



FACULTAD DE INGENIERÍA Y ARQUITECTURA
ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y SISTEMAS

**GESTIÓN DE LA CONFIGURACIÓN DE SOFTWARE PARA EL
BANCO DE LA NACIÓN ALINEADO A LA NORMA ISO 20000**

PRESENTADA POR

YOFRE KEVIN CORTEZ VÁSQUEZ

LOURDES LILIANA DÍAZ TABOADA

ASESOR

LUIS ESTEBAN PALACIOS QUICHIZ

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

LIMA – PERÚ

2015



CC BY

Reconocimiento

Los autores permiten a otros distribuir y transformar (traducir, adaptar o compilar) a partir de esta obra, incluso con fines comerciales, siempre que sea reconocida la autoría de la creación original

<http://creativecommons.org/licenses/by/4.0/>



USMP
UNIVERSIDAD DE
SAN MARTÍN DE PORRES

**FACULTAD DE
INGENIERÍA Y ARQUITECTURA**

**ESCUELA PROFESIONAL DE INGENIERÍA DE COMPUTACIÓN Y
SISTEMAS**

**GESTIÓN DE LA CONFIGURACIÓN DE SOFTWARE PARA EL
BANCO DE LA NACIÓN ALINEADO A LA NORMA ISO20000**

TESIS

**PARA OPTAR EL TÍTULO PROFESIONAL DE INGENIERO DE
COMPUTACIÓN Y SISTEMAS**

PRESENTADA POR

CORTEZ VÁSQUEZ, YOFRE KEVIN

DÍAZ TABOADA, LOURDES LILIANA

LIMA – PERÚ

2015

Dedico a mis padres, por su sacrificio abnegado, por su apoyo incondicional y toda su dedicación. Un gran ejemplo a seguir.

Yofre Cortez Vásquez

Dedicado a mis padres por todo su
esfuerzo, su confianza que me
otorgaron para lograr mis objetivos.

Lourdes Díaz Taboada

Agradezco a mis padres porque me guían en cada momento y me apoyan constantemente en el logro de mis metas.

Yofre Cortez Vásquez

Agradezco a mi familia, quienes han estado a mi lado en todo este proceso, apoyándome para lograr mis metas.

Lourdes Díaz Taboada

ÍNDICE

	Página
RESUMEN	x
ABSTRACT	xi
INTRODUCCIÓN	xii
CAPÍTULO I. MARCO TEÓRICO	
1.1 Antecedentes	1
1.2 Bases teóricas	5
1.3 Definición de términos básicos	20
CAPÍTULO II. METODOLOGÍA	
2.1 Materiales	24
2.2 Métodos	26
CAPÍTULO III. DESARROLLO DEL PROYECTO	
3.1 Planificación de la iteración	29
3.2 Ejecución de la iteración	35
3.3 Artefactos	40
3.4 Inspección y adaptación	44
3.5 Factores Críticos de éxito	44
3.6 Lecciones aprendidas	44
CAPÍTULO IV. PRUEBAS Y RESULTADOS	
4.1 Funcionalidad	45
4.2 Adaptación	46
4.3 Satisfacción	48

CAPÍTULO V. DISCUSIONES Y APLICACIONES	
5.1 Discusiones	51
5.2 Aplicaciones	55
CONCLUSIONES	56
RECOMENDACIONES	57
FUENTES DE INFORMACIÓN	58
ANEXOS	61

ÍNDICE DE TABLAS

	Página
Tabla 01. Principales Factores Críticos de éxito	10
Tabla 02. Cuadro comparativo de Metodologías de <i>Software</i>	19
Tabla 03. Requerimiento de Recursos Humanos	24
Tabla 04. Requerimiento de <i>hardware</i>	24
Tabla 05. Requerimiento de <i>software</i>	25
Tabla 06. Presupuesto	26
Tabla 07. Comparación de metodologías	27
Tabla 08. <i>Product backlog</i>	30
Tabla 09. <i>Sprint backlog</i>	31
Tabla 10. Lista de <i>Sprint</i>	34
Tabla 11. Requerimiento de Recursos Humanos	42
Tabla 12. Historias de usuario	46
Tabla 13. Aceptación del sistema	46
Tabla 14. Tabla de resultados de evaluación del sistema	47
Tabla 15. Satisfacción del sistema	48
Tabla 16. Tabla de satisfacción del usuario	49
Tabla 17. Mantenimientos atendidos por mes	52
Tabla 18. Tiempos de traslado de programas <i>host</i> por mes	53
Tabla 19. Reducción en tiempos (%) de traslado de programas <i>host</i>	54
Tabla 20. Uso del Sistema de gestión de la configuración del <i>software</i>	55

ÍNDICE DE FIGURAS

	Página
Figura 01. Sistema de Gestión de Servicio y los procesos de gestión	6
Figura 02. Ciclo de vida del servicio	8
Figura 03. Arquitectura de RUP	12
Figura 04. Registro de usuario	35
Figura 05. Autenticación de usuario	36
Figura 06. Búsqueda de Mantenimiento	36
Figura 07. Listado por tipo de programas	37
Figura 08. Traslado y versionado de programas	37
Figura 09. Log de ejecución	37
Figura 10. Cruce de programas	38
Figura 11. Flujo de estado de los mantenimientos	38
Figura 12. Actividades del mantenimiento	39
Figura 13. Historial del mantenimiento	39
Figura 14. Documentos del mantenimiento	40
Figura 15. Programas del Mantenimiento	40
Figura 16. Gráfica del producto	41
Figura 17. Gráfica de avance del producto	42
Figura 18. Gráfica de arquitectura de la aplicación	43
Figura 19. Evaluación de funcionalidades	47
Figura 20. Satisfacción del usuario	50
Figura 21. Programas por mes	52
Figura 22. Tiempos de traslados por mes	53

RESUMEN

La presente tesis consiste en la implementación de la gestión de la configuración de software para el Banco de la Nación con el objetivo de establecer un orden y correcta aplicación de procedimientos técnicos y administrativos a lo largo del ciclo de vida del software, alineándolo a la norma ISO20000. En su desarrollo fueron identificados los elementos de la gestión que deben considerarse durante el ciclo de vida del software, percibiendo la necesidad de tener un control de versiones del código fuente.

Se usó la metodología SCRUM, con la realización de un desarrollo incremental del sistema, estableciendo un conjunto de prácticas simples y evaluando las condiciones y características de los elementos y entidades identificados, como el equipo de trabajo y sus responsabilidades, la distribución de tareas, las actividades de integración y la herramienta de apoyo para la gestión de la configuración. Como resultado se implementó una solución a través del traslado y control de versiones. La investigación permite concluir que la aplicación de la gestión de la configuración ayudará al Banco de la Nación a mejorar el control de las versiones de software.

