segundo lugar, buscando una mayor claridad para el futuro mantenimiento del sistema. Dado a que la excepción hace la regla mantuvimos en la tabla User la nomenclatura original.
- Eliminar y/o añadir campos en la nueva base de datos según la información solicitada.
- Estandarizar el nombre de los campos de cada tabla de acuerdo a la función de las pantallas que invocan a dichas tablas.
- Reducir o ampliar el tamaño de los campos según los requerimientos

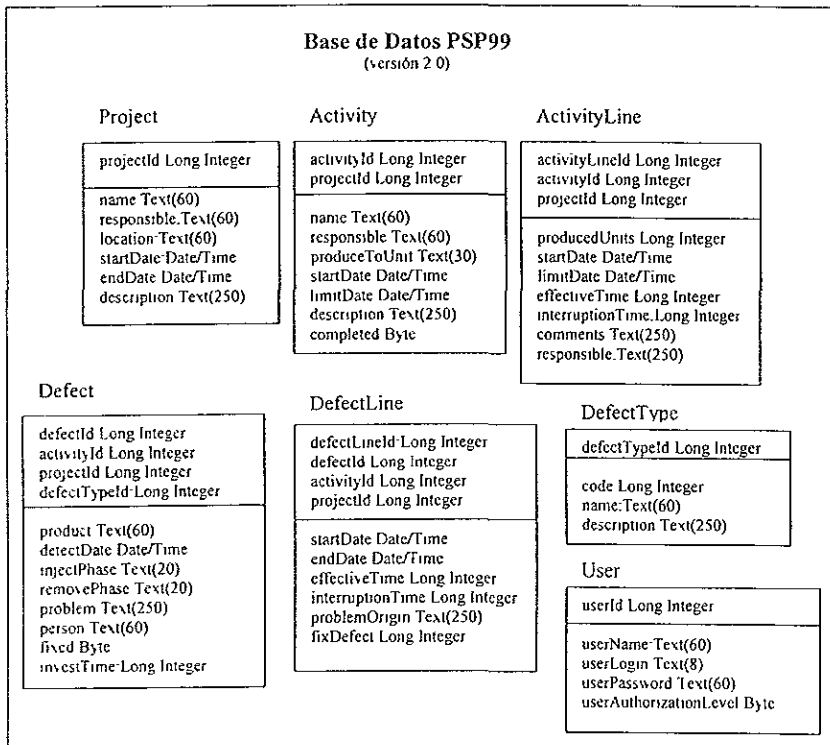


Figura V.8 Base de Datos PSP99 para la versión 2.0

1) Implementar y liberar los cambios

Conforme se iban implementados los cambios requeridos, el nuevo código se sometían a las pruebas de funcionamiento necesarias, a fin de saber si dichos cambios estaban cumpliendo el objetivo por el cual se habían sido propuestos. Verificando en cada paso, si el cambio implementado no afectaba alguna otra parte del nuevo sistema tanto en su funcionamiento como en su funcionalidad.

De esta manera, pudimos estar seguros que el sistema no fallaría en el momento de ponerlo en producción, es decir cuando fuera entregado a usuarios que no estuvieran familiarizados con él. Siendo esta la prueba crucial de desempeño del nuevo diseño. Si las pruebas resultaban favorables se liberaba el cambio, la figura V.6 y el apéndice B muestra el resultado que se obtuvo después de implementar y liberar los cambios propuestos. No obstante, el cuidado mantenido durante el desarrollo del nuevo sistema hubo algunos cambios que afectaron a otras pantallas, así hubo la necesidad de implementarlas en forma paralela. Posteriormente, se verificó el buen funcionamiento de los nuevos cambios procediendo a su liberación.

V. 2.4 Probar la nueva versión o el nuevo sistema

El objetivo de probar el producto es detectar cualquier anomalía antes de liberarlo al usuario, entendiéndose por usuario el ambiente de producción.

Una vez que los cambios implementados fueron liberados por el ingeniero de software, el sistema se puso a prueba para analizar primeramente si funcionaba correctamente y cumplía con las necesidades de los usuarios, en segundo lugar si la interfaz humana era grata, entendible y si proporcionaba la ayuda necesaria cuando se llegara a presentar una duda en el manejo del sistema. Por último, si era fácil de operar y si la información solicitada en cada pantalla era entendible para cualquier usuario, aún no estando familiarizado con el sistema.

Para decidir si el nuevo sistema estaba listo para su liberación, era necesario contar con información cualitativa y cuantitativa, así que se les solicitó a diferentes usuarios operaran el sistema y contestaran dos cuestionarios, en donde nos dieran a conocer su opinión acerca tanto de la funcionalidad y manejo como de la presentación de la información proporcionada por este (interfaz humana) Las figuras V 9 y V.10 muestran los cuestionarios requisitados.

**Cuestionario para evaluación de la interfaz humana del
"Sistema para el Control del Proceso de Software"**

Por favor realice la lista de tareas. Siéntase con la libertad de preguntar cualquier duda antes o durante la ejecución de las mismas. El objetivo es evaluar el diseño de la interfaz y su desempeño al llevar a cabo las siguientes tareas:

- a) Crear un proyecto nuevo
- b) Crear una actividad
- c) Medir el tiempo en realizar la actividad que dio de alta, por ejemplo revisar los correos de Internet
- d) Cambiar su password
- e) Consultar la actividad medida
- f) Registrar un defecto de una actividad, por ejemplo que no hay comunicación de la red para revisar los correos de Internet.
- g) Imprimir un reporte de proyecto
- h) Registrar el tiempo que le llevo arreglar que la conexión de la red estuviera trabajando correctamente para revisar sus correos de Internet.

¡Muchas gracias por su cooperación!

Figura V 9 Cuestionario N° 1 aplicado al usuario

**Cuestionario para evaluación de la interfaz humana del
"Sistema para el Control del Proceso de Software"**

Por favor conteste el siguiente cuestionario de acuerdo a sus experiencias en el uso de la interfaz. Las respuestas proporcionadas servirán para cubrir defectos de diseño y mejorar la interacción usuario – computadora. En la columna de comentarios puedes agregar cualquier información que considere importante.

a) Marque el recuadro correspondiente a su opinión para cada pregunta:

Pregunta	SI	NO	Comentarios
¿Los términos usados en la interfaz son familiares para usted?	X		
¿Considera adecuada la forma en que se presenta la información?	X		
¿Disfruto del uso de la interfaz?	X		
¿Considera complicada la navegación entre las diferentes vistas?		X	

Continúa en la otra hoja

a) Califique la interfaz de acuerdo a la escala presentada, donde 5 y 1 significan la mayor y menor puntuación respectivamente. Puede usar valores intermedios para expresar opiniones intermedias.

	5	4	3	2	1	0	
Fácil de usar	X						Difícil
Simple de aprender		X					Complicado
Funcionalidad clara			X				Confusa
Eficiente para desplegar información	X						Deficiente
Respuesta rápida	X						Lenta
Método de navegación eficiente	X						Deficiente
Apariencia estética adecuada (Colores, tipo letra, etc)	X						Inadecuada

c) ¿Que es lo que más le gusta de la interfaz?

El menú de ayuda

d) ¿Qué es lo que más le desagradó de la interfaz?

Al abrir un proyecto, se quedan abiertas las ventanas anteriores

e) ¿Cuáles son las características de la interfaz que encontró confusas o difíciles de entender?

Las imágenes que aparecen en los botones (no tienen una etiqueta de la acción a ejecutar).

f) ¿Qué cambios cree que son necesarios?

Los mencionadas en el punto d y e

g) Agregue cualquier comentario que crea útil para mejorar la interfaz.

¡Muchas gracias por contestar este cuestionario!

Figura V 10 Cuestionario N°2 Aplicado a un usuario después de manejar la interfaz humana

En el caso prueba planteado en esta tesis, una vez que se aplicaron los cuestionarios y se analizaron las respuestas, a partir de los resultados obtenidos de aceptación o rechazo, se decidió era conveniente liberar el nuevo sistema. En el caso general, deberá procederse a cuantificar y cualificar las respuestas y dependiendo de su análisis decidir si es posible la liberación o es necesario regresar a la etapa de implementación para realizar los cambios y modificaciones propuestas por los usuarios que se colaboraron a la prueba. En nuestro caso, las observaciones que hicieron los usuarios fueron mínimas y corregidas de inmediato por lo que se decidió liberar el nuevo sistema.

Dado que hablamos de un nuevo sistema, fue necesario tanto generar nuevos diagramas como adecuar otros y documentar la información que se vio afectada con los cambios propuestos, aprobados e implementados. Los documentos que se actualizaron fueron:

- Diccionarios de clases, que se encuentra integrado por:
 - Dominio del problema
 - Interfaz humana
 - Manejador de datos
- Diagramas de clase, tales como:
 - Dominio del problema
 - Interfaz humana
 - Manejador de datos
- Manual del usuario

Estos documentos se integraron en una carpeta denominada *Prosofi*, la cual reemplaza a la carpeta del sistema que fue considerado como un ejemplo para aplicar la metodología de mantenimiento, tema central de esta tesis.

VI. Resultados

A continuación se presenta la cuantificación de los resultados obtenidos en la aplicación de la metodología para mantener el Sistema para el Control del Proceso de Software.

Como un resumen general del número total de cambios presentados a la Mesa de Control, se incluye por pantalla analizada el número total de cambios sugeridos, la prioridad asignada para su implementación y una columna estado que indica si el cambio fue reportado, implementado y validado, junto con las observaciones que se derivaron de su implementación.

Esta tabla resumen incluye en la columna **Total CP**, el número total de cambios que deben ser analizados, cuya cifra debe coincidir con la columna **R** (cambios reportados). Mientras que las columnas **I** (implementado) y **V** (validado) permitirán tener el comparativo del sistema anterior con el nuevo sistema en cuanto a la funcionalidad y calidad del producto deseado. Ver figura VI.1.

Resumen de cambios propuestos e implementados en la interfaz humana y funcionalidad general del SCPS (versión 1.0)

Nombre de la pantalla	Total CP	Prioridad					Estado			Observación
		C	MI	Imp	Int	Inop	R	I	V	
Abrir proyecto	2			X			2	2	2	
Acerca de	2				X		2	2	2	
Actividad Nueva	6			X			6	5	5	No se pudo cambiar el logotipo de Java por el logotipo propuesto de la barra de título
Actualizar Actividad	10			X			10	10	10	
Actualizar Manualmente Actividad	10			X			10	10	10	
Ayuda	3	X					3	3	3	Se diseño e implemento
Configurar Puerto	4			X			4	4	4	Cronometro externo en construccion
Defecto Nuevo	7			X			7	6	6	Se hicieron 4 cambios adicionales
Ejecutar Actividad	7			X			7	6	6	Se elimino el botón X de la barra de título
Eliminar Proyecto										Se uniformizo el diseño a las demas pantallas
Enviar Actividad	4			X			4	4	4	Cronometro externo en construccion
Información						X				Se elimino, la información proporcionada se añadió en la pantalla Ejecutar Actividad
Inicio	5			X			5	4	4	No se pudo eliminar la barra de título
Mensaje	4			X			4	3	3	No se pudo cambiar el logotipo de Java por el logotipo propuesto de la barra de título
Modificar Contraseña	5			X			5	5	5	Invaldar el campo password para el administrador
Nuevo Tipo de Defecto	3			X			3	3	3	Se cambio definitivamente el diseño, hubo la necesidad de implementar las pantallas que faltaron diseñar (*)
Opciones										Se uniformizo el diseño a las demas pantallas
Principal	5			X			5	4	4	No se pudo cambiar el logotipo de Java por el logotipo propuesto
Proyecto Nuevo	8			X			8	7	7	No se pudo cambiar el logotipo de Java por el logotipo propuesto de la barra de título
Recibir Actividad	4			X			4	4	4	Cronometro externo en construccion
Requisito Detallado de la Actividad	13			X			13	12	12	No se pudo cambiar el logotipo de Java por el logotipo propuesto de la barra de título

Reporte (previo)								Se adapto a cada tipo de reporte y se cambio el diseño
Reporte por Actividad	4		X		4	4	4	
Reporte por Defecto			X					Se diseño e implemento (*)
Reporte por Proyecto	4		X		4	4	4	
Selección de Actividad			X					Se cambio el nombre por Abrir Actividad, el diseño se adapto a las demás pantallas
Selección de Defecto			X					Se cambio el nombre por Abrir Defecto, el diseño se adapto a las demás pantallas
Sesión	8		X		8	5	5	No se pudo cambiar el logotipo de Java por el logotipo propuesto en la barra de titulo de la pantalla No se limpian los campos de login y contraseña cuando el usuario proporciona los datos, estos son erroneos. No valida el campo de login, si el login se declara con mayúsculas y se teclea la primera letra con mayúscula y las demas con minúsculas y la contraseña es la correcta. I.e. permite la entrada de sesión al usuario
Usuario Nuevo	4		X		4	4	4	
Total	122				122	114	114	

Nota	Total de	En Prioridad	En Estado
CP	= Cambios Propuestos	C - Critico	R - Reportado
		MI - Muy Importante	I - Implementado
		Imp - Importante	V = Validado
		Int - Interesante	
		Inop - Inoportuno	

Figura VI I Resumen de Cambios Propuestos

En la tabla resumen, en particular en la columna observación se indica con un (*) los módulos en donde existió la necesidad de implementar y diseñar las pantallas faltantes. Un ejemplo se encuentra en la denominada "opción *Corregir*", del módulo Defecto, la cual no fue realmente implementada en el sistema anterior. Esta fue desarrollada como parte del nuevo sistema corrigiendo uno de los problemas cuya existencia es frecuente en los desarrollos que se encuentran en la practica profesional, el cual consiste en que el programador por salir del paso o cumplir con un requerimiento únicamente da título o nombre o rotulo a una subrutina o módulo sin posteriormente desarrollar código alguno para dar la funcionalidad o resolver el problema enunciado.

Por lo anterior, hubo la necesidad de implementar y diseñar las pantallas faltantes a fin de que la funcionalidad enunciada cumpliera con el objetivo por el cual fue propuesto en el sistema original.

Las pantallas faltantes que se diseñaron e implementaron en el módulo Defecto para la nueva versión fueron:

1. Ejecutar un defecto
2. Registrar automáticamente un defecto
3. Registrar manualmente un defecto
4. Registrar el defecto encontrado
5. Registro detallado de un defecto

Se construyó la pantalla *Tópicos* del módulo Ayuda (help), porque en el sistema original solo desplegaba los temas a tratar como si fuera un índice bibliográfico, pero la realidad es que no proporcionaba ninguna clase de ayuda al usuario, mostrando una figura estática de la cual al seleccionar cualquiera de sus opciones no se obtenía resultado alguno.

Para solucionar el problema explicado en el párrafo anterior se construyó una interface gráfica denominada “Ayuda de Prosoft” a la cual se le dio la funcionalidad necesaria para permitir al usuario, a través de una carpeta de archivo, poder seleccionar información de ayuda de cada uno de los módulos del Sistema para el Control del Proceso de Software, incluyendo en ello: *Proyecto, Actividad, Ejecutar actividad, Defecto, Corregir defecto, Reporte y Usuario*

Además, se diseñaron e implementaron las pantallas:

1. Eliminar usuario

Se diseñó una tabla de administración de usuarios. En el sistema anterior el administrador del sistema tenía la opción de dar de alta a usuarios, quienes se encargarían de elaborar uno o más proyectos pero no contaba con una opción que le permitiera depurar su tabla de usuarios, dado el caso, en que un usuario abandonaba su proyecto, o bien lo terminaba o no hacía más uso de él. Es por eso que se decidió diseñar una pantalla, la cual le va a permitir al administrador del sistema llevar un buen control de los usuarios, dar de baja aquellos que no hacen uso del sistema o aquellos que se lo solicitan. Esta pantalla cuenta con una restricción el administrador podrá cambiar n veces su contraseña pero no podrá eliminar su cuenta.

2. En construcción

Esta pantalla se diseñó para indicarle al usuario que el proceso de construcción del sistema continua.

El nuevo sistema, al cual decidimos dejarle el mismo nombre de “Sistema para el Control del Proceso de Software (SCPS), y lo bautizamos como versión 2.0 se encuentra integrado ahora por 40 pantallas. La razón por la cual se mantuvo el nombre es porque SCPS es ya conocido por la comunidad de la maestría en Ciencias de la Computación ubicada en las instalaciones del IIMAS.

Una vez, analizados los resultados de esta metodología que se aplicó como caso práctico para mantener principalmente el diseño de la interfaz humana y funcionalidad del Sistema para el Control del Proceso de Software, es decir haber llevado a cabo, como enunciamos en el capítulo cuatro, la revisión detallada tanto del sistema original en su totalidad como de la documentación que lo integra, se pudo observar que el sistema presentaba varias anomalías, entre ellas los módulos **Defecto** y **Ayuda**, los cuales no habían sido implementados, esto impedía que el usuario pudiera trabajar amigablemente con el sistema, en primer lugar porque no brindaba ninguna clase de ayuda, en segundo porque si se llegaba a detectar un defecto durante la realización de una actividad no permitía registrar de manera manual ni automáticamente el defecto encontrado y en tercero porque los dos módulos estaban enunciados pero no eran funcionales.

Con base a lo anterior, se elaboró un resumen de los defectos encontrados durante el análisis tanto en el diseño de la interfaz humana como en la funcionalidad general del sistema original (ver figura IV 9). En este resumen se mostró y se observó que de el número total de defectos que se habían reportado (para ser exactos 92), se lograron corregir y validar 35 de ellos, esto nos indica que la funcionalidad general del sistema y de la interfaz humana, antes de efectuar el mantenimiento, se encontraba operando en un 38.05%, mientras que el 61.95% sobrantes presentaba anomalías o fallas. Si bien, un buen número, se encontraban registrados en la carpeta de la documentación del sistema original, a través de realizar un análisis profundo se pudo detectar un número mayor de defectos lo cual enriqueció el análisis y objetivo de esta tesis, además de que nos permitió verificar la prioridad con la cual debían de ser corregidos dio soporte a nuestra decisión de reescribir el sistema en otro lenguaje.

Una vez aplicada la metodología propuesta para mantener el sistema y así darle solución al 61.95% de defectos reportados sobrante lo que representaría el 100%, se decidió que la mejor manera de mantenerlo era diseñar e implementar un nuevo sistema, en el cual se propusieran las mejoras de solución tanto para el diseño de la interfaz humana como para la funcionalidad del sistema. Estas propuestas se le presentaron a la Mesa de Control por medio de las solicitudes de cambio.

El número total de cambios que se le presentaron a la Mesa de Control para dar solución al 100% de los defectos encontrados fueron 122, lo que ahora constituye el 100% de modificaciones, de este total se implementaron y validaron sin ningún problema 114, los cuales se encuentran debidamente documentados en la carpeta denominada Prosoft. Este total nos indica que el sistema mejoró un 93.44% (del 61.95% que equivale el 100%) tanto de la interfaz humana como de su funcionalidad

El 6.56% de cambios propuestos restantes para completar el 100% por el momento no fue posible de implementar, dado que la documentación existente del lenguaje utilizado en esta versión no era suficientemente completa como para poder acceder algunas tablas en la forma que lo hacia el sistema original, pero quedaron registrados y documentados como parte de la carpeta Prosoft para una versión posterior o un nuevo sistema

CONCLUSIONES

Si mantener un sistema resulta difícil, desarrollar un sistema que presente un alto grado de mantenibilidad es mucho más difícil. Sin embargo, el mantenimiento es un proceso que se requiere aplicar constantemente cuando se desea hacer mejoras a un sistema o simplemente mantenerlo funcional. De antemano, sabemos que cualquier tipo de sistema está sujeto a cambios no inmediatos pero sí casi constantes.