Palabras claves: ISO20000, Gestión de configuración de software, *Scrum*.

ABSTRACT

The project consists in the implementation of software configuration management for the National Bank (Banco de la Nación in Spanish) with the aim of establishing an order and correctly applying the technical and administrative procedures throughout the software lifecycle, aligning it with the norm ISO20000. In the development of the project were identified management elements to be considered during the software lifecycle, recognizing the need for version control of source code.

For this project, the SCRUM methodology was used, making an incremental system development, establishing a set of simple practices and evaluating the conditions and characteristics of the elements and entities identified as the team and their responsibilities, distribution of tasks, integration activities and support tool for configuration management. As a result, a configuration management solution was implemented through the transfer and version control. The research concludes indicating that the application of configuration management helps the National Bank to improve control of software versions.

Keywords: ISO20000, Software Configuration Management, Scrum

INTRODUCCIÓN

El problema que afronta el Banco de la Nación es que no dispone de un adecuado sistema de Gestión de la configuración de *Software*, el que debe seguir los procedimientos indicados en la directiva del ciclo de vida de éste para el desarrollo, certificación, operación y mantenimiento de los CI y servicios implementados en el Banco.

Hoy en día para garantizar la rentabilidad y la prestación de servicios y productos de calidad por parte de las organizaciones y las empresas, se requiere contar con unos servicios relacionados con las TIC, orientados al usuario y asociados a un proceso de mejora continua. Para brindar estos servicios, las empresas y organizaciones hacen uso de varios marcos, normas y estándares. De todos ellos ITIL (*Infrastructure Technology Library*), es el marco de trabajo más aceptado en todo el mundo.

En el curso de *ITIL® Foundation* (s. f.), se menciona que ITIL define y establece buenas prácticas para la Gestión de Servicios Informáticos relacionándolos con los procesos de negocio; sin embargo ITIL no es medible y puede ser implementado de muchas maneras. Es por ello que la Organización Internacional de Estandarización (ISO), establece una implementación efectiva y un planteamiento estructurado a través de las normas recogidas en ISO / IEC 20000, para desarrollar servicios de tecnología de la información, fiables en lo referente a la gestión de servicios de TI.

AENOR (2011), define que en los procesos establecidos por la ISO / IEC 20000, la gestión de la configuración es uno de los más relevantes, constituyéndose como un proceso clave que permite la identificación, control, mantenimiento y verificación de los elementos de configuración (*CI, Configuration Items*). Éstos representan los ejecutables, el código fuente, los modelos de datos y procesos, la documentación y cualquier elemento de la infraestructura que sea susceptible de ser gestionado.

De acuerdo a la Jornada sobre Gestión de la Configuración, TECNOBIT, (2012) de *Software SCM* por sus siglas en inglés (*Software Configuration Management*) es una especialización de la Gestión de configuración a todas las actividades en el sector del desarrollo de *software*, la cual busca aplicar procedimientos técnicos y administrativos a lo largo del ciclo de su vida para: identificar, definir y congelar elementos *software* en un sistema; controlar modificaciones y liberaciones de los mismos; registrar e informar su estado y peticiones de modificación; asegurar la completitud, consistencia, corrección de los elementos y controlar el almacenamiento, manipulación y entrega.

El problema identificado es la ineficiente "Gestión de Configuración de *Software*", la cual dificulta mantener un control a través de todo el ciclo de desarrollo de *software* en el Departamento de Informática del Banco de la Nación. El objetivo es gestionarlo adecuadamente para aumentar la calidad sobre el diseño, desarrollo, pruebas de productos *software*, reduciendo los riesgos y posibles fallas de implantación en los ambientes de Desarrollo hasta Producción.

Se ha considerado como objetivos específicos reducir los tiempos de despliegue estableciendo un orden y asegurando la correcta configuración de los productos. Reducir los riesgos a través del seguimiento y control de los elementos de configuración de tipo *software* relacionados con la ejecución de las pruebas realizadas para cada petición de servicio, cumpliendo con las funciones establecidas en la directiva del ciclo de vida de *software*. Asimismo, implementar el sistema de Gestión de Configuración del *Software* según lo estipulado en la Norma ISO/IEC20000. Por último, identificar, organizar y

controlar las modificaciones al producto mejorando su calidad, maximizando su productividad y minimizando los errores.

La presente tesis presenta 3 tipos de justificaciones: teórica, práctica y financiera.

En la **teórica** se dispone la gestión de configuración de *software*, alineada a la Norma Técnica Peruana ISO/IEC 20000 para la identificación de los elementos de configuración de *software*, traslado y control de versiones. Además, sirve como herramienta que permitirá organizar las actividades necesarias para mantener la integridad del producto, delimitar la responsabilidad proporcionando un registro de acciones, reducir los costes del ciclo de vida y proporcionar un entorno de trabajo estable.

En la **práctica** el sistema permitirá de una manera más ágil y sencilla, la administración y traslado de varios elementos de configuración de tipo *software* o individualmente desde el ambiente de Desarrollo hasta Producción. Además, contará con la posibilidad de realizar el seguimiento a cada mantenimiento. Por estos motivos se realizará la implementación del sistema de gestión de configuración de *software* que es de vital importancia para el Departamento de Informática del Banco de la Nación.

Por último, la **financiera**, de acuerdo a la investigación realizada se ha encontrado que en promedio el costo de una licencia para una herramienta es de S/. 13,900. Considerando que son 4 secciones de Desarrollo, 1 de Aseguramiento de Calidad y 1 de Operación y Control de Plataforma (Producción) se necesitarían como mínimo 6 licencias, por lo tanto, la inversión sería de S/. 83,400. Por lo tanto, se ha decidido implementar el sistema SCM-BN, el cual se desarrollará e implementará con recursos propios del Banco.

Para finalizar, la presente tesis está estructurada de la manera siguiente: Capítulo I: Marco teórico, en el que se describe la terminología de la gestión de la configuración que hace referencia a la norma ISO/IEC 20000. Capítulo II: Metodología, donde se emplea la investigación aplicada que tiene por objeto generar conocimiento en el sector productivo. Capítulo III: Desarrollo

del Proyecto, en el que se aplica la metodología *Scrum*. Capítulo IV: Pruebas y Resultados, en el que se muestra la satisfacción del usuario. Capítulo V: Discusiones y Aplicaciones, donde se consigue automatizar el versionado de todos los programas *host*.

CAPÍTULO I

MARCO TEÓRICO

En el presente capítulo se mencionan los conceptos relacionados a la solución que se pretende brindar en el Departamento de Informática, es decir se describe la terminología de la gestión de la configuración y en particular los conceptos relacionados a los que hace referencia la norma ISO/IEC 20000 que serán mencionados en la base teórica del proyecto que se está realizando.