Para satisfacer cualquier requerimiento de cambio o alguna necesidad derivada de un defecto, el ingeniero de software debe determinar qué se requiere hacer, y decidir a la vez la mejor forma de hacerlo con base a la documentación existente y con el propio sistema, a fin de satisfacer las necesidades del usuario. Para ello, es importante que cualquier persona que desarrolle sistemas tome en cuenta que los programas también deben ser debidamente documentados, ya que la documentación constituye en realidad la información necesaria para llevar a cabo y en forma correcta la etapa de mantenimiento de cualquier sistema.

Con base a lo anterior, fue necesario aplicar la metodología (SCPS) que nos permitió de manera adecuada y ordenada, es decir bajo el seguimiento de formas de registro, dar mantenimiento a un sistema ejemplo, apoyándonos en la planeación y ejecución de plan de trabajo determinado.

Nos queda claro que antes de mantener un sistema, es necesario entender el concepto de mantenimiento, en sus diferentes clasificaciones y la necesidad de efectuar las actividades de la *Administración de la Configuración del Software* para poder obtener productos de calidad, a través del método del Proceso Personal del Software.

Otra conclusión derivada de nuestro análisis es que hay que tener calidad para producir productos de calidad, es decir sin defectos. Así mismo, que para tener sistemas que no fallen debemos de producir software con el menor número de defectos posibles. Por otra parte es necesario apuntar que *la ingeniería y la reingeniería de software no necesariamente es aplicada por ingenieros.*

La metodología seguida en este trabajo para dar mantenimiento a un sistema, nos permitió visualizar las diferentes propuestas de solución. Inclusive llegar a la decisión de realizar un nuevo sistema *Planear* el trabajo fue un proceso en el que aplicamos mayor tiempo del cual obtuvimos los mejores resultados, dado que pudimos detectar aquellos cambios que nos permitieron mejorar la calidad y funcionalidad del sistema. La *calendarización* obtenida de una buena planificación nos permitió efectuar el trabajo de acuerdo a lo planeado pudiendo cumplir con los tiempos estimados de inicio y terminación. El poder arribar a propuestas concretas de solución e implementarlas una vez evaluadas por una Mesa de Control de Cambios virtual, nos permitió conocer las particularidades a las que el ingeniero de software debe enfrentarse antes de poder desarrollar, probar y liberar los cambios implementados.

Si bien la metodología aplicada nos ha permitido recorrer el proceso completo, esta no infería la relación necesaria entre el ingeniero de software y el usuario final, dado que probar el sistema con diferentes usuarios que no se encuentren familiarizados con este, requiere de la elaboración de cuestionarios específicos y de su explicación, a fin de que cada usuario desde su nivel nos de a conocer su opinión con relación al sistema.

La correcta aplicación de la metodología planteada nos permitió de manera efectiva mejorar el funcionamiento del sistema estudiado, inclusive detectar la necesidad de reemplazarlo. Cabe aquí hacer mención de que el proceso de mantenimiento es infinito, así que con seguridad una nueva *revisión al sistema final* dará pie a un conjunto de mejoras que deberán realizarse en el futuro. Confirmándose que lo perfecto es enemigo de lo bueno.

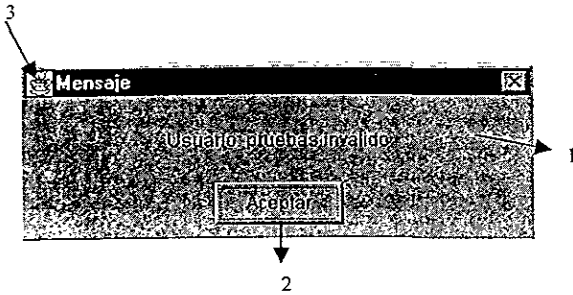
APÉNDICE A

FORMATOS DE REGISTRO DE DEFECTOS POR PANTALLA

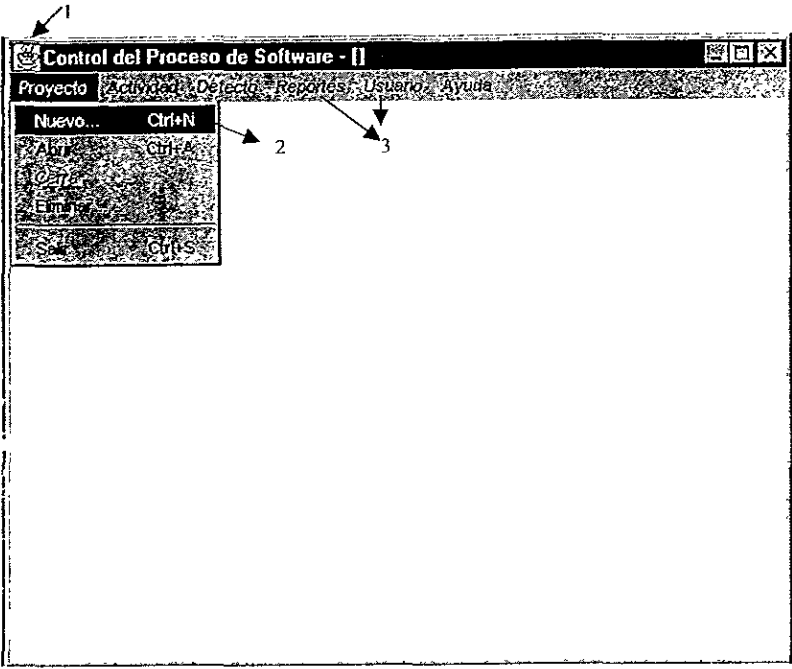
Ejemplo N° 4. Pantalla de Presentación

Registro de Defectos por Pantalla	
<p>Registro. 4 Módulo Pantalla Presentación Revisor Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción.</p> <p>Muestra el logotipo y el nombre del sistema, la versión y un mensaje de bienvenida. Esta pantalla permanece activa durante 5 segundos, después de ese tiempo se cierra y cede el control a la <i>Pantalla de Sesión</i></p>	<p>Defectos encontrados</p> <ol style="list-style-type: none"> 1 Cuando la pantalla se despliega aparece la barra de título con los botones de <i>maximizar</i>, <i>minimizar</i> y <i>cerrar</i>. 2 El logotipo que utilizan no representa al sistema, tanto el que aparece en la barra de título como el que se muestra en la pantalla de Inicio. 3 El tiempo que permanece activa es poco.

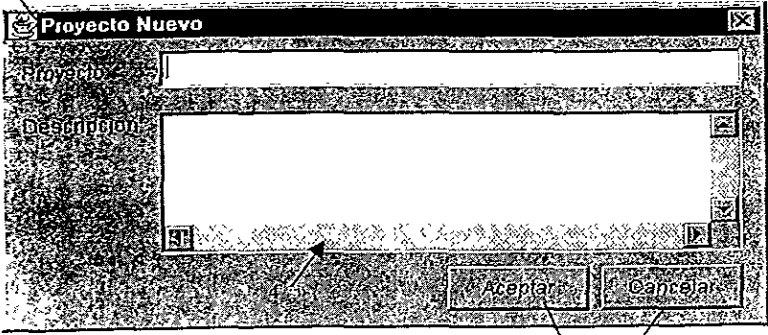
Ejemplo N° 5 Pantalla Mensajes

Registro de Defectos por Pantalla	
<p>Registro: 5 Módulo: En todos Pantalla: Mensajes Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción:</p> <p>Esta pantalla despliega un mensaje dependiendo de la situación, si el usuario deja un cuadro de texto en blanco, o bien se le olvida introducir información en alguno de ellos. El sistema desplegará un mensaje como "Faltó nombre". Si el usuario se equivoca al introducir su clave o contraseña enviará un mensaje como el que se muestra en la parte inferior del cuadro.</p>	<p>Defectos encontrados</p> <ol style="list-style-type: none"> 1. El mensaje que se muestra en esta pantalla en particular <p style="margin-left: 40px;"><i>"Usuario: pruebas inválido"</i> y los mensajes que indican que faltó X información no son muy apropiados.</p> 2. El botón <i>Aceptar</i> no responde a la tecla ENTER 3. El logotipo no es el apropiado
 <p>The screenshot shows a dialog box with a title bar 'Mensaje' and a close button. The main text area contains 'Usuario: pruebas inválido'. Below the text is a button labeled 'Aceptar'. Three arrows point to specific elements: arrow 1 points to the message text, arrow 2 points to the 'Aceptar' button, and arrow 3 points to the icon in the title bar.</p>	

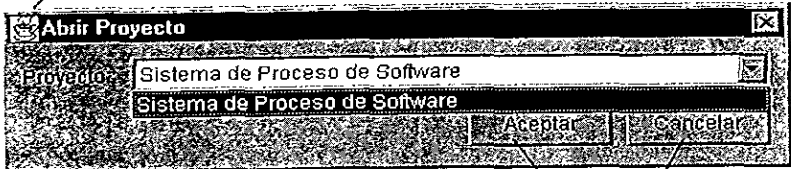
Ejemplo N° 6. Pantalla Principal (Menú del sistema)

Registro de Defectos por Pantalla	
<p>Registro: 6 Módulo: En todos Pantalla: Principal Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción:</p> <p>Muestra los menús que integran el sistema (en total son seis) y permite ver y moverse en las opciones de cada una de ellas con el mouse. Cuando se crea un nuevo proyecto las <i>opciones de actividad, reporte y defecto</i> se encuentran inactivas. Esta pantalla permanece siempre abierta hasta que se cierra por completo el sistema.</p> <p>Algunas de las opciones del <i>Menú Proyecto</i> y del <i>Menú Actividad</i> manejan las teclas SHORTCUT.</p>	<p>Defectos encontrados.</p> <ol style="list-style-type: none"> 1. El logotipo que aparece en la barra de título no es el apropiado 2. <i>No funcionan las teclas SHORTCUT.</i> 3. No se puede mover de un menú a otro con el uso de las teclas, ni permite seleccionar una opción con la tecla ENTER. 4. Pantalla un poco simple y triste 5. Algunas de las pantallas que se despliegan sobre esta, son más grandes, es decir no hay uniformidad en el diseño de las pantallas.
	

Ejemplo N° 7 Pantalla Proyecto Nuevo

Registro de Defectos por Pantalla	
Registro 7 Módulo: Proyecto Pantalla Proyecto Nuevo Revisor Analista	Fecha de revisión: 02/07/99
<p>Descripción:</p> <p>Solicita los datos del <i>Proyecto Nuevo</i>. Si los datos son correctos se oprime el botón <i>Aceptar</i> con el mouse para almacenar los datos en la base de datos, enseguida esta pantalla se cierra dejando el control a la <i>Pantalla Principal</i>, la cual en la barra de título dentro de los parentesis cuadrados ([]), despliega el <i>Nombre del Proyecto</i> que ha sido creado y con el cual se va a trabajar. <i>A partir de este momento las opciones de los menús actividad, defecto y reportes se activan. Permite moverse de un campo a otro con la tecla TAB o con el mouse.</i></p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que aparece en la barra de título no es el apropiado 2. No maneja <i>Pantalla de Mensaje de Error</i>, cuando el nombre del proyecto ya existe. 3. Los datos que solicita esta pantalla no proporcionan mucha información del proyecto creado. 4. La barra del scrollbar horizontal del campo descripción, hace que la lectura sea pesada, es decir sale sobrando 5. No responde a la tecla ENTER
 <p style="text-align: center;">5</p>	

Ejemplo N° 8. Pantalla Abrir Proyecto

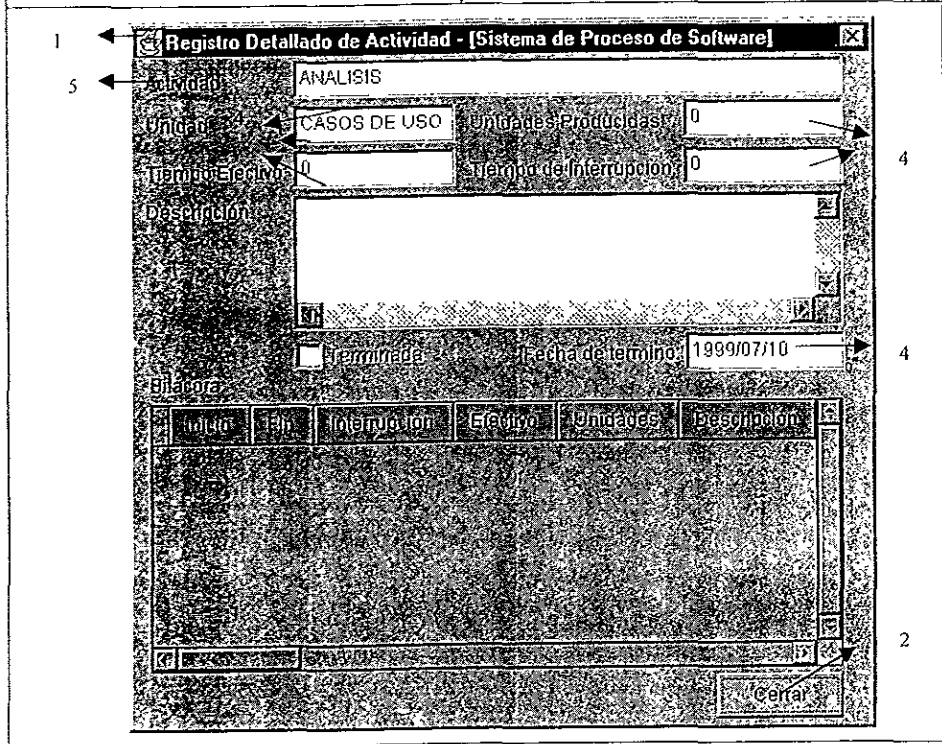
Registro de Defectos por Pantalla	
<p>Registro: 8 Módulo: Proyecto Pantalla: Abrir Proyecto Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción:</p> <p>Muestra una lista de proyectos existentes en la base de datos <i>Para Abrir un Proyecto</i> se selecciona de la lista, se oprime el botón <i>Aceptar</i> con el mouse, esta ventana de cierra y regresa el control a la <i>Pantalla Principal</i>, la cual en los paréntesis cuadrados que se encuentra en la barra de título despliega el nombre del proyecto seleccionado. Mientras no se seleccione un proyecto las opciones de los menús actividad, defecto y reporte permanecen inactivas</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utiliza no representa el sistema 2. No responde a la tecla ENTER en el botón <i>Aceptar</i> y <i>Cancelar</i>
	

Ejemplo N° 9. Pantalla Actividad Nueva

Registro de Defectos por Pantalla	
<p>Registro 9 Módulo: Actividad Pantalla: Actividad Nueva Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción:</p> <p>Solicita los datos para dar de alta una actividad dentro del proyecto creado o seleccionado. Permite moverse de un campo a otro con la tecla TAB o con el botón del mouse. Si faltó introducir un dato en alguno cuadro de texto excepto en la descripción, y se oprime el botón <i>Aceptar</i> despliega una <i>Ventana de Mensaje</i> indicando lo ocurrido. Si se cambia el formato de fecha, despliega un mensaje de error.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1 El logotipo que utilizan no representa al sistema. 2. No acepta la tecla ENTER. 3. El concepto del campo unidad no es claro, es decir falta especificar a que se refiere (ayuda). 4. La etiqueta actividad se duplica con el título de la pantalla 5. El scrollbar horizontal del campo descripción no es necesario. 6 Falta mas información relacionada con la actividad. 7 No manejan los acentos en las etiquetas de los campos.
<p>The screenshot shows a dialog box titled "Actividad Nueva - [Sistema de Proceso de Software]". It contains several input fields: "Actividad" (with a small icon to its left), "Unidad", "Fecha de término" (with a date format mask "yyyy/mm/dd"), and "Descripción". At the bottom, there are "Aceptar" and "Cancela" buttons. Numbered callouts point to: 1 (the icon), 2 (the "Aceptar" button), 3 (the "Unidad" label), 4 (the "Actividad" label), 5 (the scrollbar of the "Descripción" field), 6 (the "Fecha de término" label), and 7 (the date format mask).</p>	

Ejemplo N° 10. Pantalla Registro Detallado de Actividad

Registro de Defectos por Pantalla	
Registro 10 Módulo: Actividad Pantalla: Registro Detallado de Actividad Revisor: Analista	Fecha de revisión: 02/07/99
<p>Descripción:</p> <p>Esta pantalla se divide en dos partes, en la parte superior despliega todos los datos referentes a la actividad seleccionada, como son nombre del proyecto al que pertenece, <i>nombre de la actividad, unidades producidas, tiempo de interrupción</i>, entre otras. En la parte inferior muestra una bitácora de todas las actividades en línea. Estos datos no pueden ser modificados en esta pantalla porque son solo de lectura.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. El botón <i>Cerrar</i> no responde a la tecla ENTER. 3. El diseño de esta pantalla es mas estrecha y larga que la <i>Pantalla Principal</i>. Por lo tanto no hay uniformidad en el diseño de pantallas 4. Los datos que esta pantalla se despliega son de lectura y aparecen en cuadros de texto. 5. La etiqueta actividad se duplica con el titulo de la pantalla



Ejemplo N° 11 Pantalla Actualizar Actividad

Registro de Defectos por Pantalla	
<p>Registro. 11 Módulo. Actividad Pantalla: Actualizar Actividad Revisor Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción</p> <p>Esta pantalla es llamada por medio de botón Parar (=) de la <i>Pantalla Ejecutar Actividad</i>. Los campos actividad y unidad se despliegan automáticamente (son de lectura) Permite registrar los campos de <i>Unidades Producidas, Descripción</i> y si la actividad <i>término o no</i>. Los demás datos son calculados automáticamente por el sistema. Si no se introduce el número de unidades producidas y se oprime el botón <i>Aceptar</i> con el mouse, se despliega inmediatamente una <i>Ventana de Mensaje</i> avisándole lo ocurrido y por lo tanto no permite guardar los cambios. El botón <i>Cancelar</i>, cierra el cronómetro y regresa el control a la <i>Pantalla Principal</i>.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. Los mensajes de error no coordinan con el campo a evaluar 3. Los campos Actividad y Unidad son de lectura y se despliegan en cuadros de texto 4. Los botones no responde a la tecla ENTER 5. El campo Unidades Producidas es numérico y acepta cualquier carácter. 6. Los demás campos son calculados automáticamente por el sistema. Sin embargo permite que el usuario los modifique, lo cual ocasiona que al oprimir el botón <i>Aceptar</i> se confunda y la pantalla no responde. 7. La etiqueta Hora de término, no esta acentuada la palabra termino.

Ejemplo N° 12. Pantalla Actualizar Manualmente la Actividad

Registro de Defectos por Pantalla	
Registro: 12 Módulo: Actividad Pantalla: Actualizar Manualmente la Actividad Revisor: Analistal	Fecha de revisión: 02/07/99
Descripción: Permite registrar manualmente los tiempos de una actividad en línea de base. Los datos solicitados son fecha y hora de inicio, y de término, unidades producidas, descripción, y si esta o no terminada. El <i>Tiempo efectivo</i> se calcula automáticamente. Los campos <i>Actividad</i> y <i>Unidad</i> se despliegan automáticamente por el sistema. Permite moverse de un campo a otro con la tecla TAB, o con el mouse	Defectos encontrados: <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. Los campos de fecha y hora no están validados 3. Los campos de tiempo, no indican en que unidades de tiempo se miden o toman. 4. Los campos de actividad y unidad son de lectura y su información la despliegan en cuadros de texto. 5. Los botones no responden a la tecla ENTER. 6. No maneja mensajes de error. 7. El scrollbar horizontal del campo descripción, no es necesario. 8. La etiqueta Hora de término, no esta acentuada la palabra termino

Ejemplo N° 13. Pantalla Enviar Actividad

Registro de Defectos por Pantalla	
<p>Registro. 13</p> <p>Módulo Actividad</p> <p>Pantalla Externa opción Enviar Actividad</p> <p>Revisor: Analista I</p>	<p>Fecha de revisión 02/07/99</p>
<p>Descripción:</p> <p>Muestra una lista de todas las actividades que pertenecen a un proyecto y que pueden ser transmitidas al dispositivo externo. Esta pantalla se despliega pero no hace nada, porque no cuenta con el dispositivo externo para hacer la transferencia. Nota Lo mismo sucede con la pantalla recibir actividad</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1 El logotipo que utilizan no representa al sistema 2. No permite seleccionar la actividad que se desea enviar. 3. No aparece dentro del campo de texto de actividades el scrollbar vertical. 4. No despliega el nombre del proyecto en la barra de título. 5. Los botones no responde a la tecla ENTER.
<p>The screenshot shows a window titled 'Enviar Actividad'. The window contains a list of activities under the heading 'ANALISIS'. The activities listed are 'Diseño de clases de interfaz humana', 'Elaboracion de Imagenes', and 'Implementacion de Project'. At the bottom of the window are two buttons: 'Aceptar' and 'Cancelar'. Five numbered arrows point to specific elements: 1 points to the logo in the top-left corner; 2 points to the 'Aceptar' button; 3 points to the vertical scrollbar on the right side of the activity list; 4 points to the title bar of the window; and 5 points to the 'Cancelar' button.</p>	

Ejemplo N° 14 Pantalla Selección de Defecto

Registro de Defectos por Pantalla	
Registro: 14 Módulo Defecto Pantalla Selecccion de Defecto Revisor Analista	Fecha de revisión: 02/07/99
Descripción: Despliega una lista de actividades y otra de defectos Al seleccionar una actividad y un defecto, se oprime el botón <i>Aceptar</i> con el mouse (pero esta pantalla no responde), debería desplegar la pantalla <i>Ejecutar Defecto</i> . Esta pantalla la llama la opción <i>Corregir</i> . Sin embargo no hace nada. El botón <i>Cancelar</i> cierra la pantalla y regresa a la pantalla Principal.	Defectos encontrados: 1 El logotipo que utilizan no representa al sistema. 2 Despliega la lista de actividades y defectos, pero al parecer esta pantalla no fue terminada. Porque debería desplegar la pantalla del cronometro, llamada <i>Ejecutar Defecto</i> para llevar la contabilidad del tiempo, y posteriormente desplegar la pantalla <i>Registrar Defecto Automático</i> , si se oprimió el botón <i>parar</i> . 3. Los botones no responde a la tecla ENTER.

Ejemplo N° 15. Pantalla Nuevo Tipo de Defecto

Registro de Defectos por Pantalla	
Registro: 15 Módulo: Defecto Pantalla: Nuevo Tipo de Defecto Revisor: Analista	Fecha de revisión: 02/07/99
Descripción: Solicita la descripción de un nuevo tipo de defecto	Defectos encontrados <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema. 2. Los botones no responde a la tecla ENTER 3. Registra un Nuevo tipo de defecto. Esta pantalla solicita la descripción del defecto, pero no indica de que defecto se trata y si este tiene alguna clave para identificarlo. 4. Porque la pantalla se llama <i>Nuevo Tipo de Defecto</i> si no registra defectos.

Ejemplo N° 16. Pantalla Reporte por Proyecto

Registro de Defectos por Pantalla	
Registro 16 Módulo Reporte Pantalla Reporte por Proyecto (Presenta el mismo problema en los otros reportes) Revisor Analista1	Fecha de revisión: 02/07/99
<p style="text-align: center;">Descripción:</p> <p>Despliega en pantalla el reporte del proyecto seleccionado con la fecha de impresión, nombre y descripción del mismo, además lista las actividades que le pertenecen. Esta pantalla solo es de consulta, pero se puede imprimir desde aquí.</p>	<p style="text-align: center;">Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema. 2. Permite escribir y modificar el contenido del reporte, no debería hacerlo porque es una pantalla de consulta, pero al momento de imprimir los cambios no los hace. 3. No despliega el Nombre del Proyecto en la barra de título. 4. No responden los botones a la tecla ENTER.

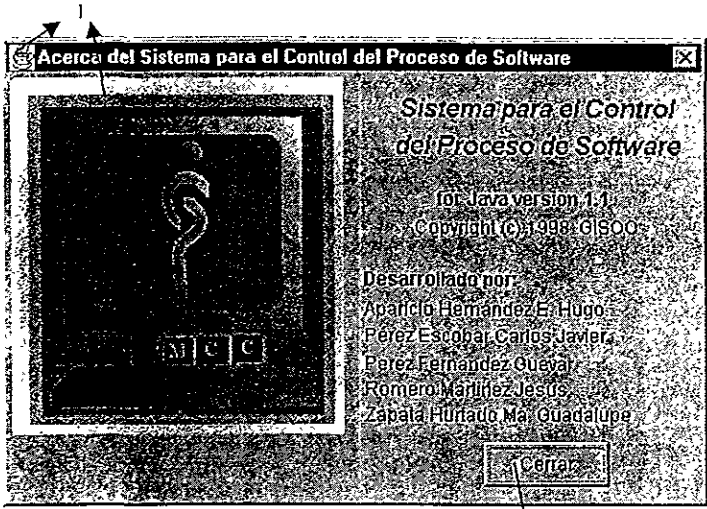
Ejemplo N° 17. Pantalla Modificar Contraseña

Registro de Defectos por Pantalla	
Registro: 17 Módulo: Usuario Pantalla: Modificar Contraseña Revisor: Analista	Fecha de revisión: 02/07/99
<p>Descripción</p> <p>Permite que el usuario o administrador cambien su contraseña Si oprime el botón <i>Aceptar</i> con el mouse guarda los cambios en la base de datos. En caso contrario, cierra la pantalla y regresa el control a la <i>Pantalla Principal</i></p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. Los botones no responden a la tecla ENTER. 3. Los nombres de las etiquetas no son claros, porque clave y contraseña son sinónimos. 4. La etiqueta del botón aceptar debería ser "salvar".
<p>The screenshot shows a dialog box titled 'Modificar Contraseña'. It contains three input fields: 'Nombre' (with the value 'Haparici'), 'Contraseña', and 'Confirmación'. At the bottom are two buttons: 'Aceptar' and 'Cancelar'. Numbered callouts are as follows: 1 points to the application icon in the title bar; 2 points to the 'Aceptar' and 'Cancelar' buttons; 3 points to the 'Nombre' and 'Contraseña' labels; 4 points to the 'Aceptar' button label.</p>	

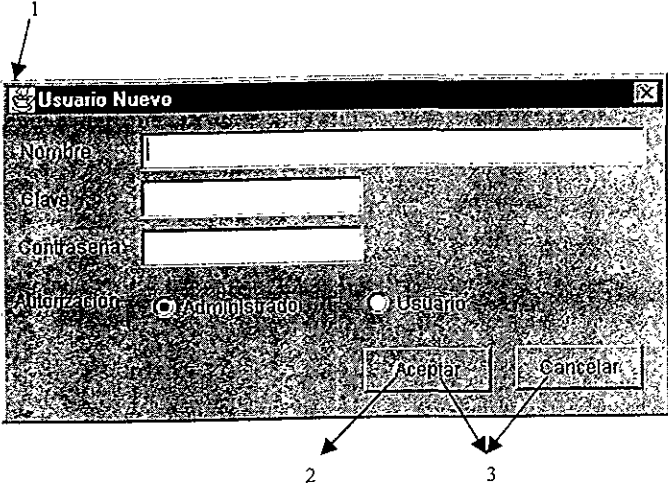
Ejemplo N° 18 Pantalla Ayuda

Registro de Defectos por Pantalla	
<p>Registro: 18 Módulo: Ayuda Pantalla: Ayuda Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción:</p> <p>Muestra los temas de ayuda que el sistema proporciona al usuario, en caso de existir duda, o bien le muestra los pasos que debe seguir a través de un ejemplo de cómo registrar un proyecto, entre otras</p> <p>Nota: Esta pantalla no está terminada</p>	<p>Defectos encontrados</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. No despliega el nombre del sistema en la barra de título. 3. Solo despliega un índice. 4. El botón <i>Cerrar</i> no responde a la tecla ENTER 5. En conclusión es una pantalla de consulta, pero no funciona como tal.
<p>The screenshot shows a window titled 'Ayuda' with a standard Windows-style title bar. The main content area displays the following text: 'Grupo de Ingeniería de Software Orientado a Objetos', 'Copyright (c) 1998 GISOO', and a list of help topics under the heading 'TEMAS DE AYUDA': 'Que es un proyecto?', 'Que es una actividad?', 'Que es un defecto?', 'Como se registra un proyecto?', 'Como se registra una actividad?', 'Como se registra un defecto?', 'Como se checa el seguimiento de una actividad?', 'Reportes', and 'Seguridad'. At the bottom right, there is a button labeled 'Cerrar'. Four numbered callouts are present: 1 points to the application icon in the title bar, 2 points to the title text, 3 points to the list of help topics, and 4 points to the 'Cerrar' button.</p>	

Ejemplo N° 19 Pantalla Acerca de ...

Registro de Defectos por Pantalla	
<p>Registro: 19 Módulo: Ayuda Pantalla: Acerca de .. Revisor: Analista1</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción.</p> <p>Muestra la información general del sistema, el nombre de los integrantes del sistema y despliega el logotipo del sistema.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1 El logotipo que utilizan no representa al sistema, tanto el que aparece en la barra de titulo como el que representa el sistema 2. El botón <i>Cerrar</i> no responde a la tecla ENTER
	

Ejemplo N° 20. Pantalla Usuario Nuevo

Registro de Defectos por Pantalla	
<p>Registro: 20 Módulo: Usuario Pantalla: Usuario Nuevo Revisor: Analista</p>	<p>Fecha de revisión: 02/07/99</p>
<p>Descripción Permite que el usuario cree una nueva cuenta para un usuario, el cual puede tener el nivel de administrador, o bien de usuario normal.</p>	<p>Defectos encontrados:</p> <ol style="list-style-type: none"> 1. El logotipo que utilizan no representa al sistema 2. La etiqueta del botón aceptar debería ser guardar 3. Los botones no responde a la tecla ENTER. 4. No confirma la contraseña 5. Clave y contraseña son sinónimos.
 <p>The screenshot shows a window titled 'Usuario Nuevo' with a small icon in the top-left corner. Below the title bar are three text input fields labeled 'Nombre', 'Clave', and 'Contraseña'. Underneath these fields are two radio buttons for 'Autorización', with 'Administrador' selected. At the bottom right of the form are two buttons: 'Aceptar' and 'Cancelar'. Three numbered arrows point to these elements: arrow 1 points to the window icon, arrow 2 points to the 'Aceptar' button, and arrow 3 points to the 'Cancelar' button.</p>	

APÉNDICE B

FORMAS PARA LAS PROPUESTAS DE CAMBIO

Solicitud N° 2.

Solicitud de Cambio	
Solicitud N° <u>2</u>	Fecha de solicitud: <u>22/12/1999</u> Hora: <u>09 30 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Ejecutar Actividad</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar la presentación de la pantalla que sea más amigable 2 Eliminar el botón de Información. Para que la información que este botón despliega aparezca en esta misma pantalla a través de etiquetas en color azul 3 Cambiar la etiqueta del botón Defecto por Registrar Defecto 4 Cambiar la etiqueta del botón Cancelar por una imagen que ejemplifique la función que realiza dicho botón 5 Eliminar los botones minimizar y restaurar de la barra de título 6 Activar el uso de las teclas TAB y ENTER en el teclado 7 Cambiar el logotipo de Java que aparece en la barra de título por el icono del nuevo sistema. 	
Razones para el cambio:	
<ul style="list-style-type: none"> ◆ Hacer la pantalla más dinámica para el usuario ◆ No limitar al usuario a manejar la pantalla solo con el uso del mouse. Además, puedan hacer uso de las teclas TAB y ENTER ◆ Cuando esta pantalla se abra, la pantalla principal quedará oculta, así al eliminar el botón Información, los datos que este botón desplegaba, aparecerán directamente sobre la pantalla Ejecutar como etiquetas. Es decir, el sistema desplegará automáticamente el nombre del proyecto y actividad en la que el usuario se encuentra trabajando actualmente en color azul. También, la hora del reloj se desplegará en color azul 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
Fecha de recepción: <u>22/12/1999</u> Hora: <u>10:00 a.m.</u>	
Solicitud N° <u>2</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada (<input type="checkbox"/>)
Autorizado por: <u>Mesa de Control de Cambios virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio:	
() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas. _____	
Comentarios:	
Para poder implementar esta pantalla era necesario que el módulo de Proyecto, sobre todo la opción abrir Proyecto estuviera funcionando sin ningún problema. Ya que no se puede abrir una actividad sin antes pertenecer a un proyecto.	
Fecha de decisión: <u>22/12/1999</u>	
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio

Solicitud N° 2

Fecha de recepción 22/12/1999

Hora: 11:30 a.m.

Ingeniero(s) de software Programador 1 y Programador 2

Versiones: 1

N° total de cambios implementados en los componentes: 2

Descripción de los cambios:

De los siete cambios propuestos fue posible realizar los seis primeros, el séptimo quedó registrado en una bitácora para una versión posterior. El cambio pendiente es:

- No se pudo cambiar el logotipo de Java por el icono del nuevo sistema en la barra de título de la pantalla.
- Además se corrigió que al cerrar la ventana con el botón X de la barra de título, el control del sistema regresa a la pantalla principal. Para llevar a cabo los cambios solicitados fue necesario importar la clase `ExecutiveActivityStopwatchCommand`. Así como, hacer uso de las clases `Database`, `RaiseError`, `Activity`, `Project`, `Timer`, `ManageTimer`, e invocar a `OpenActivityWindow`, `AutomaticRecordActivityWindow` y `CommandDefectCommand`. Se generó la clase `ExecuteActivityStopwatchWindow`, las variables utilizadas en la clase `Command` se estandarizaron con el nombre de las variables declaradas para esta pantalla. Como es importante hacer notas que los nombres de las clases `Command` del sistema original se uniformizaron con el nombre de las clases `Windows`.

Fecha de inicio 07/01/2000

Fecha de terminación 10/01/2000

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 2

Fecha de recepción: 11/01/2000

Hora: 06:30 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Después de varias pruebas que se realizaron, se decidió liberar la nueva pantalla, pero antes fue necesario que los programadores implementaran las pantallas de Registrar automáticamente una actividad, registrar defecto y abrir actividad.

Fecha de liberación: 12/01/2000

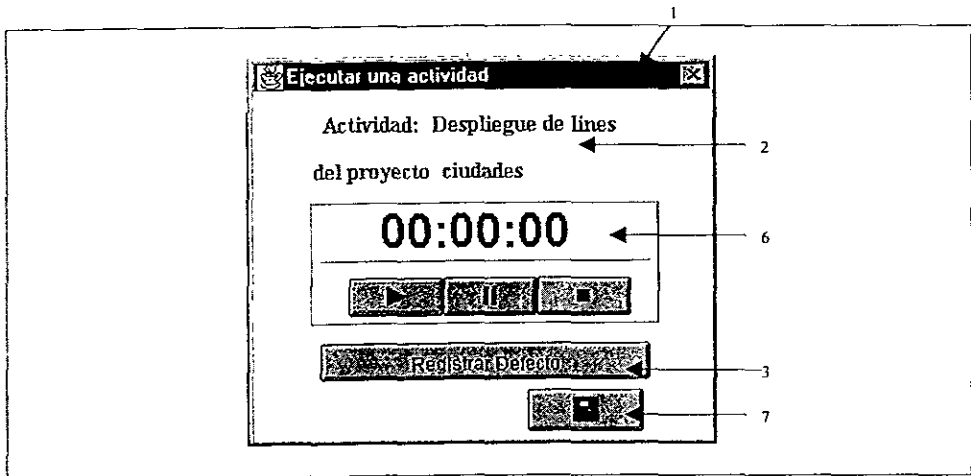
Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedó la Pantalla Ejecutar Actividad una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 2

Los cambios efectuados fueron:

1. Se eliminaron los botones minimizar y restaurar de la barra de título
2. Se eliminó el botón *Información*. Los datos proporcionados por este botón aparecen como etiquetas en la parte superior de la pantalla. Los datos de color azul son proporcionados directamente por el sistema.
3. Se cambió el nombre de la etiqueta Defecto por Registrar Defecto.
4. Los botones se pueden activar con la tecla ENTER o bien a través del mouse. Y moverse de una opción a otra con la tecla TAB o con el mouse.
5. Se cambió el diseño de la pantalla por completo.
6. El reloj se presenta de color azul.
7. Se cambió la etiqueta Cancelar por una imagen que represente a esta etiqueta.

Continúa en la siguiente hoja



Solicitud N° 3.

Solicitud de Cambio	
Solicitud N° <u> 3 </u>	Fecha de solicitud: <u>13/01/2000</u> Hora: <u>11:00 a.m</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Defecto Nuevo</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar la presentación de la pantalla hacerla más amigable y sobretodo entendible para el usuario 2 Los datos que el sistema despliega automáticamente deberá aparecer en letras amarillas oscuras con el fondo del campo azul oscuro 3 Agregar el formato correspondiente a los campos <i>Fecha</i> y <i>Hora de Detección</i>. 4 Las etiquetas de los campos <i>Actividad que inyectó</i> y <i>Actividad que corregirá</i> cambiarlos por <i>Inyectado en la fase</i> y <i>Removido en la fase</i> 5 Los nombres de las etiquetas de algunos campos serán más explicitos 6 La etiqueta <i>Defecto que introduyo</i> se cambiará por <i>Producto</i>. 7 Cambiar el logotipo de Java que aparece en la barra de titulo por el icono del nuevo sistema. 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Los datos que esta pantalla solicita <i>no muy claros pueden llegar a confundir al usuario</i> Los cambios que se proponen son tanto para mejorar la funcionalidad del sistema como hacer más amigables las pantallas. Esto se puede logra si la información solicitada por esta pantalla en particular es mas explicita en la información que solicitan o proporcionan el sistema • Los campos <i>Inyectado</i> y <i>Removido en la fase</i> no serán de selección, porque la finalidad principal de este sistema es que pueda aplicar a cualquier tipo de proyecto 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio

Fecha de recepción: 14/01/2000

Hora: 09:45 a.m.