1.1 Antecedentes

Según el artículo de la *Revista Cubana de Ingeniería* de Espinosa, M. (2012), considera lo siguiente:

La gestión de configuración de *software* abarca un conjunto de actividades y técnicas para iniciar, evaluar y controlar los cambios del producto de *software* durante y después del proceso de desarrollo. Haciendo énfasis en el control de la configuración dentro de la administración de producción de *software*. Entre sus principales funciones se encuentran el velar que exista: una documentación referente a los cambios realizados y productos que de alguna manera no ocasionen la ruptura de la integridad del *software*. De manera adicional, brinda garantía de la calidad del *software*, lo cual influye en todas las fases del proceso de Ingeniería de *Software*.

No tiene establecida métricas cuantitativas que evalúen los resultados o que permitan conocer el estado actual de un grupo de desarrollo o empresa. Sin embargo, se pueden evaluar los niveles de madurez alcanzados en esta disciplina utilizando métricas cualitativas. Estas métricas pueden ser preguntas estructuradas en función de los aspectos a evaluar.
(p.2)

En la tesis de Fernandez, S. & Osso, M. (2010) se menciona que, en el desarrollo de *software* los cambios, debidos principalmente a modificaciones de requisitos y fallos, son inevitables. Además, en la gran mayoría de los casos los cambios al *software* están debidamente justificados. Normalmente se trabaja en equipo por lo que es preciso llevar un control y registro de los cambios con el fin de reducir errores, aumentar la calidad y evitar los problemas que pueda acarrear una incorrecta sincronización en dichos cambios, al afectar a otros elementos del sistema o a las tareas realizadas por otros miembros del equipo de proyecto.

En el artículo en conferencia de Daniele, M., Uva, M., Martelloto, P. & Picco, G. (2010), se menciona que la gestión de la configuración del *software* es uno de los procesos clave para toda organización dedicada a la Ingeniería del *Software*, ya que posibilita una mejor organización del desarrollo y mantenimiento, facilitando el resto de procesos de producción.

Durante el proceso de construcción de un *software*, los cambios son inevitables. Los cambios provocan confusión e incertidumbre, sobre todo cuando no se han analizado o pronosticado correctamente. Es importante considerar ciertas modificaciones que pueden ocurrirle al *software* dentro de todo el proceso de ingeniería. El arte de coordinar el desarrollo de *software* para minimizar la confusión, se denomina gestión de la configuración. La gestión es el arte de identificar, organizar y controlar las modificaciones que sufre el *software*, la finalidad es maximizar la productividad minimizando las fallas.

Spillner, Linz, & Schaefer, (2014) mencionaron que, un sistema de *software* se compone de una multitud de componentes individuales que deben encajar juntos para asegurar la funcionalidad del sistema como un todo. Si la

administración de configuración no se realiza correctamente en un proyecto, se puede observar lo siguiente:

- Desarrolladores sobrescriben mutuamente las modificaciones de los demás en el código fuente u otros.
- Las actividades de integración son obstaculizadas, debido a que:
 - No está claro qué versión de código de un componente específico existe en el equipo de desarrollo y cuáles son las versiones actuales
 - No está claro qué versiones de varios componentes van de la mano y se pueden integrar a un subsistema grande.
 - Se utilizan diferentes versiones de compiladores y otras herramientas de desarrollo.
- Los análisis de problemas, corrección de fallos y pruebas de regresión son complicados debido a que:
 - Se desconoce dónde y por qué el código de un componente se cambió con respecto a una versión anterior.
 - No se sabe de qué archivos de código se origina un subsistema integrado concreto (código objeto).

La insuficiente gestión de configuración conduce a una serie de posibles problemas que perturban el proceso de desarrollo y prueba. Por ejemplo, si no está claro durante una prueba de nivel si los objetos de prueba examinados son la versión más reciente, las pruebas pierden rápidamente cualquier significado.

De acuerdo a *Business Innovatios* (2014), para las pruebas, la gestión de configuración debe abordar las siguientes cuestiones:

La gestión del *testware* y los resultados: Tenemos que saber cuál versión del *testware* fue usada para ejecutar las pruebas, si tenemos que decir por seguro lo que los resultados de las pruebas significan y lo que ellos significan.

La gestión de los objetos de pruebas: Para que los resultados de las pruebas sean significativos y escrutables hasta el nivel de los ítems individuales, tenemos que saber la

versión de cada unidad que haya estado presente en un objeto particular de pruebas.

La entrega de una versión única y conocida del objeto de prueba: Para ser capaz de comprender el significado de los resultados de las pruebas, tenemos que ser capaces de entregar una versión de las pruebas en el laboratorio de pruebas. (p.50)

En la tesis Pytel, (2011) mencionó que:

El objetivo de la gestión de la configuración es mantener la integridad de los productos que se obtienen a lo largo del desarrollo de los sistemas de información, garantizando que no se realizan cambios incontrolados y que todos los participantes en el desarrollo del sistema disponen de la versión adecuada de los productos que manejan.

La gestión de configuración se realiza durante todas las actividades asociadas al desarrollo del sistema, y continúa registrando los cambios hasta que este deja de utilizarse. Durante el desarrollo permite controlar el sistema como producto global, obtener informes sobre el estado de desarrollo en que se encuentra y reducir el número de errores de adaptación del sistema, lo que se traduce en un aumento de calidad del producto, de la satisfacción del cliente y, en consecuencia, de mejora de la organización. Además, al aportar información precisa para valorar el impacto de los cambios solicitados y reducir el tiempo de implementación de un cambio (tanto evolutivo como correctivo) facilita el mantenimiento del sistema. (p.92)

En el Anexo 1, podemos revisar el conocimiento del Banco de la Nación y de la Sección Calidad de Soluciones respecto a las funciones y las soluciones aplicadas enfocándose en la implementación de un sistema de gestión de configuración alineada a la norma ISO/IEC 20000.

1.2 Bases teóricas

1.2.1 Gestión de la Configuración

De acuerdo a la Norma Técnica Peruana (2006) es uno de los procesos de apoyo:

Gestión de Configuración es aplicar los procedimientos técnicos y administrativos a lo largo del ciclo de vida del *software* para: identificar, definir y establecer la línea base de los elementos *software* en un sistema; controlar modificaciones y *releases* de los elementos; registrar e informar del estado de los elementos y peticiones de modificación; asegurar la completitud, consistencia y corrección de los elementos; y controlar el almacenamiento, manipulación y entrega de los elementos.

Este proceso consta de implementación del proceso, identificación de la configuración, control de la configuración, determinación del estado de la configuración, evaluación de la configuración y gestión de *releases* y entrega. (p.6)

Para Merchan & Gomez (2011) la gestión de la configuración representa un elemento clave en el proceso de desarrollo de *software* ya que proporciona estabilidad a la producción de *software*, controla el cambio continuo y concurrente que viene con la evolución del producto de *software* y obliga a implementar estrategias de versionamiento.