Solicitud N° 3

Aprobada (X) Rechazada ()

Autorizado por: Mesa de Control de Cambios Virtual

Valoración del cambio: El cambio no repercute en ninguna parte del sistema

Prioridad del cambio:

() Crítico () Muy importante (X) Importante () Interesante () Inoportuno

Costo estimado del cambio en horas: _____

Comentarios:

Fecha de decisión: 14/01/2000

Firma de la MCC

Resultado de la Implementación de Cambio

Solicitud N° 3

Fecha de recepción: 15/01/2000

Hora: 09:45 a.m.

Ingeniero(s) de software: Programador 1 y Programador 2

N° total de cambios implementados en los componentes: 2

Versiones: 1

Descripción de los cambios:

- Además se agregó el campo y etiqueta *Detectado por* para saber que persona encontró el defecto.
- Se agregó la tecla ENTER para que el usuario pueda seleccionar el botón de *Aceptar* y *Cancelar* con el mouse y la tecla
- Se agregó la etiqueta mins. para que el usuario conozca que el tiempo invertido en buscar un defecto se mide en minutos.
- Se eliminaron los corchetes que aparecen en la barra de título, en donde se desplegaba el nombre del proyecto, porque esta pantalla al ser invocada por el sistema se desplegará dentro del área de trabajo de la pantalla principal, con esto se evita repetir el nombre del proyecto n veces.

Para llevar a cabo los cambios solicitados fue necesario importar la clase FoundDefectCommand. Así como, hacer uso de las clases MainWindow, Database e invocar a DefectType, RaiseError, ManageTime y Defect. Se genero la clase FoundDefectWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.

Fecha de inicio: 15/01/2000

Fecha de terminación: 17/01/2000

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 3

Fecha de recepción: 18/01/2000

Autoridad responsable: Seguridad y Calidad

Comentarios

Para probar esta pantalla fue necesario que los programadores implementará las pantalla Ejecutar un Defecto, puesto que esta pantalla puede ser invocada desde Ejecutar una actividad, o bien desde Ejecutar un defecto, esta última no fue diseñada en el sistema original.

Fecha de liberación: 19/01/2000

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Defecto Nuevo una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio.

1. Los cambios efectuados fueron
 1. Se cambio el nombre del titulo de la pantalla " Defecto Nuevo - [Sistema de Proceso de Software]" que aparece en la barra de titulo por el nombre del grupo " Prosoft"
 2. Se cambio el diseño de la pantalla por completo.
 3. Se cambiaron el nombre de las etiquetas *Actividad que Detecto* por *Actividad en la cuál se encontró el defecto*, *Actividad que Inyectó por Inyectado en la fase*, *Actividad que Corregirá por Removido en la fase*, *Defecto que introdujo por Producto*, *Tiempo de Búsqueda por Tiempo Invertido en buscar el defecto*
 4. Se agrego el campo *Detectado por*
 5. Se cambiaron las etiquetas de los botones *Aceptar* y *Cancelar* por las imágenes que representan *guardar* y *cancelar*.
 6. Se agrego la etiqueta *min* para indicar que el tiempo se mide en minutos
 7. Se agrego el formato a los campos *Fecha* y *Hora de detección*.

Solicitud N° 4.

Solicitud de Cambio	
Solicitud N° <u>4</u>	Fecha de solicitud: <u>29/09/1999</u> Hora: <u>09:45 a m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Inicio</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar la presentación de la pantalla 2 Cambiar el icono que usaron en la versión original por un icono que represente la función principal del Sistema para el Control del Proceso de Software 3 Resaltar más el título del sistema 4 Aumentar más el tiempo en que permanece activa esta pantalla 5 Eliminar la barra de título de dicha pantalla 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Hacer la pantalla más dinámica para el usuario, es decir presentar una pantalla más estética con un icono que indique que el sistema registrar, o bien toma el tiempo que le lleva a un usuario elaborar cualquier tipo de proyecto, puede ser por ejemplo desde su análisis, diseño, prueba e implementación usando como herramienta la computadora. • La pantalla permanecerá activa el tiempo necesario para que el usuario el usuario la pueda observar en detalle 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
	Fecha de recepción. <u>29/09/1999</u> Hora: <u>12:50 p m.</u>
Solicitud N° <u>4</u>	Aprobada (X) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio:	() Crítico () Muy importante (X) Importante () Interesante () Inoportuno
Costo estimado del cambio en horas _____	
Comentarios	Fecha de decisión. <u>29/09/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio

Solicitud N° 4

Fecha de recepción: 30/09/1999

Hora: 10:00 a.m.

Ingeniero(s) de software: Programador 1 y Programador 2

Versiones: 1

N° total de cambios implementados en los componentes: 5

Descripción de los cambios:

De los cinco cambios propuestos fue posible realizar los cuatro primeros, el quinto quedo registrado en una bitácora para una versión posterior

El cambio pendiente es:

- No se pudo eliminar la barra de título de la pantalla. Sin embargo, se pudieron eliminar los botones minimizar y restaurar.

Fecha de inicio: 30/09/1999

Fecha de terminación: 30/09/1999

Firma del Ingeniero de software

Liberación de Cambio

N° de solicitud: 4

Fecha de recepción: 01/10/1999

Autoridad responsable: Seguridad y Calidad

Comentarios

El cambio se libero sin ningún problema

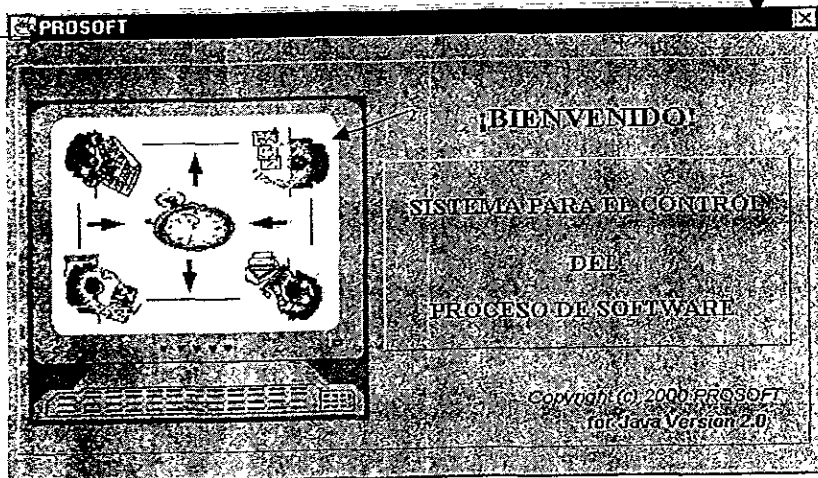
Fecha de liberación: 01/10/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Inicio una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 4

Los cambios efectuados fueron:

- 1 Se eliminaron los botones minimizar y restaurar de la barra de título
- 2 Se cambio el logotipo del sistema por uno que ejemplifica la función del SCPS
- 3 Se cambio el diseño de la pantalla por completo
- 4 La pantalla permanece abierta un tiempo más razonable
- 5 Se agrego el nombre del grupo en la barra de título



Solicitud N° 5.

Solicitud de Cambio

Solicitud N° 5 Fecha de solicitud: 52/10/1999
Hora: 11:00 a.m.

Propuesto por: Solicitante 1 y Solicitante 2
 Sistema/Proyecto: Control del Proceso de Software Versión: 1.0
 Componente a ser cambiado: Pantalla Mensajes Versión: 1.0

Descripción del cambio:

- 1 Cambiar el mensaje *Usuario pruebas inválido* por *Usuario [nombre del usuario] no ha sido dado de alta*
- 2 Cambiar el mensaje *Faltó nombre* por *Favor de proporcionar el nombre de X.*
- 3 Cambiar la presentación de la pantalla
- 4 Cambiar el logotipo de Java que aparece en la barra de título, por el icono del nuevo sistema

Razones para el cambio:

- ♦ Los mensajes que despliega esta pantalla deben estar de acorde a la información que cada pantalla solicita. Si el usuario deja un campo en blanco y este se declara como un campo validado, le enviará al usuario un mensaje indicándole que no puede continuar trabajando sino proporciona los datos solicitados. En caso que el usuario se equivoque al proporcionar los datos en un campo validado y oprime el botón Aceptar, o bien continúa capturando datos en otro campo, entonces el sistema le indicara que a cometido un error, que deberá de corregirlo antes de continuar.

Firma del solicitante

Aprobación de Cambio

Fecha de recepción: 25/10/1999
Hora: 12:00 p.m.

Solicitud N° 5 Aprobada (X) Rechazada ()

Autorizada por: Mesa de Control de Cambios Virtual

Valoración del cambio. El cambio no repercute en ninguna parte del sistema.

Prioridad del cambio:
 Crítico Muy importante Importante Interesante Inoportuno

Costo estimado del cambio en horas: _____

Comentarios:

Fecha de decisión: 25/10/1999

Firma de la MCC

Resultado de la Implementación de Cambio

Solicitud N° 5 Fecha de recepción: 25/10/1999
Hora: 05:00 p.m.

Ingeniero(s) de software: Programador 1 y Programador 2
 N° total de cambios implementados en los componentes: 4 Versiones: 1

Descripción de los cambios:

De los cuatro cambios propuestos fue posible realizar los tres primeros, el cuarto quedó registrado en una bitácora para una versión posterior.

El cambio pendiente es:

- ♦ No se pudo cambiar el logotipo de Java por el icono del nuevo sistema en la barra de título.
- ♦ Se cambió la pantalla sin ningún problema, los campos que no pueden quedarse en blanco se validan perfectamente.

Fecha de inicio: 26/10/1999 Fecha de terminación: 26/10/1999

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 5

Fecha de recepción: 27/10/1999

Hora: 11:00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios

Después de varias pruebas se liberaron los mensajes de error y de validación.

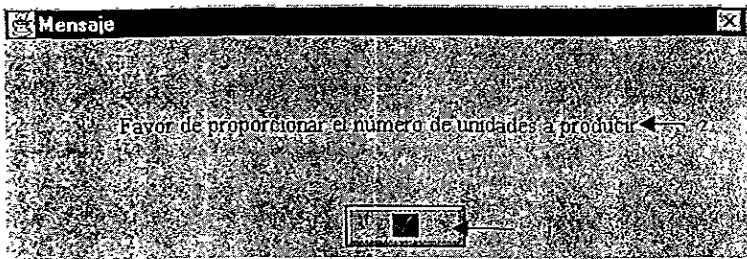
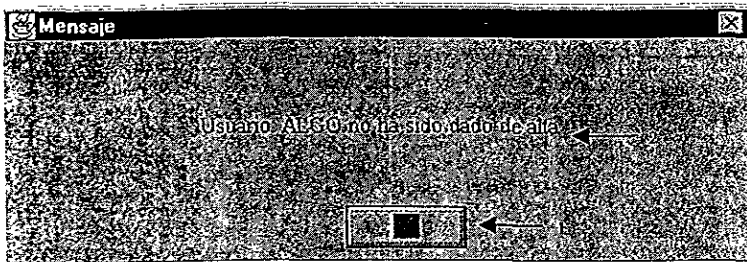
Fecha de liberación: 27/10/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Mensaje una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 5

Los cambios efectuados fueron

- 3 Se cambio la etiqueta del boton Aceptar por una imagen que representa a dicha función.
- 4 Se cambio el contenido de los mensajes que el sistema despliega cuando al usuario se equivoco al introducir su login o contraseña, o bien cuando deja un campo en blanco y este se encuentra validado.



Solicitud N° 6.

Solicitud de Cambio	
Solicitud N° <u>6</u>	Fecha de solicitud: <u>28/10/1999</u> Hora <u>10.00 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Principal</u>	Versión: <u>1.0</u>
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Eliminar las teclas <i>shortcut</i> que aparecen en las opciones de algunos módulos 2 Permitir moverse de un módulo a otro con el uso de las teclas (arriba, abajo, izquierda y derecha) del teclado 3 Permitir seleccionar las opciones con el uso de las teclas del teclado 4 Poner un fondo en el área de trabajo 5 Cambiar el logotipo de Java que aparece en la barra de título, por el icono del nuevo sistema 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Se pretende cambiar la presentación de la pantalla poniendo un fondo en el área de trabajo, para que no de la impresión de que es un editor de texto • Se propone eliminar las teclas <i>shortcut</i>, porque es más fácil que el usuario se mueva de un módulo a otro usando las teclas (arriba, abajo, izquierda y derecha) del teclado o el mouse 	
<hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma del solicitante	

Aprobación de Cambio	
	Fecha de recepción: <u>28/10/1999</u> Hora: <u>12:00 p.m.</u>
Solicitud N° <u>6</u>	Aprobada (X) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio:	
() Crítico () Muy importante (X) Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios:	
<u>Con los cambios propuestos la funcionalidad de la pantalla mejorará bastante y será más grata para la vista de cualquier usuario.</u>	
Fecha de decisión: <u>28/10/1999</u> <hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma de la MCC	

Resultado de la Implementación de Cambio

Solicitud N° 6

Fecha de recepción: 29/10/1999

Hora: 09:30 a.m.

Ingeniero(s) de software: Programador 1 y Programador 2

N° total de cambios implementados en los componentes: 5

Versiones: 1

Descripción de los cambios:

De los cinco cambios propuestos fue posible realizar los cuatro primeros, el séptimo queda registrado en una bitácora para una versión posterior. El cambio pendiente es:

- No se pudo cambiar el logotipo de Java por el icono propuesto para el nuevo sistema en la barra de título
- La presentación de la pantalla es más agradable a la vista de cualquier usuario
- El usuario puede moverse con más libertad de un módulo a otro, sin ningún problema
- Los módulos Actividad, Defecto, Reporte, la opción Cerrar del módulo Proyecto y la opción Modificar contraseña del módulo Usuario permanece inactivas mientras no se cree un nuevo proyecto o se abra un proyecto existente
- Si el usuario no es el administrador del sistema ni tiene atributos de administrador, entonces las opciones Nuevo y Eliminar usuario del módulo Usuario permanecerán inactivas

Fecha de inicio: 29/10/1999

Fecha de terminación: 30/10/1999

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 6

Fecha de recepción: 30/10/1999

Hora: 06:00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Se liberó la pantalla sin ningún problema, la presentación de la pantalla es agradable y original. Además permite que el usuario se puede mover a través de las diferentes opciones que presentan cada uno de los módulos que integran dicho sistema.

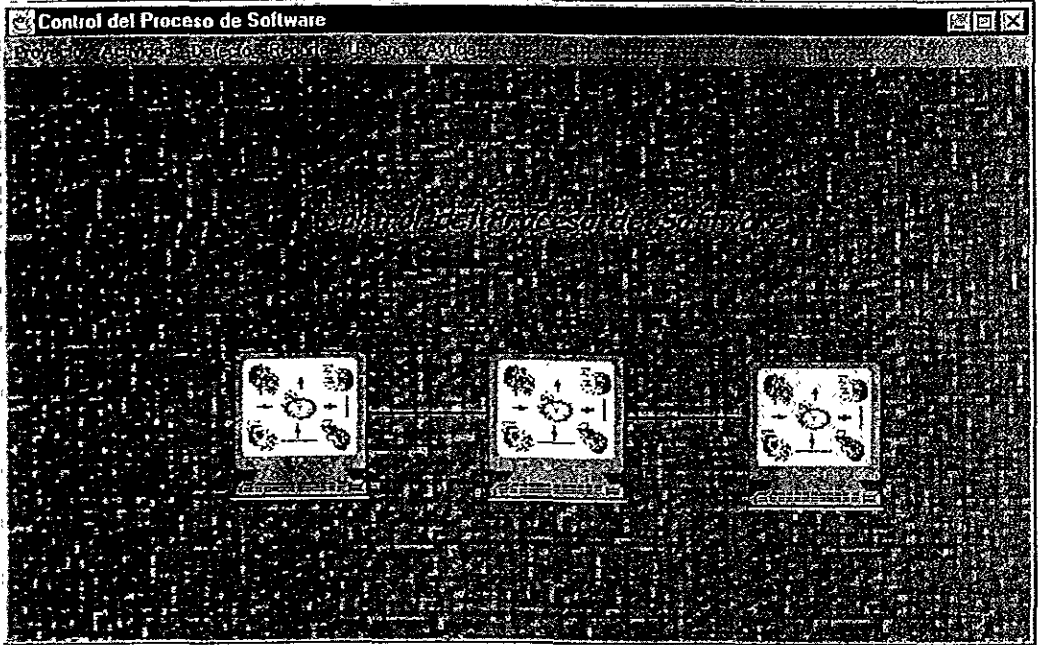
Fecha de liberación 31/10/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

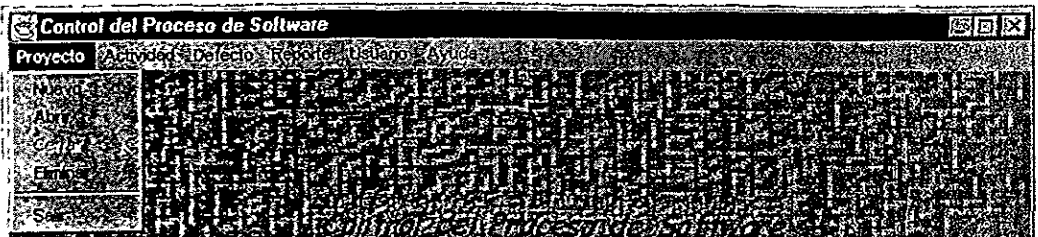
Esta imagen muestra como quedo la Pantalla Principal una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 6

Los cambios efectuados fueron

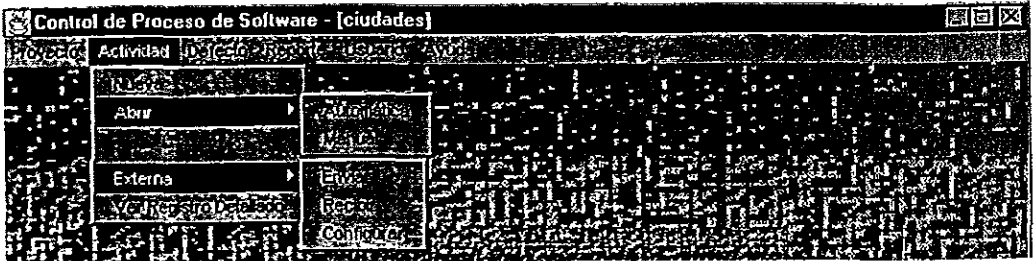
- 1 Se eliminaron las teclas shortcut.
- 2 Se puso un fondo en el área de trabajo de acorde al sistema para evitar que pareciera un editor de texto
- 3 El usuario se puede mover de un módulo a otro, seleccionar opciones a través de las teclas o con el uso del mouse



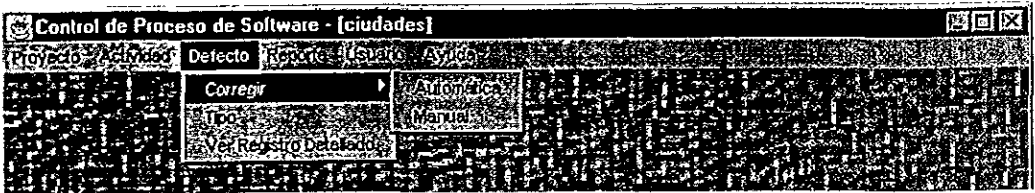
El módulo Proyecto el unico cambio que sufrio fue que se eliminaron las teclas shortcut, pero se implemento el uso de las teclas Enter y TAB para que el usuario pueda moverse con mas libertad.



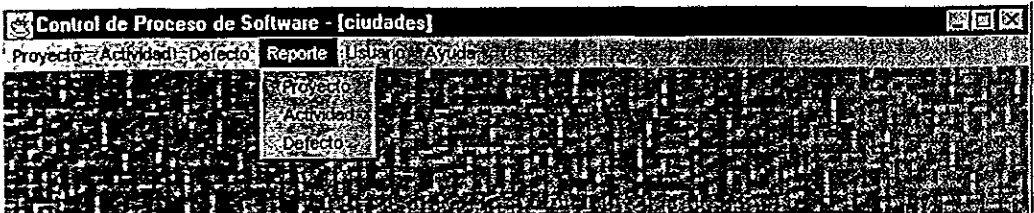
Se eliminó del módulo Actividad las opciones Ejecutar , y Manual . En su lugar se creó la opción Abrir , de esta se derivan dos opciones que son Automática y Manual , las cuales permiten registrar una actividad automáticamente o manualmente según sea la elección del usuario



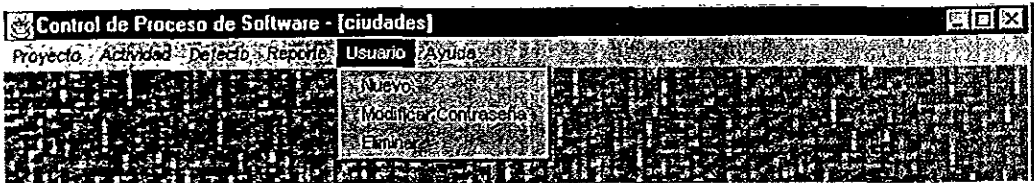
A la opción Corregir se le agregaron dos subopciones que son Automática y Manual que permiten al usuario registrar un defecto. La opción Corregir no fue implementada en la versión anterior. También se agregó la opción Ver Registro Detallado la cual permite ver el registro de todos los defectos que pertenecen a una actividad



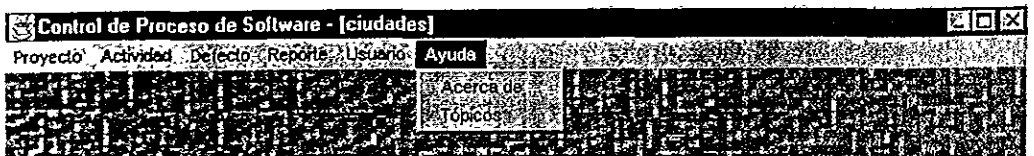
A este modulo se le eliminó la opción por Fecha.



A este modulo se le agregó la opción Modificar contraseña, esta opción puede ser usada tanto por el usuario como por el administrador.



Se cambió la opción Temas Ayuda por Tópicos y se implementó realmente la ayuda, en la versión anterior solo presentaba en esta pantalla los puntos que se iban a implementar pero no se llevaron a cabo.



Solicitud N° 7.

Solicitud de Cambio	
Solicitud N° <u>7</u>	Fecha de solicitud: <u>04/10/1999</u> Hora: <u>09:45 a.m.</u>
Propuesto por <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto. <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Proyecto Nuevo</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Agregar las etiquetas y los campos <i>Responsable</i>, <i>Ubicación</i>, <i>Fecha de inicio</i> y <i>Fecha de terminación</i> 2 Validar los campos <i>Nombre</i>, <i>Responsable</i>, <i>Fecha de inicio</i> y <i>Fecha de terminación</i> 3 Eliminar el scrollbar horizontal del campo de <i>Descripción</i> y dejar solo el scrollbar horizontal, lo cual va a permitir leer mejor el contenido de este campo 4 El tamaño de esta pantalla debe cubrir la área de trabajo de la pantalla principal 5 Los botones <i>Aceptar</i> y <i>Cancelar</i> respondan a la tecla TAB y ENTER del teclado 6 Desplegar el nombre de la pantalla en el centro de la misma, en la parte superior y de color azul 7 Cambiar el nombre de las etiquetas <i>Aceptar</i> y <i>Cancelar</i> por imágenes que ejemplifique la acción de estos botones 8 Cambiar el logotipo de Java que aparece en la barra de título por el icono del nuevo sistema. 	
Razones para el cambio:	
<ul style="list-style-type: none"> • La pantalla del sistema original no proporciona información real del proyecto. Es importante saber quién es el responsable del proyecto, cuando inicio y en cuanto tiempo se estimo terminarlo, para poder ver si se cumplio con el objetivo planeado 	
Firma del solicitante	

Aprobación de Cambio	
	Fecha de recepción: <u>04/10/1999</u> Hora: <u>05:00 p.m.</u>
Solicitud N° <u>7</u>	Aprobada (X) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio:	() Critico () Muy importante (X) Importante () Interesante () Inoportuno
Costo estimado del cambio en horas: _____	
Comentarios:	Fecha de decisión: <u>05/10/1999</u>
Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>7</u>	Fecha de recepción: <u>05/10/1999</u> Hora: <u>12.00 p.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>8</u>	
Descripción de los cambios:	
De los ocho cambios propuestos fue posible realizar los siete primeros, el octavo quedó registrado en una bitácora para una versión posterior.	
El cambio pendiente es,	
<ul style="list-style-type: none"> • No se pudo cambiar el logotipo de Java por el icono del nuevo sistema en la barra de título de la pantalla 	
Otros cambios	
<ul style="list-style-type: none"> • La pantalla cambio por completo, los datos que se agregaron a la pantalla, también se incluyeron en la tabla Project • Se cambio el botón <i>Aceptar</i> por <i>Salvar</i> 	
Para llevar a cabo los cambios solicitados fue necesario importar la clase <i>NewProjectCommand</i> Así como, hacer uso de las clases <i>MainWindow</i> , <i>Database</i> , <i>RaiseError</i> , <i>Project</i> , <i>ManageTime</i> . Se genero la clase <i>NewProjectWindow</i> , las variables utilizadas en la clase <i>Command</i> se estandarizaron con el nombre de las variables declaradas para esta pantalla	
Fecha de inicio: <u>05/10/1999</u>	Fecha de terminación. <u>07/10/1999</u>
Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 7

Fecha de recepción: 07/10/1999

Hora: 03 00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Se probaron y se validaron cada uno de los cambios propuestos, se revisó que los datos proporcionados por el usuario se almacenaran correctamente en la base de datos.
Además, los datos solicitados por esta pantalla dan una idea más general del objetivo del proyecto a crear. El usuario o responsable del proyecto podrá comprobar si la fecha estimada para la terminación del proyecto se cumplió, o bien se aproximó a la fecha señalada.

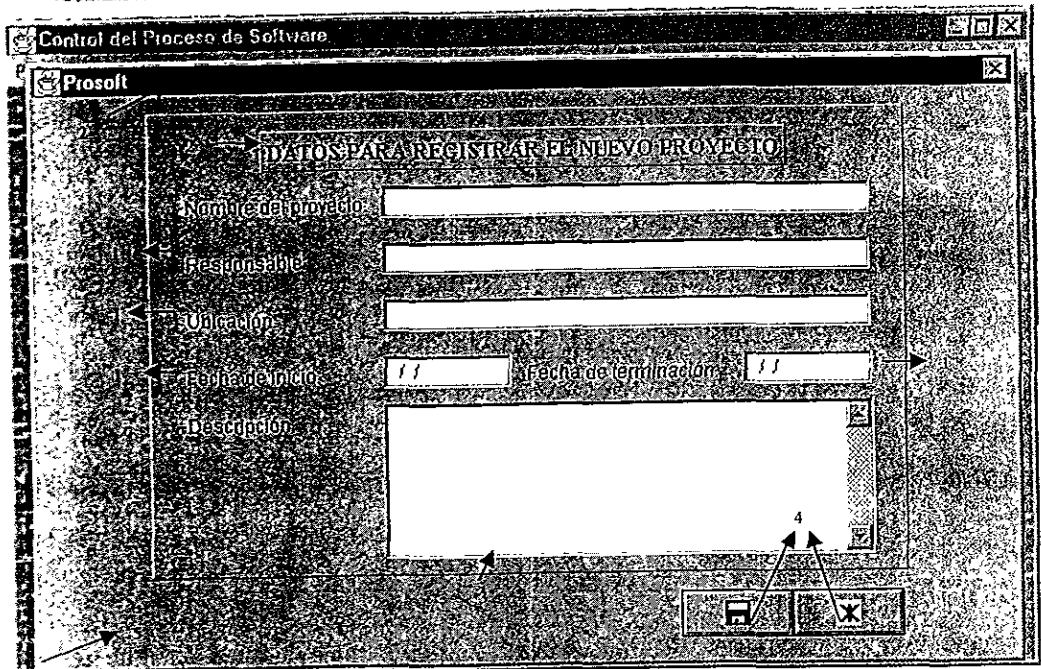
Fecha de liberación: 08/10/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Proyecto Nuevo una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 7

Los cambios efectuados fueron

- 1 Se agregaron las etiquetas y los campos Responsable, Ubicación, Fecha de inicio y Fecha de terminación
- 2 Se elimino el nombre de la pantalla de la barra de título y se despliega dentro de la misma pantalla en la parte superior y de color azul
- 3 Se elimino el scrollbar horizontal del campo de descripción
- 4 Se cambio el nombre de las etiquetas de los botones por imágenes
- 5 La pantalla abarca el área de trabajo de la pantalla principal



Solicitud N° 8.

Solicitud de Cambio	
Solicitud N° <u>8</u>	Fecha de solicitud: <u>09/10/1999</u> Hora: <u>10:00 a.m.</u>
Propuesto por <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Abrir Proyecto</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el nombre de las etiquetas Aceptar y Cancelar por imágenes que represente la función de dichos botones 2 Colocar en la parte superior de la pantalla una etiqueta de color azul que indique la función de dicha pantalla 	
Razones para el cambio:	
<ul style="list-style-type: none"> ♦ Hacer la pantalla más amigable y estandarizarla con las otras pantallas 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
Solicitud N° <u>8</u>	Fecha de recepción: <u>09/10/1999</u> Hora: <u>12:00 p.m.</u>
	Aprobada (<input checked="" type="checkbox"/>) Rechazada (<input type="checkbox"/>)
Autorizado por <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio:	
() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas _____	
Comentarios	
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>8</u>	Fecha de recepción: <u>10/10/1999</u> Hora: <u>09:30 a m.</u>
Ingeniero(s) de software. <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>2</u>	
Comentarios:	
<ul style="list-style-type: none"> ♦ Cambio la presentación de la pantalla, los datos no se ven amontonados. ♦ El usuario puede seleccionar una opción por medio del mouse o bien con las teclas ENTER o TAB del teclado. ♦ El cursor del mouse cambia de una flecha a una mano cuando se selecciona un proyecto de la lista, o bien cuando se selecciona un botón. ♦ Esta pantalla se despliega en el centro del área de trabajo de la pantalla principal cuando es invocada por medio del botón Aceptar. Para llevar a cabo los cambios solicitados fue necesario importar la clase NewProjectStopwatchCommand. Así como, hacer uso de las clases MainWindow, Database, RaiseError, Project, ManageTime. Se genero la clase NewProjectWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.	
Fecha de inicio: <u>10/10/1999</u>	Fecha de terminación: <u>11/10/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 8

Fecha de recepción 11/10/1999
 Hora: 05:00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

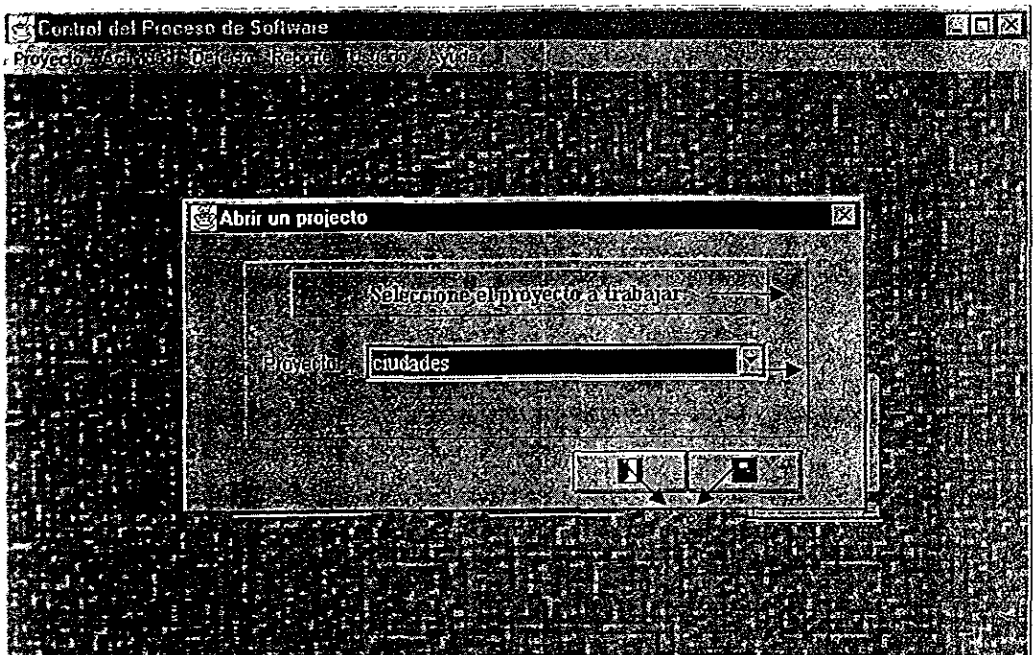
Los cambios implementados se probaron y validaron sin ningún problema

Fecha de liberación: 12/10/1999

Firma del Depto. de Seguridad y Calidad
 (Autoridad responsable de liberar el cambio)

La imagen muestra como quedo la Pantalla Abrir un proyecto una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 8

- 1 Los cambios efectuados fueron
- 1 La pantalla se despliega en el centro del área de trabajo de la pantalla principal
- 2 Se cambio el nombre de las etiquetas de los botones por imágenes que representa la función que desempeñan dichos botones
- 3 En la parte superior de la pantalla se coloco una etiqueta de color azul que muestra la función de dicha pantalla
- 1 Al seleccionar un objeto el cursor cambia de una flecha a una mano



Solicitud N° 9.

Solicitud de Cambio	
Solicitud N° <u>9</u>	Fecha de solicitud: <u>03/11/1999</u> Hora: <u>09:30 a m</u>
Propuesto por <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Actividad Nueva</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Eliminar el nombre del proyecto que se despliega en la barra de título 2 Agregar los campos y las etiquetas <i>Responsable</i> y <i>Fecha de inicio</i> 3 Cambiar la etiqueta <i>Unidad</i> por <i>Unidades a producir</i> 1 Cambiar la etiqueta <i>Fecha de término</i> por <i>Fecha límite</i> 5 Cambiar el formato de fecha yyyy/mm/dd por dd/mm/aaaa, pero que este formato no aparezca visible dentro del campo de texto 6 Eliminar el scrollbar del campo de Descripción 	
Razones para el cambio:	
<ul style="list-style-type: none"> • No son suficientes los datos solicitados por la pantalla original. Por ejemplo solicita la Fecha de terminación de la actividad pero nunca indica cuando inicio esta, así como quién es responsable y por último la etiqueta <i>Unidad</i> no especifica a que se refiere con la palabra unidad 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
Fecha de recepción: <u>03/11/1999</u> Hora: <u>09:30 a.m.</u>	
Solicitud N° <u>9</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio	() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno
Costo estimado del cambio en horas: _____	
Comentarios:	Fecha de decisión: <u>04/11/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>9</u>	Fecha de recepción: <u>05/11/1999</u> Hora: <u>10:00 p.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>6</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> • Se mejora la <u>funcionalidad e interfaz grafica la pantalla, ahora es más agradable</u> • El formato de la fecha se maneja con (/ /). Si el usuario se equivoca al teclear el formato, aparecerá un Mensaje de Error indicándole cual es el formato correcto que deberá introducir en dicho campo • Se agrego el uso de las teclas TAB y ENTER a los botones Aceptar y Cancelar, además se cambio la etiqueta Aceptar por Salvar. • Los cambios que se efectuaron en esta pantalla, afectaron la tabla Activity Para llevar a cabo los cambios solicitados fue necesario importar la clase <u>NewActivityCommand</u> Así como, hacer uso de las clases <u>MainWindow, Database, RaiseError, Project, ManageTime</u> y <u>Activity</u> Se genero la clase <u>NewActivityWindow</u> , las variables utilizadas en la clase <u>Command</u> se estandarizaron con el nombre de las variables declaradas para esta pantalla	
Fecha de inicio: <u>05/11/1999</u>	Fecha de terminación: <u>15/11/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 9 Fecha de recepción: 16/11/1999
Hora: 10:30 a.m.

Autoridad responsable. Seguridad y Calidad

Comentarios.
 Los cambios implementados se probaron y se validaron principalmente los campos de formato de fecha para verificar que aceptar el formato propuesto (dd/mm/aaaa). Después de efectuar las pruebas y de corregir los algunos pequeños problemas. Se liberó la pantalla se presentar ninguna anomalía.

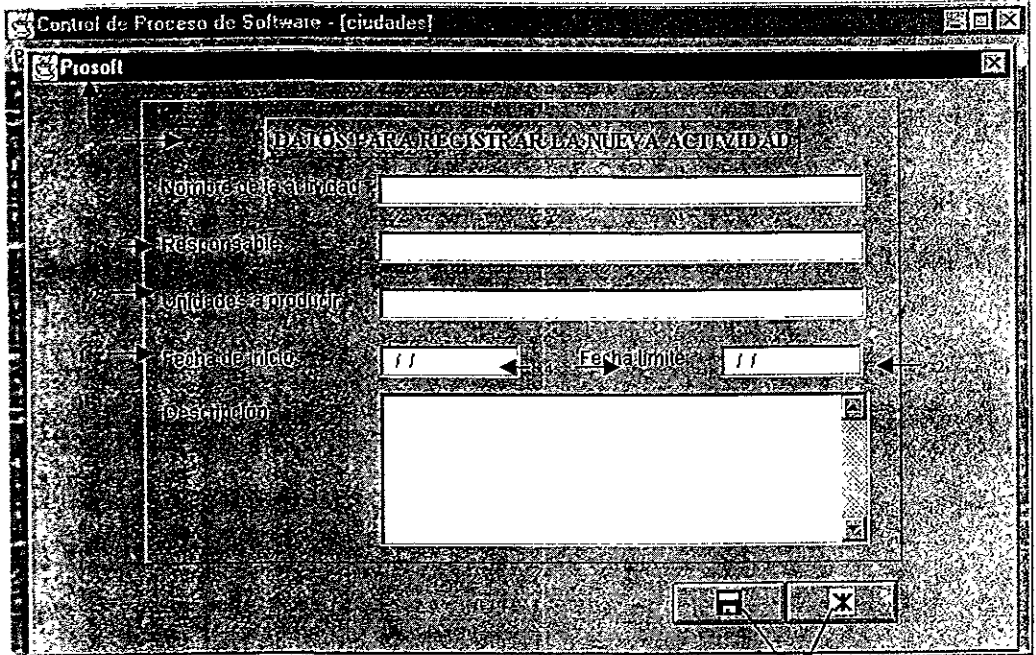
Fecha de liberación: 17/11/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Actividad nueva una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N°9

Los cambios efectuados fueron

- 1 Se agregaron las etiquetas y los campos *Responsable* y *Fecha de inicio*
- 2 Se cambiaron las etiquetas *Unidad* por *Unidades a producir*, y *Fecha de término* por *Fecha limite*
- 3 Se cambiaron las etiquetas de los botones por imágenes que representan la función que estos botones realizan
- 4 Se cambio y se valido el formato de fecha
- 5 Se elimino el nombre de la pantalla de la barra de titulo y en su lugar aparece el nombre de Prosoft. Y el nombre de la pantalla aparece en la parte superior de la pantalla en color azul
- 6 Se modifiko y actualizo la tabla Activity



Solicitud N° 10.