1.2.2 ISO/ IEC 20000

AENOR (2011) indica que ISO/IEC 20000 es una norma que contiene un Sistema de Gestión del Servicio (SGS). Especifica al proveedor del servicio los requisitos para planificar, establecer, implementar, operar, monitorizar, revisar, mantener y mejorar un SGS. Los requisitos incluyen el diseño, transición, provisión, y la mejora de los servicios para satisfacer los requisitos de servicio.

La Figura 01 muestra el SGS, e incluye los procesos de gestión del servicio.

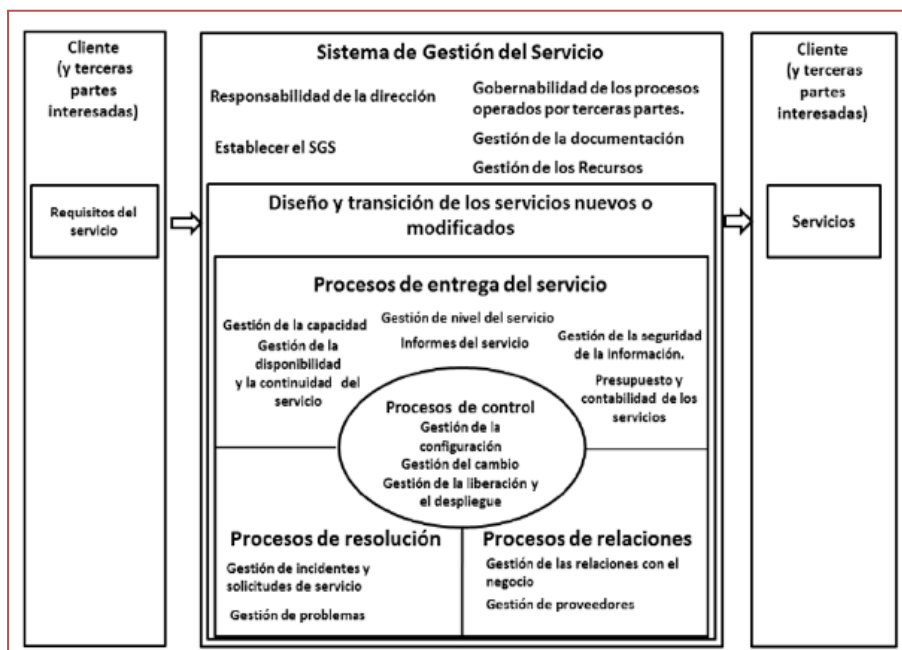


Figura 01. Sistema de Gestión de Servicio y los procesos de gestión
Fuente: Norma ISO/IEC 20000 Parte 1

Los procesos de gestión del servicio y las relaciones entre éstos se pueden implementar de diferentes formas por cada proveedor del servicio. La naturaleza de la relación entre cada proveedor del servicio y el cliente influirá en la forma de implementar los procesos de gestión del servicio.

Dentro de los procesos de control en AENOR (2007) se tiene por objetivo de la Gestión de la Configuración: definir y controlar los componentes del servicio y de la infraestructura. Asimismo, mantener actualizada la información de la configuración.

Se considera 5 divisiones:

Planificación e implementación: la gestión de la configuración se debe planificar con la gestión del cambio y de las entregas para asegurar que el proveedor del servicio pueda gestionar sus activos y configuraciones de TI de forma efectiva. Debería estar disponible una información precisa sobre la configuración para dar soporte a la planificación y al control de los cambios a medida que los sistemas y los servicios nuevos y modificados son liberados y distribuidos. Todos los activos y configuraciones principales se deberían tener en cuenta y tener un gestor responsable que asegure que se mantienen la protección y el control apropiados, por ejemplo: los cambios son autorizados antes de la implementación.

Identificación de configuración: todos los elementos de configuración deberían estar identificados de manera unívoca y estar definidos por atributos que describan sus características funcionales y físicas. La información debería ser relevante y auditable. En la base de datos de la configuración se deberían utilizar y registrar los marcados apropiados u otros métodos de identificación.

Control de la configuración: el proceso debería garantizar que solo los elementos de la configuración autorizados e identificables son aceptados y registrados desde su recepción hasta su baja. Ningún elemento de la configuración se debería añadir, modificar, reemplazar o eliminar/retirar sin la documentación de control apropiada, por ejemplo: aprobación de la solicitud de cambio, información actualizada de la versión.

Seguimiento del estado de configuración y elaboración de informes: los registros de la configuración se deberían mantener actualizados y con la precisión adecuada para reflejar los cambios en el estado, localización y versión de los elementos de la configuración. El seguimiento del estado debería proporcionar información sobre los datos actuales e históricos de cada elemento de configuración a lo largo de su ciclo de vida. Esto debería permitir el seguimiento de los cambios en los elementos de la configuración a través de sus diferentes estados, por ejemplo: solicitado, recibido, en pruebas de aceptación, activo, bajo cambio, retirado y eliminado.

Verificación y auditoría de la configuración: estos procesos, en sus aspectos físicos y funcionales, se deberían planificar y se debería realizar una comprobación para asegurar que los procesos y recursos adecuados están establecidos. Periódicamente se deberían realizar auditorías de la configuración, antes y después de un cambio importante, después de un desastre y a intervalos aleatorios.

En el libro De Pablos, C., Lopez, J., Romo, M., & Medina, S. (2012) indicaron: la ISO 2000-1 recoge las especificaciones en las cuales se describe la adopción de un proceso de mejora integrado para el desempeño y gestión de los servicios acorde a los requisitos del negocio y del cliente.

1.2.3 ITIL

En el curso *ITIL Foundation*, se menciona que ITIL puede ser definido como un conjunto de buenas prácticas destinadas a mejorar la gestión y provisión de servicios TI. Su objetivo último es mejorar la calidad de los servicios TI ofrecidos, evitar los problemas asociados a los mismos y en caso de que estos ocurran ofrecer un marco de actuación para que estos sean solucionados con el menor impacto y a la mayor brevedad posible.

ITIL fue desarrollada al reconocer que las organizaciones dependen cada vez más de la Informática para alcanzar sus objetivos corporativos. Esta dependencia en aumento ha dado como resultado una necesidad creciente de servicios informáticos de calidad que se correspondan con los objetivos del negocio, y que satisfagan los requisitos y las expectativas del cliente.