Solicitud de Cambio

Solicitud N° 10

Fecha de solicitud: 06/01/2000

Hora: 09:30 a m

Propuesto por: Solicitante 1 y Solicitante 2

Versión: 1.0

Sistema/Proyecto: Control del Proceso de Software

Versión: 1.0

Componente a ser cambiado: Pantalla Registro Detallado de Actividad

Descripción del cambio:

- 1 Eliminar el nombre del proyecto que se despliega en la barra de título
- 2 El tamaño de esta pantalla deberá abarcar el área de trabajo de la pantalla principal para que sea uniforme a las otras pantallas
- 3 Eliminar el campo y etiqueta Actividad
- 4 Agregar en la parte superior una etiqueta de color azul con el título de la pantalla y enseguida se despliegue automáticamente el nombre de la actividad que se selecciono con anterioridad
- 5 Cambiar la etiqueta *Unidad* por *Unidades a producir*.
- 6 Cambiar la etiqueta *Tiempo Efectivo* por *Tiempo efectivo total*
- 7 Cambiar la etiqueta *Tiempo de Interrupción* por *Tiempo de interrupción total*
- 8 Agregar al final de los campos de tiempo la etiqueta [mins.]
- 9 Agregar la etiqueta *Información General*
- 10 Agregar el campo y la etiqueta *Responsable*
- 11 Cambiar la etiqueta *Descripción* por *Comentarios*
- 12 Eliminar el scrollbar horizontal del campo de Descripción. Todos los datos que en esta pantalla se despliega son solo de lectura, por lo tanto se propone cambiar el texto que se despliega en ellos de color amarillo y el fondo de azul oscuro
- 13 Implementar el uso de la tecla ENTER en el botón cerrar.

Razones para el cambio:

- Se propone la etiqueta Información General para diferenciar la información que aparece en la bitácora.
- Uniformizar el tamaño de la pantalla con las otras pantallas que el sistema despliega tales como de captura, reporte y ayuda
- Distinguir los datos proporcionados por el sistema por los datos capturados por el usuario.
- Se propone eliminar el scrollbar del campo Descripción para que la lectura sea más legible para el usuario

Firma del solicitante

Aprobación de Cambio

Fecha de recepción: 06/01/2000

Hora: 05:30 p.m.

Solicitud N° 10

Aprobada (X) Rechazada ()

Autorizado por: Mesa de Control de Cambios Virtual

Valoración del cambio: El cambio no repercute en ninguna parte del sistema.

Prioridad del cambio:

() Crítico () Muy importante (X) Importante () Interesante () Inoportuno

Costo estimado del cambio en horas: _____

Comentarios:

Fecha de decisión: 07/01/2000

Firma de la MCC

Resultado de la Implementación de Cambio

Solicitud N° 10

Fecha de recepción: 07/01/2000
 Hora: 06.30 p.m.

Ingeniero(s) de software: Programador 1 y Programador 2

N° total de cambios implementados en los componentes: 13

Versiones: 1

Descripción de los cambios:

- ♦ Los cambios se implementaron sin ningún problema y el sistema mejoró bastante.
- ♦ Se modificó y actualizó la tabla ActivityLine, por los cambios la pantalla sufrió.

Para llevar a cabo los cambios solicitados fue necesario importar la clase DetailRecordActivityCommand. Así como, hacer uso de las clases Database, RaiseError, Project, Activity, ActivityLine, ManageTime y se invocó a OpenActivityWindow. Se generó la clase DetailRecordActivityWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.

Fecha de inicio: 08/01/2000

Fecha de terminación: 10/01/2000

 Firma del Ingeniero de Software

Liberación de Cambio

Solicitud N° 10

Fecha de recepción: 11/01/2000
 Hora: 10:00 a.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Los cambios implementaron se probaron sin ningún problema, se logró ver la diferencia entre los datos proporcionados automáticamente por el sistema y los datos capturados por el usuario.

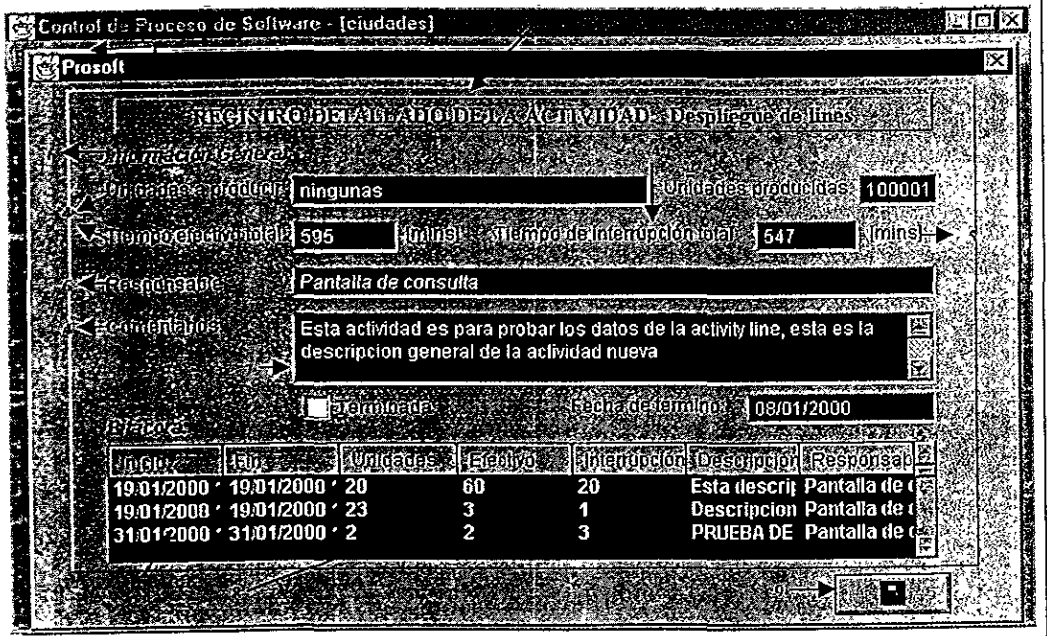
Fecha de liberación: 12/01/2000

 Firma del Depto. de Seguridad y Calidad
 (Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Registro detallado de la actividad una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 10

Los cambios efectuados fueron

- 1 Se eliminó de la barra de titulo el nombre de la pantalla y en su lugar aparece el nombre del grupo de desarrollo "Prosoft"
- 2 Se cambio por completo el diseño de la pantalla y el tamaño es uniforme con las otras pantallas que integran dicho sistema
- 3 Se eliminó el campo y etiqueta Actividad Se agrego en la parte superior una etiqueta de color azul con el titulo de la pantalla al final de esta etiqueta se despliega automáticamente el nombre de la actividad que se selecciono con anterioridad
- 4 Se cambiaron las etiquetas *Unidad* por *Unidades a producir*, *Tiempo Efectivo* por *Tiempo efectivo total*, *Tiempo de Interrupción* por *Tiempo de interrupción total* y *Descripción* por *Comentarios*
- 5 Se agrego la etiqueta [mins.] al final de los campos de tiempo
- 6 Se agregaron las etiquetas *Información General* y *Responsable*, esta última con su respectivo campo de texto
- 7 Se elimino el scrollbar horizontal del campo de Descripción (actualmente Comentarios).
- 8 Como todos los datos que en esta pantalla se despliega son solo de lectura, el texto se desplegará en color amarillo y el fondo de azul oscuro
- 9 Se implemento el uso la tecla Enter en el botón cerrar y se cambio la etiqueta cerrar por una imagen



Solicitud N° 11 y 12.

Solicitud de Cambio	
Solicitud N° <u>11 y 12</u>	Fecha de solicitud: <u>15/11/1999</u> Hora: <u>10:00 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Actualizar Actividad y Actualizar Manualmente la Actividad</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el nombre de las pantallas <i>Actualizar Actividad</i> por <i>Registrar automáticamente una actividad y Actualizar Manualmente la Actividad</i> por <i>Registrar manualmente una actividad</i> 2 Eliminar el nombre del proyecto que se despliega en la barra de título. 3 Cambiar las etiquetas <i>Unidad</i> por <i>Unidades a producir</i>, <i>Descripción</i> por <i>Descripción actual</i> 4 Manejar mensajes de error en los formatos de fecha y hora. El formato de fecha debe ser dd/mm/aaaa en lugar de yyyy/mm/dd 5 Agregar al final de los campos de tiempo la etiqueta [mins.] y agregar el campo y la etiqueta <i>Responsable</i> 6 Agregar en la parte superior una etiqueta de color azul con el título de la pantalla 7 El tamaño de esta pantalla deberá abarcar el área de trabajo de la pantalla principal para que sea uniforme a las otras pantallas 8 Implementar el uso de la tecla TAB y ENTER en el teclado para activar los botones <i>Aceptar</i> y <i>Cancelar</i>. Y cambiar la etiqueta del botón <i>Aceptar</i> por la imagen de salvar 9 Los datos que el sistema toma de la base de datos y despliega automáticamente que los muestre de color amarillo con fondo azul. 10 Validar los campos de captura con el uso de mensajes para que el usuario no deje campos en blanco 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Se propone ampliar la pantalla primero para uniformizar el tamaño con las otras pantallas y segundo porque la pantalla original es muy estrecha para la cantidad de datos que maneja. Se pretende distribuir los datos de manera que la pantalla no se vea cargada • Diferenciar los datos que el usuario captura con los datos proporcionados por el sistema • Se propone cambiar el nombre de algunas etiquetas para que el concepto quede más claro para el usuario. • Se propone cambiar el nombre de la pantalla <i>Actualizar</i> por <i>Registrar</i>, porque su función es registrar (no actualizar) las subactividades que integran una actividad 	
<p>_____</p> <p>Firma del solicitante</p>	

Aprobación de Cambio	
Fecha de recepción: <u>16/11/1999</u> Hora: <u>10:00 p.m.</u>	
Solicitud N° <u>11 y 12</u>	Aprobada (X) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema.</u>	
Prioridad del cambio	
() Crítico (X) Muy importante () Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios:	
Fecha de decisión: <u>17/11/1999</u>	
<p>_____</p> <p>Firma de la MCC</p>	

Resultado de la Implementación de Cambio

Solicitud N° 11 y 12.

Fecha de recepción: 17/11/1999

Hora: 01.00 p.m.

Ingeniero(s) de software Programador 1 y Programador 2

Versiones 1

N° total de cambios implementados en los componentes 10

Descripción de los cambios

- Se validó el campo Unidades producidas para que solo acepte números.
- No permite que los campos de lectura sean modificados.
- Al invocar esta pantalla, el mouse se coloca en el campo Unidades producidas.
- Se eliminó el scrollbar horizontal del campo Descripción actual para permitir que la lectura sea legible. Sin embargo, el usuario no se puede salir de este campo por medio de las tecla TAB.
- Se validaron los campos de fecha, hora y tiempo, por medio de mensajes de error le indican al usuario que debe proporcionar los datos solicitados, o bien que el formato que introdujo no es el apropiado.

Para llevar a cabo los cambios solicitados fue necesario importar la clases ManualRecordActivityCommand y AutomaticRecordActivityCommand Así como, se hizo uso de las clases Database, RaiseError, Project, Activity, ActivityLine, ManageTime y se invoco a OpenActivityWindow. Se generaron las clases ManualRecordActivityWindow y AutomaticRecordActivityWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.

Fecha de inicio: 18/11/1999

Fecha de terminación: 23/11/1999

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 11 y 12.

Fecha de recepción: 24/11/1999

Hora: 06:30 p.m.

Autoridad responsable Seguridad y Calidad

Comentarios:

Después de realizar las pruebas necesarias y validar los campos, así como revisar que los datos se almacenaban correctamente en la base de datos, se procedió a la liberación de dicha pantalla.

Fecha de liberación: 26/11/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedó la Pantalla Registrar manualmente una actividad una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 12. No se ejemplifica la solicitud 11, pero los cambios son prácticamente los mismos

Los cambios efectuados fueron

- 1 Se elimino de la barra de titulo el nombre de la pantalla y en su lugar aparece el nombre del grupo de desarrollo "Prosoft"
- 2 Se cambio por completo el diseño de la pantalla y el tamaño es uniforme con las otras pantallas que integran dicho sistema
- 3 Se agrego en la parte superior una etiqueta de color azul con el título de la pantalla
- 4 Se cambiaron las etiquetas *Unidad por Unidades a producir* y *Descripción por Descripción actual*
- 5 Se agrego al final de los campos de tiempo la etiqueta {mins.}
- 6 Se agrego la etiqueta *Responsable* con su respectivo campo de texto
- 7 Se elimino el scrollbar horizontal del campo de Descripción (actualmente Descripción actual)
- 8 Los datos que el sistema toma de la base de datos (de lectura) se desplegará en color amarillo y el fondo de azul oscuro
- 9 Se implemento el uso la tecla Enter en el botón Salvar y Cancelar y se cambiaron las etiquetas imágenes
- 10 Se valido el campo Unidades producidas para que solo acepte números
- 11 Se cambio el formato de fecha y se valido el formato de fecha y hora, así como el de tiempo

6

9

Solicitud N° 13.

Solicitud de Cambio	
Solicitud N° <u>13</u>	Fecha de solicitud: <u>08/11/1999</u> Hora: <u>10:00 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión <u>1.0</u>
Componente a ser cambiado: <u>Opción Externa Pantalla Enviar Actividad (*)</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el diseño de la pantalla hacerla más agradable y uniforme a las demás pantallas 2 Agregar una etiqueta que indique la función que desempeña dicha pantalla 3 Agregar una etiqueta que pregunte ¿Está seguro que quiere enviar (o recibir) las actividades?. 4 Implementar el uso de las teclas TAB y ENTER en los botones <i>Aceptar</i> y <i>Cancelar</i>. Y además, cambiar las etiquetas de los botones por imágenes 	
Razones para el cambio:	
<ul style="list-style-type: none"> ♦ Se propone cambiar el diseño de la pantalla para que quede de acorde al sistema 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	
(*) Las subopciones Recibir Actividad y Configuración del puerto pertenecen a la opción Externa Solamente se va a explicar Enviar Actividad porque esta a igual que las otras subopciones no fueron implementadas en el sistema original ni en este nuevo sistema	

Aprobación de Cambio	
Solicitud N° <u>13</u>	Fecha de recepción: <u>08/11/1999</u> Hora: <u>11:00 a.m.</u>
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada ()
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema</u>	
Prioridad del cambio	() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno
Costo estimado del cambio en horas: _____	
Comentarios:	Fecha de decisión: <u>08/11/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N°: <u>13</u>	Fecha de recepción: <u>08/11/1999</u> Hora: <u>01:00 a.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>4</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> ♦ Esta pantalla esta al igual que Recibir Actividad y Configuración de puerto están en construcción porque no se diseño el dispositivo externo para poder hacer la transferencia de actividades ♦ Sin embargo se hizo un diseño para avisarle al usuario que estas opciones podrían ser implementadas en una version posterior 	
Fecha de inicio: <u>08/11/1999</u>	Fecha de terminación: <u>09/11/1999</u>
<hr style="width: 20%; margin-left: auto;"/> Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N°. 13

Fecha de recepción: 09/11/1999

Hora: 02:00 p.m.

Autoridad responsable Seguridad y Calidad

Comentarios:

Esta pantalla realmente no se libero, queda pendiente para una nueva versión. Sin embargo se trato de implementar hasta donde fue posible

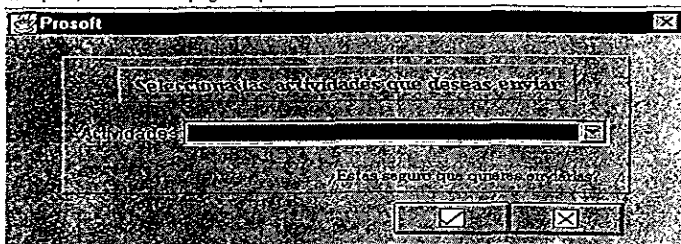
Fecha de liberación. 09/11/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

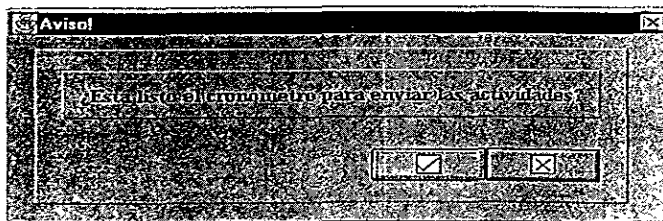
Esta imagen muestra como quedo la Pantalla Enviar Actividad una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 13

Los cambios efectuados fueron

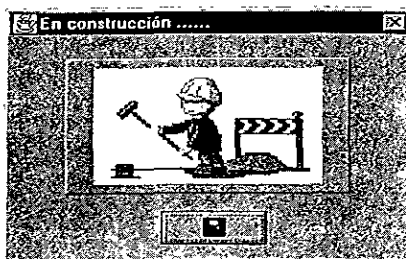
- 1 Se eliminó de la barra de titulo el nombre de la pantalla y en su lugar aparece el nombre del grupo de desarrollo "Prosoft"
- 2 Se cambio por completo el diseño de la pantalla y el tamaño es uniforme con las otras pantallas que integran dicho sistema
- 3 Se agrego en la parte superior una etiqueta de color azul con la función que la pantalla desempeña
- 4 Se agrego una etiqueta de color azul en la cual se pregunta al usuario si quiere enviar las actividades.
- 5 Si el usuario contesta que si, entonces se desplegara la pantalla Aviso!



A su vez esta pantalla pregunta si el cronómetro esta listo para enviar o recibir las actividades



Si la respuesta es si, esta pantalla debería activar el dispositivo externo, pero en su lugar muestra la pantalla llamada "En construcción". Lo mismo sucede para el caso del Recibir actividades y configuración de puerto



Solicitud N° 14.