ITIL Foundation estructura la gestión de los servicios TI sobre el concepto de Ciclo de Vida de los Servicios. Este enfoque tiene como objetivo ofrecer una visión global de la vida de un servicio, desde su diseño hasta su eventual abandono, sin por ello ignorar los detalles de todos los procesos y funciones involucrados en la eficiente prestación del mismo.

El Ciclo de Vida del Servicio consta de cinco fases que se corresponden con los nuevos libros de ITIL, tal como se muestra en la figura 02:

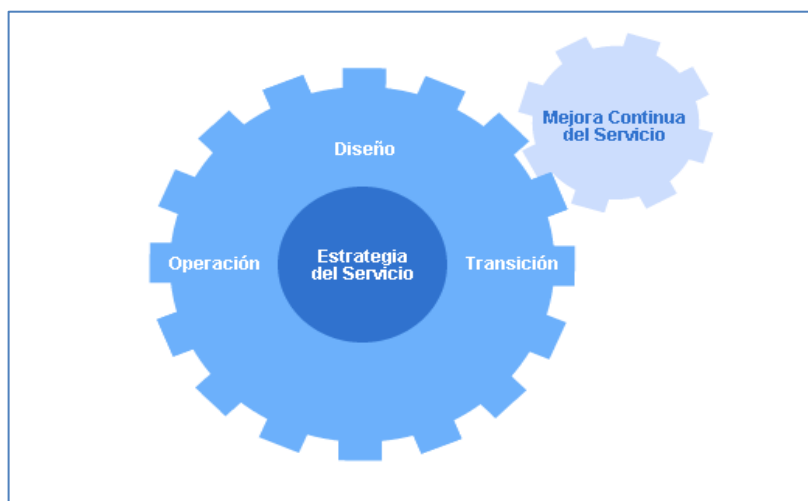


Figura 02. Ciclo de vida del servicio
Fuente: Portal de Osiatis

- Estrategia del Servicio: propone tratar la gestión de servicios no solo como una capacidad sino como un activo estratégico.
- Diseño del Servicio: cubre los principios y métodos necesarios para transformar los objetivos estratégicos en portafolios de servicios y activos.
- Transición del Servicio: cubre el proceso de transición para la implementación de nuevos servicios o su mejora.
- Operación del Servicio: cubre las mejores prácticas para la gestión del día a día en la operación del servicio.
- Mejora Continua del Servicio: proporciona una guía para la creación y mantenimiento del valor ofrecido a los clientes a traves de un diseño, transición y operación del servicio optimizado.

Estas cinco fases no son departamentos estancos e ITIL tiene en cuenta las múltiples interrelaciones entre ellos y como estas afectan a los aspectos globales de todo el ciclo de vida del servicio. Estos cinco libros ofrecen una guía práctica sobre como estructurar la Gestión de Servicios TI de forma que estos estén correctamente alineados con los procesos de negocio.

1.2.4 Gestión del Cambio

Para ITIL el principal objetivo de la Gestión de Cambios es la evaluación y planificación del proceso de cambio para asegurar que, si este se lleva a cabo, se haga de la forma más eficiente, siguiendo los procedimientos establecidos y asegurando en todo momento la calidad y continuidad del servicio TI. Siendo un proceso que utiliza herramientas y técnicas para gestionar la transición hacia una nueva realidad, intentando que las personas involucradas sean capaces y deseen trabajar en el nuevo contexto definido y así consigan los resultados esperados.

En ITIL la Gestión de Cambios debe trabajar para asegurar que los cambios:

- Están justificados
- Se llevan a cabo sin perjuicio de la calidad del servicio TI
- Están convenientemente registrados, clasificados y documentados

- Han sido cuidadosamente testeados en un entorno de prueba
- Se ven reflejados en la CMDB
- Pueden deshacerse mediante planes de "retirada del cambio" (*back-outs*) en caso de un incorrecto funcionamiento tras su implementación

Para Acosta (2012), “El cambio organizacional es el conjunto de transformaciones que se realizan en las distintas dimensiones de las organizaciones, es producido tanto por fuerzas naturales como impulsado por la voluntad de quienes las crean e impulsan.” (p.5)

1.2.5 Factores Críticos de Éxito

Además de seguir las normas y estándares que podemos encontrar en la literatura debemos poner especial atención a la institucionalización del cambio y esto solo se logra poniendo atención en los llamados “Factores Críticos de éxito”.

Según lo expresan Bayona, Calvo, Cuevas & San-Feliu (2012) en “Taxonomía de factores críticos para el despliegue de procesos *software*” desplegar procesos basados en cualquiera de los modelos y/o estándares para la mejora de procesos requiere de una estrategia para lograr el uso y la adopción de los nuevos procesos. Esta estrategia debe de estar basada en la gestión del cambio y enfocada fundamentalmente en las personas. Este enfoque facilitará el proceso de transición a los cambios que implica el despliegue de los nuevos procesos, y permitirá minimizar la resistencia a dichos cambios.

Podemos encontrar en la literatura diversos factores siendo los más resaltantes los observados en la tabla 01.

Tabla 01. Principales Factores Críticos de éxito

Nº	Factores Críticos
1	Compromiso de la alta dirección
2	Objetivos de mejora claros, relevantes y aplicables/objetivos medibles
3	Asignación de responsabilidades clara y compensada
4	Participación del personal /Iniciativas Bottom-up

5	Personas altamente respetadas
6	Tiempo del personal y recursos
7	Creación de equipos
8	Agentes de cambio y líderes de opinión
9	Fortalecimiento de la comunicación y la colaboración
10	Gestión del proceso de mejora (seguimiento)

Fuente: **Revista Española de Innovación, Calidad e Ingeniería del Software**, Vol.6, No. 3, 2010

1.2.6 RUP

En el artículo de *Anwar* (2014), se indicó que *Rational Unified Process (RUP)* es un proceso de ingeniería de *software*. Proporciona un enfoque disciplinado para la asignación de tareas y responsabilidades dentro de una organización de desarrollo. Su objetivo es asegurar la producción de *software* de alta calidad que satisfaga las necesidades de sus usuarios finales, dentro de un horario predecible y presupuesto.

RUP mejora la productividad del equipo, proporcionando a cada miembro del equipo, fácil acceso a una base de conocimientos con las directrices, plantillas y mentores de herramientas para todas las actividades críticas de desarrollo. Al contar todos con la misma base de conocimientos, no importa si trabajan con requerimientos, diseño, pruebas, gestión de proyectos, o la gestión de configuración; se aseguran de que todos los miembros del equipo utilizan actividades de proceso unificado para crear y mantener modelos. En lugar de centrarse en la producción de gran cantidad de documentos en papel, el Proceso Unificado enfatiza el desarrollo y el mantenimiento de los modelos.

Un proyecto RUP tiene cuatro fases: inicio, elaboración, construcción y transición; como en la Figura 03 a continuación. Cada fase tiene una o más iteraciones y se completa con un hito. En ese momento, se evalúa el progreso del proyecto, y las decisiones importantes se hacen sobre la base de ese progreso. El foco de las iteraciones en cada fase es producir entregables técnicos que darán cumplimiento a los objetivos de la fase.

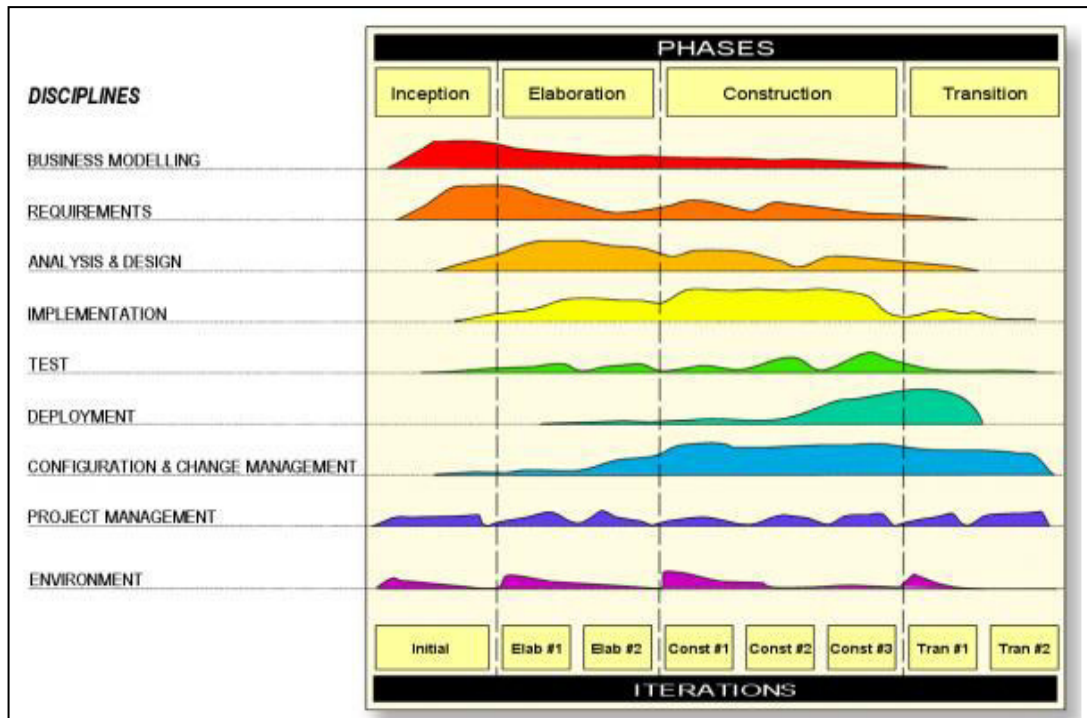


Figura 03. Arquitectura de RUP
Fuente: *International Journal of Software Engineering (IJSE)*

De acuerdo a López (2013), la metodología RUP (*Rational Unified Process*) se divide en 7 etapas o fases el desarrollo de un proyecto de desarrollo de aplicaciones de *software*:

- Modelo del Negocio (*Business Modeling*): hacer claro las reglas del negocio relativas al manejo de la información.
- Requisitos funcionales (*Requirements*): determinar los requisitos de funcionamiento y de operación.
- Análisis y Diseño (*Analysis/Design*): diseñar los programas, módulos, rutinas y demás componentes del sistema, buscando una arquitectura óptima del mismo.
- Implementación (*Implementation*): se instalan los bancos de datos y se montan igualmente las facilidades de comunicación de los programas e interfaces previstas.
- Pruebas (Test): se realizan todas las pruebas tanto a nivel de los módulos independientes como las resultantes de la integración de estos.
- Configuración y Administración de Cambios (*Configuration & Change Mange*): el propósito aquí es llevar a cabo los cambios tanto en la

configuración de equipos, servidores y programas, así como en las diferentes interfaces.

- Instalación (*Deployment*): poner en funcionamiento el producto del proyecto.

Cada una de estas etapas es desarrollada mediante un ciclo de iteraciones, lo cual consiste en reproducir el ciclo de vida en cascada a una menor y cada vez menor escala. Los objetivos de una iteración se establecen en función de la evaluación que se haga de las cantidades y calidades de las iteraciones precedentes o precursoras.

Vale mencionar que el ciclo de vida que se desarrolla por cada iteración, es llevado a cabo bajo la guía combinada de dos disciplinas muy interrelacionadas, a saber:

- Disciplina de desarrollo

Ingeniería de Negocios: consiste en entender las necesidades del negocio.

Análisis de Requerimientos: trasladando las necesidades del negocio a un sistema automatizado de manejo de información.

Análisis y Diseño: trasladando los requerimientos dentro de la arquitectura de *software*.

Implementación: creando *software* que se ajuste a la arquitectura y que tenga el comportamiento deseado.

Pruebas: asegurándose que el comportamiento requerido es el correcto y que todo lo solicitado este presente.

- Disciplina de soporte

Configuración y administración del cambio: guardando todas las versiones del proyecto.

Administrando el proyecto: administrando horarios y recursos.

Ambiente: administrando el ambiente de desarrollo.

Distribución: hacer todo lo necesario para la salida del proyecto.

Es recomendable que a cada una de estas iteraciones se les clasifique y ordene según su prioridad, y que cada una pueda ser convertida luego, en un entregable al cliente. Esto trae como beneficio una conveniente y necesaria retroalimentación que se tendría en cada paso que signifique un entregable o en cada iteración.

En la tesis de (Tabares,L. (2011), RUP identifica seis buenas prácticas, con las que define una forma de trabajar para los equipos de desarrollo de *software*:

- Gestión de requisitos: RUP brinda una guía para encontrar, organizar, documentar, y seguir los cambios de los requisitos funcionales y restricciones. Utiliza una notación de Caso de Uso y escenarios para representar los requisitos.
- Desarrollo de *software* iterativo: desarrollo del producto mediante iteraciones con hitos bien definidos, en la cuales se repiten las actividades, pero con distinto énfasis, según la fase del proyecto.
- Desarrollo basado en componentes: la creación de sistemas intensivos en *software* requiere dividir el sistema en componentes con interfaces bien definidas, que posteriormente serán ensamblados para generar el sistema. Esta característica en un proceso de desarrollo permite que el sistema se vaya creando a medida que se obtienen o se desarrollan sus componentes.
- Modelado visual: utiliza UML, lenguaje para visualizar, especificar, construir y documentar los artefactos de un sistema de *software*. Utilizar éstas herramientas, facilita la gestión de dichos modelos, permitiendo ocultar o exponer detalles cuando sea necesario.
- Verificación continua de la calidad: es importante que la calidad de todos los artefactos se evalúe en varios puntos durante el proceso de desarrollo, especialmente al final de cada iteración.
- Gestión de los cambios: el cambio es un factor de riesgo crítico en los proyectos de *software*. Los artefactos *software* cambian no sólo debido a acciones de mantenimiento posteriores a la entrega del producto, sino también durante el proceso de desarrollo. (p.37)