Solicitud de Cambio	
Solicitud N° <u>14</u>	Fecha de solicitud: <u>13/01/2000</u> Hora: <u>09:30 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	
Sistema/Proyecto: <u>Control del Proceso de Software</u>	
Componente a ser cambiado Opción Externa <u>Pantalla Selección de Defecto.</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el diseño de la pantalla haréla más agradable y uniforme a las demás pantallas 2 Cambiar el nombre de la pantalla <i>Selección de Defecto</i> por <i>Abrir un defecto.</i> 3 Agregar una etiqueta que indique la función que desempeña esta pantalla 4 Cambiar la etiqueta <i>Defecto</i> por <i>Corregir defecto</i> 5 Implementar el uso de las teclas TAB y ENTER en el teclado, para activar los botones <i>Aceptar</i> y <i>Cancelar</i> Además, cambiar el nombre de las etiquetas de los botones por imágenes. 6 Implementar la función del botón <i>Aceptar</i> 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Mejorar la funcionalidad e interfaz humana del sistema, implementado la función del botón <i>Aceptar</i> y de las pantallas que faltaron para que funcione apropiadamente la opción <i>Corregir</i> Para ello es necesario. • Crear y diseñar la pantalla <i>Ejecutar un defecto</i> • Crear y diseñar la pantalla <i>Registrar automáticamente un defecto</i> • Crear y diseñar la pantalla <i>Registrar manualmente un defecto.</i> 	
Firma del solicitante	

Aprobación de Cambio	
Solicitud N° <u>14</u>	Fecha de recepción: <u>13/01/2000</u> Hora: <u>11 00 a.m.</u>
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema</u>	
Prioridad del cambio	
<input type="checkbox"/> Crítico <input checked="" type="checkbox"/> Muy importante <input type="checkbox"/> Importante <input type="checkbox"/> Interesante <input type="checkbox"/> Inoportuno	
Costo estimado del cambio en horas _____	
Comentarios	
Fecha de decisión: <u>13/01/2000</u>	
Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>14</u>	Fecha de recepción: <u>13/01/2000</u> Hora: <u>02:00 p.m.</u>
Ingeniero(s) de software <u>Programador 1 y Programador 2</u>	
N° total de cambios implementados en los componentes. <u>6</u>	
Versiones: <u>1</u>	
Comentarios	
<ul style="list-style-type: none"> • Para realizar los cambios propuestos hubo necesidad de actualizar y modificar la tabla DefectLine y Defect. • Se implementaron las pantallas faltantes <ol style="list-style-type: none"> 1 Ejecutar un defecto 2 Registrar automáticamente un defecto 3 Registrar manualmente un defecto • El diseño y funcionalidad es similar a las pantallas <i>Ejecutar</i>, <i>Registrar automáticamente</i> y <i>manualmente</i> una actividad. 	
Fecha de inicio. <u>14/01/2000</u>	Fecha de terminación: <u>21/01/2000</u>
Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 14

Fecha de recepción: 21/01/2000

Hora: 07:00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Respecto al módulo Defecto no esta funcionando en su totalidad Pero se logro implementar los cambios sin ningún problema y mejoro mucho la funcionalidad del sistema.

Fecha de liberación: 23/01/2000

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

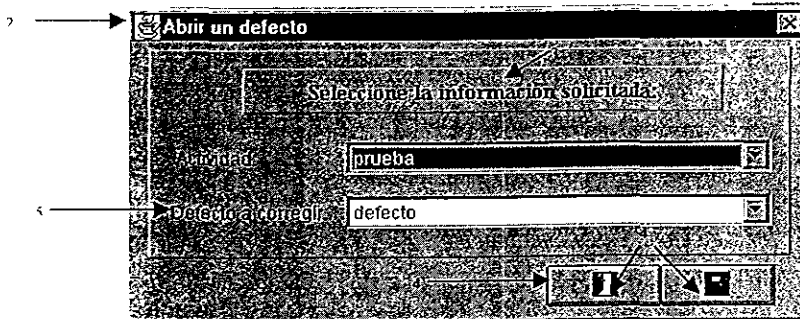
Esta imagen muestra como quedo la Pantalla Abrir un defecto una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 14

Para que la opción Corregir del módulo Defecto funcione correctamente es necesario implementar primero las pantallas *Ejecutar un defecto*, *Registrar automaticamente un defecto* y *Registrar manualmente un defecto*

Como se pudo observar en la pantalla Principal la opción Corregir se encuentra integrada por dos subopciones que son Automática y Manual Al seleccionar cualquiera de las opciones entonces se desplegará la pantalla Abrir un defecto

Los cambios que se le hicieron a esta pantalla (que en la versión 1.0 se llamaba Selección de Defecto) fueron

- 1 Se cambio por completo el diseño de la pantalla y el tamaño es uniforme con las otras pantallas que integran dicho sistema
- 2 Se cambio el nombre de la pantalla Selección de Defecto por Abrir un defecto
- 3 Se agrego en la parte superior una etiqueta de color azul con la función que la pantalla desempeña.
- 4 Se implemento la función del botón Aceptar y el uso de la tecla Enter
- 5 Se cambio la etiqueta Defecto por Defecto a corregir

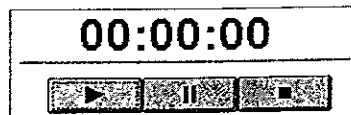


Si la opción seleccionada fue automática entonces se desplegará la pantalla Ejecutar un defecto. No se explica con mas detalle porque esta pantalla ha sido creada por primera vez para esta versión



Defecto: defecto

de la actividad: prueba



Registrar Defecto



Si el usuario detiene el reloj por medio del botón parar entonces el sistema deslizará la siguiente pantalla.

CONTROL DE PROCESO DE SOFTWARE - [ciudades]

Prossoft

RECONTAR AUTOMATICAMENTE UN DEFECTO

Actividad en la que se encontró:

Producto:

Base de datos con error: Base de datos de cambio:

Fecha de inicio: Fecha de término:

Hora de inicio: Hora de término:

Tiempo efectivo: (min) Tiempo de información: (min)

Que origen de defecto:

Corregido

Si ves este defecto mientras corre word, Selecciona el defecto que te interesa:

Pero si el usuario oprime el botón pausa y enseguida oprime el botón Registrar defecto el sistema deslizará la siguiente pantalla.

Prossoft

DATOS PARA REGISTRAR UN DEFECTO ENCONTRADO

Actividad en la cual se originó el defecto:

Tipo de defecto:

Producto:

Fecha de detección: Hora de detección:

Inyectado en la base: Remoción en la base:

Tiempo en verificación de la búsqueda del defecto: (min)

Comentarios:

Detectado por:

Una vez que el usuario proporcionó los datos solicitados y oprime el botón Aceptar, esta pantalla regresa el control a Ejecutar un defecto, para que el usuario continúe registrando los defectos encontrados previamente.

Si el usuario seleccionó la opción *Manual* del módulo *Corregir*, la pantalla que el sistema presenta es la siguiente

Solicitud N° 15

Solicitud de Cambio

Solicitud N° 15

Fecha de solicitud: 17/01/2000

Hora: 11:30 a.m.

Propuesto por: Solicitante 1 y Solicitante 2

Sistema/Proyecto: Control del Proceso de Software

Versión: 1.0

Componente a ser cambiado: Pantalla Nuevo Tipo de Defecto

Versión: 1.0

Descripción del cambio:

- 1 Cambiar el nombre de la pantalla Nuevo Tipo de Defecto por Tipo de Defecto
- 2 Eliminar el nombre del proyecto de la barra de título
- 3 Cambiar definitivamente el diseño de la pantalla porque la anterior no proporciona ni solicita realmente información apropiada

Razones para el cambio:

- ♦ Se propone ampliar la pantalla para uniformizar el tamaño con las demás pantallas que integran el sistema
- ♦ La pantalla original, el único dato que solicita es Descripción Y no hace referencia a ningún tipo de defecto Para solucionar este problema se propone incluir dos campos más que serían código y nombre del defecto, a través de una imagen mostrarle al usuario el uso y finalidad de esta pantalla

Firma del solicitante

Aprobación de Cambio	
	Fecha de recepción: <u>17/01/2000</u> Hora: <u>02:00 a.m.</u>
Solicitud N° <u>15</u>	Aprobada (X) Rechazada ()
Autorizado por <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio repercute en la tabla DefectType.</u>	
Prioridad del cambio () Crítico (X) Muy importante () Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios:	
	Fecha de decisión: <u>17/01/2000</u>
	_____ Firma de la MCC

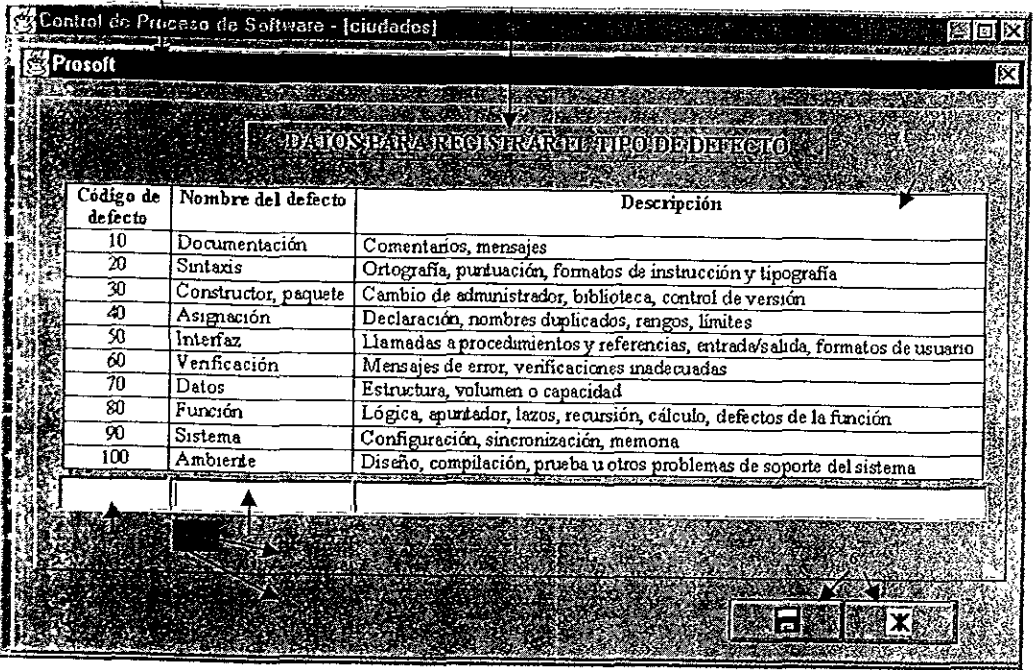
Resultado de la Implementación de Cambio	
	Fecha de recepción: <u>17/01/2000</u> Hora: <u>05:00 p.m.</u>
Solicitud N° <u>15</u>	Versiones: <u>1</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	
N° total de cambios implementados en los componentes: <u>3</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> * Los cambios se implementaron sin ningún problema. La pantalla quedó bastante explícita para el usuario, ya que podrá identificar el defecto por nombre o código. * Se modificó y actualizó la tabla DefectType. * Se agregaron los campos y etiquetas nombre y código del defecto. * Por medio de una imagen se simuló la tabla tipo de defecto, donde el usuario al final de esta imagen podrá capturar los datos del tipo de defecto, pero no se desplegarán en esta imagen. 	
Para llevar a cabo los cambios solicitados fue necesario importar la clase NewDefectTypeCommand. Así como, hacer uso de las clases MainWindow, Database, RaiseError, DefectType. Se generó la clase NewDefectTypeWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.	
Fecha de inicio: <u>18/01/2000</u>	Fecha de terminación: <u>20/01/2000</u>
	_____ Firma del Ingeniero de software

Liberación de Cambio	
	Fecha de recepción: <u>20/01/2000</u> Hora: <u>06:30 p.m.</u>
Solicitud N° <u>15</u>	
Autoridad responsable: <u>Seguridad y Calidad</u>	
Comentarios:	
Los datos que el usuario proporcione en la pantalla Tipo de defecto se almacenarán en la tabla pero no se desplegarán en dicha pantalla.	
	Fecha de liberación: <u>21/01/2000</u>
	_____ Firma del Depto. de Seguridad y Calidad (Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Tipo de defecto una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 15

Los cambios efectuados fueron

- 1 Se elimino de la barra de titulo el nombre de la pantalla y en su lugar aparece el nombre del grupo de desarrollo "Prosoft"
- 2 Se cambio por completo el diseño de la pantalla y el tamaño es uniforme con las otras pantallas que integran dicho sistema
- 3 Se agrego en la parte superior una etiqueta de color azul con la función que la pantalla desempeña
- 4 Se cambio el nombre de la pantalla *Nuevo Tipo de Defecto por Tipo de defecto*
- 5 Se agregaron las etiquetas *Código de defecto* y *Nombre del defecto*.
- 6 Se implemento el uso de la tecla Enter en los botones Salvar y Cancelar.
- 7 Para ejemplificar el funcionamiento de esta pantalla se coloca una imagen de la tabla de tipo de defecto



Nota: Se creo la pantalla *Registro detallado del defecto* que pertenece a la opción *Ver registro* del módulo *Defecto*, no se explican los cambios ni se ejemplifica porque es similar a la pantalla *Registro detallado de la actividad*

Solicitud N° 16.

Solicitud de Cambio	
Solicitud N° <u>16</u>	Fecha de solicitud: <u>31/01/2000</u> Hora: <u>09:30 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Reporte</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el diseño de la pantalla 2 Agregar una etiqueta de color azul en la cual se despliegue automáticamente el nombre del proyecto 3 Agregar una etiqueta que le indique al usuario cuáles son las opciones que puede realizar en esta pantalla. 4 Cambiar el nombre de las etiquetas de los botones por imágenes que representen la función que estos desempeñan 	
Razones para el cambio:	
<ul style="list-style-type: none"> • La pantalla de la versión 1.0 muestra los botones en el área de trabajo y en la barra de título despliega el nombre del proyecto que se desea imprimir. Sin embargo los datos que esta pantalla presenta son los mismos, lo único que se pretende cambiar es el diseño y la presentación de los datos 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
	Fecha de recepción: <u>31/01/2000</u> Hora: <u>12:00 p.m.</u>
Solicitud N° <u>16</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema</u>	
Prioridad del cambio	
() Crítico () Muy importante (<input checked="" type="checkbox"/>) Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios:	
Fecha de decisión: <u>01/02/2000</u> <hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>16</u>	Fecha de recepción: <u>01/02/2000</u> Hora: <u>12:00 p.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>4</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> • Los cambios se implementaron sin ningún problema. • Se implementó el uso de la tecla Enter y Tab en los botones y se cambiaron las etiquetas por imágenes. Para llevar a cabo los cambios solicitados fue necesario importar la clase ProjectReportCommand. Así como, se hizo uso de las clases Database, RaiseError, Project, ProjectReport, Report. Se generó la clase ProjectReportWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla.	
Fecha de inicio: <u>02/02/2000</u>	Fecha de terminación: <u>05/02/2000</u>
<hr style="width: 20%; margin-left: auto;"/> Firma del Ingeniero de software	
Nota: Para el reporte de actividad se importó la clase ActivityReportCommand y se hizo uso de las clases Database, Project, Activity, RaiseError, ActivityReport, Report. Para el reporte de defecto se importó la clase DefectReportCommand y se hizo uso de las clases Database, Project, Activity, Raise Error, DefectReport.	

Libерación de Cambio

Solicitud N° 16

Fecha de recepción: 05/02/2000

Hora: 05:30 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

Después de hacer varias pruebas se decidió liberar los formatos de reporte.

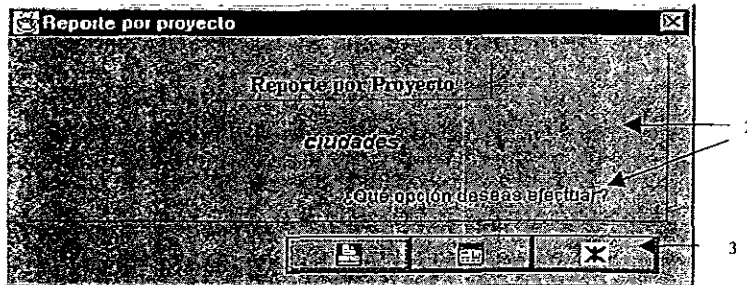
Fecha de liberación: 06/02/2000

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

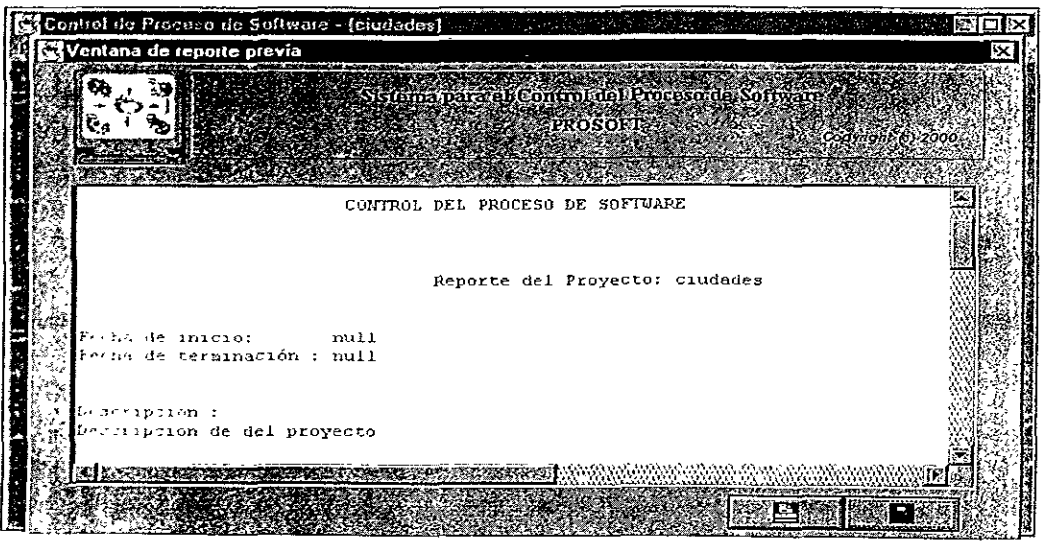
Esta imagen muestra como quedo la Pantalla Reporte por Proyecto una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 16

Los cambios efectuados fueron

1. Se cambio el diseño y presentación de los datos de la pantalla
2. Se agregaron dos etiquetas en la primera se despliega automáticamente el nombre del proyecto que se va a imprimir y en la segunda le pregunta al usuario que opción desea realizar
3. Se implemento el uso de las teclas ENTER y TAB en el teclado para activar los botones



Si el usuario oprime el botón de previo se desplegara la ventana de reporte previo como la que se muestra abajo.



Solicitud N° 17.