Según Christou, Ponis, & Palaiologou (2010), el sector bancario es bien conocido por el uso de sistemas grandes, a veces monolíticos, heredados. Ahora, los bancos se encuentran con que tienen que ponerse al día con los rápidos avances en el desarrollo de *software* que requieren nuevos paradigmas de computación orientada a servicios. Por

desgracia, esta tarea no es trivial y requiere a menudo grandes proyectos que son costosos, consumen tiempo y son arriesgados. La opción segura para una metodología de desarrollo es un marco de proceso como el *Rational Unified Process* (RUP), que es lo suficientemente adaptable a cualquier proyecto.

1.2.7 Métrica V3

Para el Ministerio de Administraciones Públicas (2001), la Metodología MÉTRICA V3, ofrece a las Organizaciones un instrumento útil para la sistematización de las actividades que dan soporte al ciclo de vida del *software* dentro del marco que permite alcanzar los siguientes objetivos:

- Proporcionar o definir Sistemas de Información que ayuden a conseguir los fines de la Organización mediante la definición de un marco estratégico para el desarrollo de los mismos.
- Dotar a la Organización de productos *software* que satisfagan las necesidades de los usuarios dando una mayor importancia al análisis de requisitos.
- Mejorar la productividad de los departamentos de Sistemas y Tecnologías de la Información y las Comunicaciones, permitiendo una mayor capacidad de adaptación a los cambios y teniendo en cuenta la reutilización en la medida de lo posible.
- Facilitar la comunicación y entendimiento entre los distintos participantes en la producción de *software* a lo largo del ciclo de vida del proyecto, teniendo en cuenta su papel y responsabilidad, así como las necesidades de todos y cada uno de ellos.
- Facilitar la operación, mantenimiento y uso de los productos *software* obtenido.

La interfaz de gestión de configuración de MÉTRICA Versión 3 permite definir las necesidades de gestión de configuración para cada sistema de información, recogéndolas en un plan de gestión de configuración, en el que se especifican actividades de identificación y registro de productos, que se realizan durante todas las actividades de MÉTRICA Versión 3 asociadas al desarrollo y mantenimiento del sistema de información.

La gestión de configuración facilita además el mantenimiento del sistema, aportando información precisa para valorar el impacto de los cambios solicitados y reduciendo el tiempo de implementación de un cambio, tanto evolutivo como correctivo.

Los procesos principales de Métrica V3 para el desarrollo de sistemas de información son:

- Estudio de Viabilidad del Sistema (EVS)
- Análisis del Sistema de Información (ASI)
- Diseño del Sistema de Información (DSI)
- Construcción del Sistema de información (CSI)
- Implantación y Aceptación del Sistema (ASI)

1.2.8 *Scrum*

Según Nazareno, Leone & Gonnet (2013), *Scrum* se define como un “*framework*” basado en los principios ágiles, utilizado para el desarrollo y gestión de productos complejos, como lo son los productos de *software*. Puede ser visto como un proceso iterativo e incremental que ayuda a involucrar buenas prácticas ingenieriles dentro de una perspectiva iterativa controlada.

Según la publicación de Jaibeer (2013) los componentes de *Scrum* son: los equipos, funciones, eventos, artefactos y reglas:

- **Equipo *Scrum***
 - **Propietario del producto:** gestiona la cartera de producto en forma priorizada para entregar el mejor valor para el negocio, toma las decisiones, y es la persona que realmente conoce el negocio del cliente y su visión del producto. Se encarga de escribir las ideas del cliente, las ordena por prioridad y las coloca en el *product backlog*.
 - **Equipo de desarrollo:** se compone de individuos auto-gestionados (desarrolladores) trabajando como equipo, que hacen el trabajo en incrementos para entregar productos. Suele ser un equipo pequeño de unas 5 - 9 personas y tienen autoridad para organizar y tomar decisiones para conseguir su objetivo.

- **Scrum Master:** dirige el equipo para asegurarse de que se implementen las prácticas y normas de *scrum*, trabaja en estrecha coordinación con el propietario del producto, el equipo de desarrollo, y la organización también.

Sobre la base de proceso *Scrum* híbrido, diferentes equipos también tienen funciones dedicadas para Arquitecto y Project Manager en el equipo para cumplir los requisitos específicos de cada equipo.

- **Eventos Scrum**

- **Sprint:** periodo de casi un mes o menos para entregar el incremento del producto en base a los elementos del *backlog* por la prioridad establecida del equipo.
- **Reunión de planificación de Sprint:** el equipo se compromete en lo que puede ser entregado de elementos del *backlog* priorizados y cómo se logrará.
- **Scrum diario:** reunión diaria en la que cada miembro del equipo detalla lo que ha logrado desde la última reunión, qué se llevará a cabo antes de la próxima reunión, y si existieran inconvenientes.
- **Revisión de sprint:** el equipo muestra el trabajo realizado en el *sprint* y recoge la retroalimentación.
- **Sprint retrospectivo:** oportunidad para que el equipo identifique que salió bien y lo que necesita mejorar.

- **Artefactos de Scrum**

- **Backlog del producto:** lista ordenada de los requisitos que se requieren en el producto
- **Sprint Backlog:** elementos del *backlog* de producto seleccionados y clasificados planificadas para ser entregado en un *Sprint*.
- **Incremento:** elementos del *backlog* del producto entregado en todos los anteriores *sprints*, y están listos como parte del producto terminado.

De acuerdo a la página web ProyectosAgiles.org, *Scrum* realiza entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, *Scrum* está especialmente indicado para proyectos en entornos complejos, donde se

necesita obtener resultados pronto, donde los requisitos son cambiantes o poco definidos, donde la innovación, la competitividad, la flexibilidad y la productividad son fundamentales.

1.2.9 Cuadro Comparativo de Metodologías de *Software*

En el siguiente cuadro se muestra una comparación entre las Metodologías de *Software* revisadas, ver tabla 02.