Solicitud de Cambio	
Solicitud N° <u>17</u>	Fecha de solicitud: <u>20/12/1999</u> Hora: <u>10:00 p.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Modificar Contraseña</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar la presentación y diseño de la pantalla 2 Cambiar el nombre de las etiquetas <i>Clave</i> por <i>Login</i>, <i>Confirmación</i> por <i>Confirmar contraseña</i> 3 Agregar el nombre de la etiqueta y el campo <i>Nueva contraseña</i> 4 Cambiar el nombre del botón <i>Aceptar</i> por <i>Salvar</i> y representar los nombres de las etiquetas con imágenes 5 Implementar el uso de las teclas TAB y ENTER en el teclado para activar ambos botones 	
Razones para el cambio:	
<ul style="list-style-type: none"> ◆ Hacer la pantalla más dinámica para el usuario. ◆ Quedaría más clara esta pantalla si el usuario introduce su contraseña actual, enseguida teclea su nueva contraseña y entonces confirme si esta correcta 	
Firma del solicitante	

Aprobación de Cambio	
Solicitud N° <u>17</u>	Fecha de recepción: <u>20/12/1999</u> Hora: <u>12:00 p.m.</u>
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	Aprobada (X) Rechazada ()
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema, solo en la tabla User.</u>	
Prioridad del cambio	() Crítico () Muy importante (X) Importante () Interesante () Inoportuno
Costo estimado del cambio en horas: _____	
Comentarios:	
Fecha de decisión: <u>20/12/1999</u>	
Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>17</u>	Fecha de recepción: <u>20/12/1999</u> Hora: <u>01:00 p.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>5</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> ◆ <u>Los cambios se lograron implementar sin ningún problema.</u> ◆ <u>Solo hubo necesidad de modificar y actualizar la tabla User.</u> <p>Para llevar a cabo los cambios solicitados fue necesario importar la clase <u>PasswordModifyCommand</u>. Así como, se hizo uso de las clases <u>MainWindow</u>, <u>Scope</u>, <u>User</u>, <u>RaiseError</u>. Se genero la clase <u>PasswordModifyWindow</u>, las variables utilizadas en la clase <u>Command</u> se estandarizaron con el nombre de las variables declaradas para esta pantalla.</p>	
Fecha de inicio: <u>20/12/1999</u>	Fecha de terminación <u>22/12/1999</u>
Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 17

Fecha de recepción: 23/12/1999

Hora: 12.00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios:

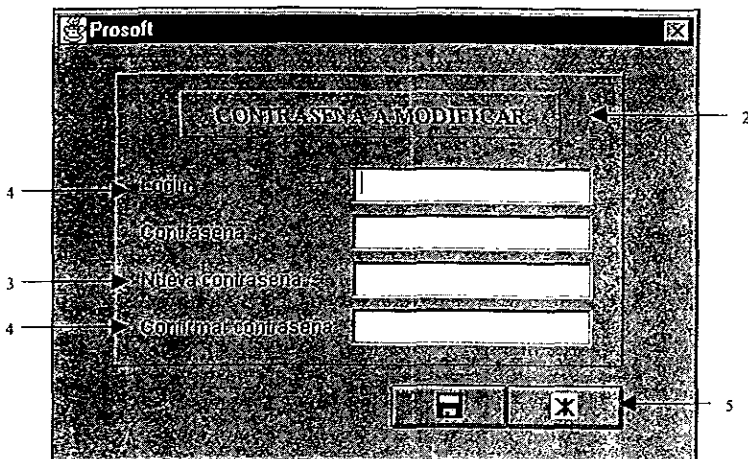
Los cambios implementados se probaron sin ningún problema, se reviso que el usuario pudiera modificar su contraseña si que el sistema marcar errores fuera de lo normal.

Fecha de liberación: 23/12/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Cambiar contraseña una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 17.

- Los cambios efectuados fueron
- 1 Se cambio el diseño y presentación de los datos de la pantalla
 - 2 Se agrego en la parte superior una etiqueta de color azul con el nombre de la pantalla.
 - 3 Se agregaron la etiqueta y el campo *Nueva contraseña*
 - 4 Se cambiaron los nombres de las etiquetas *Clave por Login*, *Confirmación por Confirmar contraseña*
 - 5 Se implemento el uso de la tecla Enter en los botones



Solicitud N° 18

Solicitud de Cambio	
Solicitud N° <u>18</u>	Fecha de solicitud: <u>07/02/2000</u> Hora: <u>09:30 a.m.</u>
Propuesto por: <u>Solicitante 1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado. <u>Pantalla Ayuda</u>	
Descripción del cambio:	
<ol style="list-style-type: none"> 1 Cambiar el nombre de la pantalla <i>Ayuda</i> por <i>Ayuda de PROSOFT (Tópicos)</i> 2 Cambiar por completo el diseño y presentación de la pantalla y de los datos que en ella se presentan 3 Implementar dicha pantalla, los cambios que se proponen son <ul style="list-style-type: none"> • Mostrar la ayuda a través del componente TabPanel, en cada una de las pestañas se explique y ejemplifique el funcionamiento de los módulos que integran el sistema. • Que le permita al usuario moverse de una opción a otro por medio de las pestañas, le permita mover a través del contenido de cada opción con el scroll vertical 	
Razones para el cambio:	
<ul style="list-style-type: none"> • Terminar e implementar esta opción, que fue propuesta en la versión 1.0 	
<hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma del solicitante	

Aprobación de Cambio	
Solicitud N° <u>18</u>	Fecha de recepción: <u>07/02/2000</u> Hora: <u>11:00 a.m.</u>
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada (<input type="checkbox"/>)
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema</u>	
Prioridad del cambio () Crítico (<input checked="" type="checkbox"/>) Muy importante () Importante () Interesante () Inoportuno	
Costo estimado del cambio en horas: _____	
Comentarios	Fecha de decisión: <u>08/02/2000</u>
<hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>18</u>	Fecha de recepción: <u>08/02/2000</u> Hora: <u>12:00 a.m.</u>
Ingeniero(s) de software: <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>3</u>	
Descripción de los cambios.	
<ul style="list-style-type: none"> • <u>Los cambios se implementaron sin ningún problema. La pantalla quedó bastante explícita para que el usuario pueda consultar sus dudas y pueda consultarla cada vez que lo requiera.</u> 	
Fecha de inicio: <u>09/02/2000</u>	Fecha de terminación: <u>20/02/2000</u>
<hr style="width: 20%; margin-left: auto; margin-right: 0;"/> Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 18

Fecha de recepción. 20/02/2000

Hora 06:30 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios: La pantalla esta bastante accesible para que el usuario pueda consultarla y sea capaz de iniciar un proyecto sin ningún problema

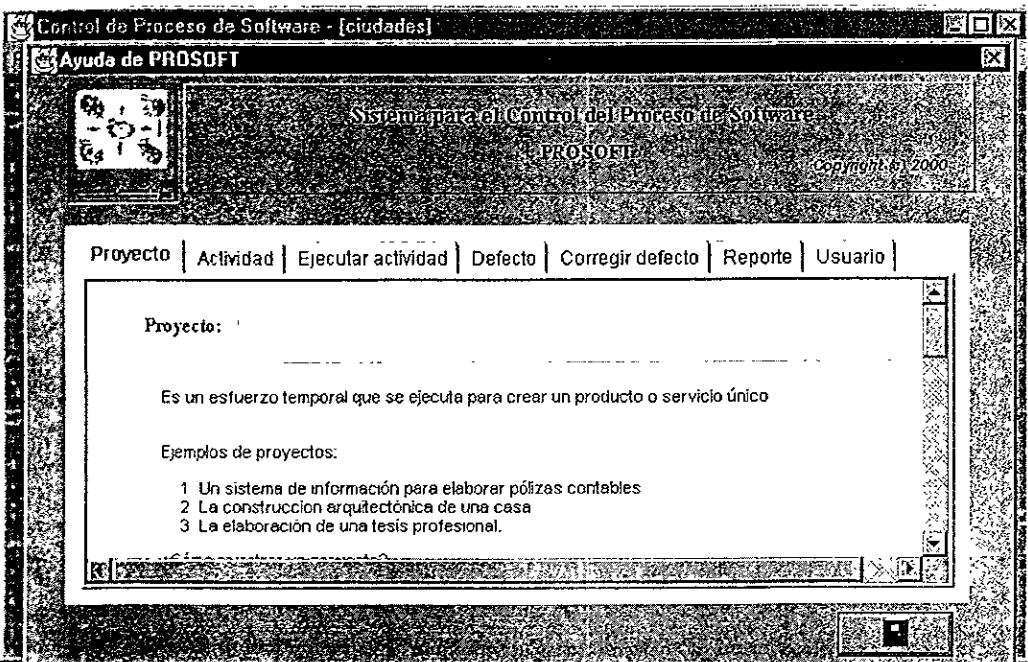
Fecha de liberación. 21/02/2000

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Ayuda de PROSOFT (Tópicos) una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 18

Los cambios efectuados fueron

- 1 Se diseño y se implemento la ayuda
- Si comozaran el formato de registro de defecto con la solicitud de cambio, podrá observar los cambios fueron satisfactorios.



Solicitud N° 19

Solicitud de Cambio	
Solicitud N° <u>19</u>	Fecha de solicitud: <u>07/02/2000</u> Hora: <u>10:00 a.m.</u>
Propuesto por <u>Solicitante1 y Solicitante 2</u>	Versión: <u>1.0</u>
Sistema/Proyecto: <u>Control del Proceso de Software</u>	Versión: <u>1.0</u>
Componente a ser cambiado: <u>Pantalla Acerca de</u>	
Descripción del cambio:	
1 Cambiar el logotipo de la versión anterior por el nuevo logotipo propuesto en la pantalla de Presentación	
2 Cambiar los datos de los integrantes del grupo de desarrollo, así como la presentación de los datos	
Razones para el cambio:	
<ul style="list-style-type: none"> • Actualizar los datos y uniformizar el diseño de la pantalla con las otras pantallas que integran el sistema 	
<hr style="width: 20%; margin-left: auto;"/> Firma del solicitante	

Aprobación de Cambio	
	Fecha de recepción: <u>07/02/2000</u> Hora: <u>12:00 p.m.</u>
Solicitud N° <u>19</u>	Aprobada (<input checked="" type="checkbox"/>) Rechazada ()
Autorizado por: <u>Mesa de Control de Cambios Virtual</u>	
Valoración del cambio: <u>El cambio no repercute en ninguna parte del sistema</u>	
Prioridad del cambio	() Crítico () Muy importante () Importante (<input checked="" type="checkbox"/>) Interesante () Inoportuno
Costo estimado del cambio en horas: _____	
Comentarios:	Fecha de decisión: <u>07/02/2000</u>
<hr style="width: 20%; margin-left: auto;"/> Firma de la MCC	

Resultado de la Implementación de Cambio	
Solicitud N° <u>19</u>	Fecha de recepción: <u>07/02/2000</u> Hora: <u>12:30 p.m.</u>
Ingeniero(s) de software <u>Programador 1 y Programador 2</u>	Versiones: <u>1</u>
N° total de cambios implementados en los componentes: <u>2</u>	
Descripción de los cambios:	
<ul style="list-style-type: none"> • <u>1 os cambios se implementaron sin ningún problema</u> 	
Fecha de inicio: <u>07/02/2000</u>	Fecha de terminación: <u>07/02/2000</u>
<hr style="width: 20%; margin-left: auto;"/> Firma del Ingeniero de software	

Liberación de Cambio

Solicitud N° 19

Fecha de recepción: 07/02/2000

Hora: 01:30 p.m.

Autoridad responsable: Seguridad y Calidad

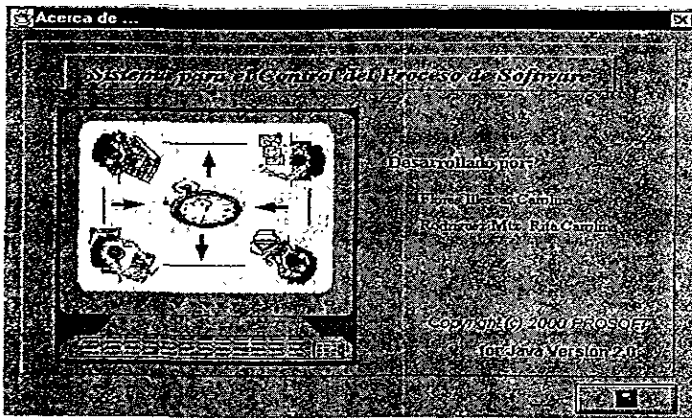
Comentarios:

Se libero sin presentar ningún problema.

Fecha de liberación: 07/02/2000

Firma del Depto de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

Esta imagen muestra como quedo la Pantalla Acerca de una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 19



Solicitud N° 20.

Solicitud de Cambio

Solicitud N° 20

Fecha de solicitud: 20/12/1999

Hora: 09:30 a.m.

Propuesto por: Solicitante 1 y Solicitante 2

Sistema/Proyecto: Control del Proceso de Software

Componente a ser cambiado: Usuario Nuevo

Versión: 1.0

Versión: 1.0

Descripción del cambio:

- 1 Agregar el campo *Confirmar contraseña*
- 2 Cambiar el nombre de las etiquetas *Nombre* por *Nombre del usuario*, *Clave* por *Login*
- 3 Cambiar el diseño de la pantalla para que sea uniforme a las demás.
- 4 Implementar el uso de la tecla Enter en ambos botones, así como cambiar el botón *Aceptar* por *Salir*

Razones para el cambio:

- ♦ Hacer la pantalla uniforme a las demás pantallas
- ♦ Los datos solicitados por esta pantalla estén mas claros. El administrador del sistema será el único que podrá crear cuentas, pero si él lo desea el usuario podrá elegir e introducir su contraseña.

Firma del solicitante

Aprobación de Cambio

Fecha de recepción: 20/12/1999

Hora: 11:00 a.m.

Solicitud N° 20

Aprobada () Rechazada ()

Autorizado por Mesa de Control de Cambios Virtual

Valoración del cambio. El cambio no repercute en ninguna parte del sistema, solo en la tabla User

Prioridad del cambio

() Crítico () Muy importante () Importante () Interesante () Inoportuno

Costo estimado del cambio en horas: _____

Comentarios

Fecha de decisión: 20/12/1999

Firma de la MCC

Resultado de la Implementación de Cambio

Solicitud N° 20

Fecha de recepción. 21/12/1999

Hora: 10:00 a.m.

Ingeniero(s) de software. Programador 1 y Programador 2

N° total de cambios implementados en los componentes: 4

Versiones: 1

Descripción de los cambios:

- ♦ Los cambios se lograron implementar sin ningún problema
- ♦ Solo hubo necesidad de modificar y actualizar la tabla User

Para llevar a cabo los cambios solicitados fue necesario importar la clase NewUserCommand. Así como, hacer uso de las clases Database, MainWindow, User y RaiseError. Se genero la clase NewUserWindow, las variables utilizadas en la clase Command se estandarizaron con el nombre de las variables declaradas para esta pantalla

Fecha de inicio 21/12/1999

Fecha de terminación: 22/12/1999

Firma del Ingeniero de software

Liberación de Cambio

Solicitud N° 20

Fecha de recepción: 22/12/1999

Hora: 06:00 p.m.

Autoridad responsable: Seguridad y Calidad

Comentarios Se implemento además la opción Eliminar usuario, está opción solo podrá ser usada por el administrador.

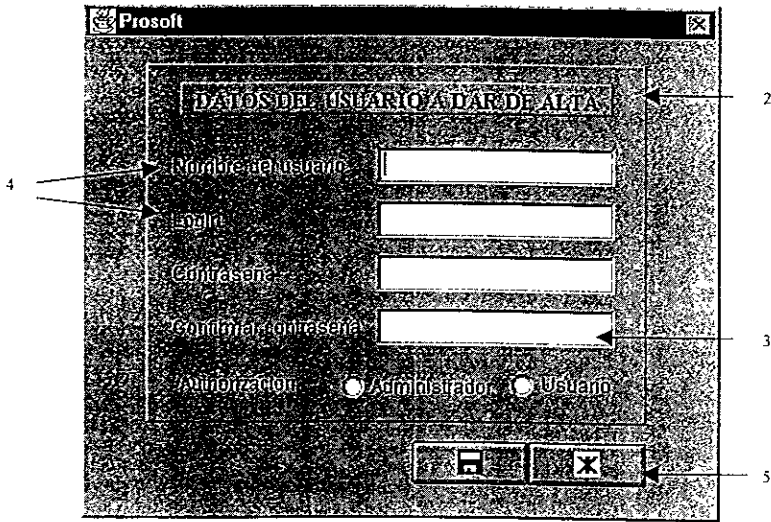
Fecha de liberación 23/12/1999

Firma del Depto. de Seguridad y Calidad
(Autoridad responsable de liberar el cambio)

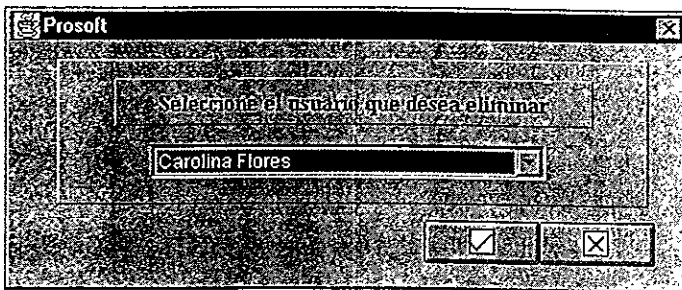
Esta imagen muestra como quedo la Pantalla Creación de un nuevo usuario una vez efectuados los cambios presentados a la MCC por medio de la solicitud de cambio N° 20

Los cambios efectuados fueron

- 1 Se cambio el diseño y presentación de los datos de la pantalla
- 2 Se agrego en la parte superior una etiqueta de color azul con el nombre de la pantalla
- 3 Se agregaron la etiqueta y el campo *Confirmar contraseña*
- 4 Se cambiaron los nombres de las etiquetas *Nombre por Nombre del usuario, Clave por Login*
- 5 Se implemento el uso de la tecla Enter en los botones.



Enseguida se muestra la opción *Eliminar usuario*, que solo podrá ser usada por el administrador del sistema. Es importante hacer notar que el administrador podrá cambiar su cuenta pero nunca eliminarla



BIBLIOGRAFÍA

[GECC 80] Prepared by General Electric Company Corporate Information System Briggport Connecticut, Software Engineering Handbook, McGraw-Hill Book Company, 1986.

[Bers 80] Bersoff, E.H, V.D Herderson and S.G. Siegel, Software Configuration Management, Prentice-Hall.

[MIL STD 480] U.S. Government, Department of Defense, Configuration Control Engineering Changes, Desviation, and Waiver (supersede MIL STD 480 30 Oct. 1968), 1978.

[Somm 92] Sommerville IAN, Software Engineering/IAN Sommerville, Addison-Wesley 1992.

[Babi 86] Babich W.A; Software Configuration Management, Addison -Wesley, 1986.

[IEEE-729 83] ANSI/IEEE Std 729'1983, *Glossary of Software Engineering Terminology*, IEEE, New York, 1988.

[IEEE Std 610.12-90] IEEE Standard Glossary of Software Engineering Terminology. IEEE, 1994.

[Rios 98] Rios Campos Anibal, *Administración de la Configuración del Software*, Tesis de Maestría, 1998.

[Capr 96] Capretz Luis Fernando, Capretz Miriam A. M. *Object-Oriented Software: Design and Maintenance*. Ed. World Scientific Publishing Co. Pte. Ltd. 1996

[Ibar 99] Ibarquengoitia González Ma. Guadalupe. *Memorias del 2º Encuentro Nacional de Computación, Taller de Ingeniería de Software*, Pachuca Hgo, septiembre 1999.

[Okta 99] Oktaba Hanna, Ibarquengoitia González Ma. Guadalupe. *Patrones para la Recolección de Métricas de Tiempo y Tamaño*, Rev. Soluciones Avanzadas N° 71, julio 1999 pag. 50-57

[Rios 99] Rios Campos Anibal, Ibarquengoitia González Ma. Guadalupe, *La administración de la configuración del software*, Rev. Soluciones Avanzadas N° 66, febrero 1999, pag 48-55

[Booc 96] Booch Grady *Análisis y Diseño Orientado a Objetos con Aplicaciones*, Addison-Wesley. 2da. Edición 1996.

[Hump 97] Humphrey, Watts S. *Introduction to the Personal Software Process*, Addison-Wesley, 1997.

[Hump 97]. Humphrey, Watts S *A Discipline for Software Engineering*, Addison-Wesley, 6th. Edición 1997.

[Leve 95] Leveson, Nancy G. *Safeware, System Safety and Computers*, Addison-Wesley, 1995.

[Kelln 88] M. I. Kellner, Representation Formalisms for Software Process Modeling, 4th International Software Process Workshop, ACM Press, 1988. Pags 93-96

[Burc 88] John G. Burch Jr, Felix R. Strater Jr. Sistemas de Información teoría y práctica, Limusa 1988.

[Quin 99] Gloria Quintanilla Integrando la Administración de Pruebas con la Administración del Software, curso de calidad, 1999.