Tabla 02. Cuadro comparativo de Metodologías de *Software*

	RUP	SCRUM	METRICA V3
Ciclo	4 fases (Inicio, Elaboración, Construcción, Transición)	Cada <i>Sprint</i> (iteración) es un ciclo completo (Evaluación y priorización, Requerimiento detallado, Diseño y análisis, Pruebas, Despliegue)	3 Procesos (Planificación, Desarrollo, Mantenimiento)
Planificación	Se cuenta con hitos intermedios y se tiene se cuenta con una fecha final.	Al final de cada iteración se planifica la siguiente. Usuario final es quien determina el momento en que el proyecto se lleva a cabo.	Gran probabilidad de desviaciones en la planificación
Alcance	Al inicio del proyecto se define el alcance y se documenta, este puede ser revisado durante el proyecto, los cambios estarán sujetos a un procedimiento controlado.	Se genera "Historias de Usuario" para entender y manejar el requerimiento desde el punto de vista del usuario final.	Requisito inicial inestable
Los artefactos	Modelo de casos de uso, documentación (conceptual, funcional, técnica, construcción, pruebas) plan de desarrollo, etc.	<i>Software</i> operativo	Diagrama de clases, de secuencia, de transición de estados; matriz de procesos, diagrama de pantallas, actividades detalladas en la planificación
Tipo de proyecto / producto	Proyectos medianos o grandes, complejos y a largo plazo.	Proyectos que requieren rapidez.	Proyectos muy grandes, no importa su complejidad

Elaboración: los autores

1.3 Definición de términos básicos

- **Control de Versiones**

Según Tello, Sosa & Tello D (2012) es el proceso de almacenar y recuperar cambios de un proyecto de desarrollo. Los sistemas de control de versiones SCV, permiten disponer de una versión anterior para corregir errores o actualizar funciones. Dentro de sus funcionalidades está el conservar las versiones que se hayan generado a través del tiempo, así como los diferentes archivos que integran el proyecto en cuestión, uniendo en forma automática las aportaciones de los integrantes de un equipo de trabajo. Los SCV pueden ser utilizados en muchos entornos, tales como el gestionar las versiones de documentos generados por procesadores de texto, presentaciones multimedia, archivos del sistema, correo electrónico y código fuente, por mencionar algunos.

- **Gestión de Cambios**

Según el Portal de Osiatis - «Gestión de Cambios - Visión General» es la evaluación y planificación del proceso de cambio para asegurar que, si este se lleva a cabo, se haga de la forma más eficiente, siguiendo los procedimientos establecidos y asegurando en todo momento la calidad y continuidad del servicio TI.

- **Gestionar la Configuración**

Definir y mantener las definiciones y relaciones entre los principales recursos y capacidades necesarias para la prestación de los servicios proporcionados por TI, incluyendo la recopilación de información de configuración, el establecimiento de líneas de referencia, la verificación y auditoría de la información de configuración y la actualización del repositorio de configuración.

- **IBM Rational *ClearQuest***

Es un *software* de gestión del ciclo de vida de las aplicaciones que proporciona un seguimiento flexible de los cambios y los defectos, procesos personalizables, generación de informes en tiempo real y la capacidad de rastreo del ciclo de vida para obtener una mejor visibilidad y el control del ciclo de vida de desarrollo del *software*. Esta solución proporciona soporte multi-plataforma escalable a

organizaciones de cualquier tamaño, por lo que puede seguir utilizando los procesos personalizados a medida que las necesidades de la organización evolucionan.

- **ITOP**

De acuerdo a Combodo, (2014) es una aplicación web de código abierto para el día a día de las operaciones de un entorno de TI. ITOP fue diseñado con las mejores prácticas de ITIL, pero no dicta ningún proceso específico. No necesita implementar ningún *software* de cliente en la PC de los usuarios.

- **Normas**

Un documento establecido por consenso y aprobado por un organismo reconocido, que establece normas de uso común y repetido, directrices o características para actividades o sus resultados, encaminadas al logro del grado óptimo de orden en un contexto dado.

- **Ramas**

De acuerdo a Collins, Fitzpatrick, & Pilato, (2004) Línea de desarrollo que existe de forma independiente a otra, pero comparte una historia común si mira suficientemente atrás en el tiempo. Una rama siempre nace como una copia de algo, y a partir de ahí, pasa a generar su propia historia.

- **Repositorio**

Para Collins et al. (2004), el Almacén central de datos. El repositorio guarda información en forma de árbol de archivos—una típica jerarquía de archivos y directorios. Cualquier número de clientes puede conectarse al repositorio y luego leer o escribir en esos archivos. Al escribir datos, un cliente pone a disposición de otros la información; al leer datos, el cliente recibe información de otro.

- **Servicio**

Es un medio para entregar valor a los clientes facilitándoles un resultado deseado sin la necesidad de que estos asuman los costes y riesgos específicos asociados.

- **Subversion**

Según Collins et al., (2004) es un sistema de control de versiones libre y de código fuente abierto. Es decir, maneja ficheros y directorios a través del tiempo, hay un árbol de ficheros en un repositorio central. Es un sistema general que puede ser usado para administrar cualquier conjunto de ficheros y no para la administración de árboles de código fuente, ni el entendimiento nativo de lenguajes de programación, o el suministro de herramientas para la construcción de *software*.

- **Tecnologías de la información (TI)**

Capacidades tecnológicas utilizadas para el almacenamiento, comunicación o procesamiento de información.

La tecnología incluye típicamente ordenadores, telecomunicaciones, aplicaciones y otro *software*. La información puede incluir datos de negocio, voz, imágenes, video, etc.

La Tecnología de la Información (TI) es a menudo usada para soportar los procesos de negocio a través de servicios de TI.

CAPÍTULO II METODOLOGÍA

Para el presente proyecto se utiliza la Investigación Aplicada, que según José Lozada (2014) tiene por objetivo la generación de conocimiento con aplicación directa y a mediano plazo en la sociedad o en el sector productivo. La sección Calidad de Soluciones del Departamento de Informática del Banco de la Nación es la encargada de certificar todos los elementos que realiza la División Desarrollo de Sistemas de Información, siendo su ambiente por donde se despliegan todas las versiones antes de llegar al ambiente de Producción. Este proceso requiere la implementación de una gestión de la configuración, sustentadas en bases sólidas como la norma ISO / IEC 20000 y la experiencia laboral.

Para la revisión de documentos se obtuvo los tres (03) archivos de la norma ISO /IEC 20000, las directivas del ciclo de vida de *software* y el manual de procedimientos de la Sección.

- Parte 1: Requisitos del Sistema de Gestión del Servicio (SGS)
- Parte 2: Código de buenas prácticas
- Parte 3: Directrices para la definición del alcance y aplicabilidad de la Norma ISO/IEC 20000
- BN – DIR- 2400 – 147 – 01 Ciclo de vida del *software* publicado el 24 de enero del 2011
- Procedimiento Certificación de productos *software* de la Sección Calidad de Soluciones publicado en agosto del 2011

La investigación de documentos se basó en la recopilación de datos, para lo cual se procedió a realizar entrevistas al jefe de la Sección de Calidad de Soluciones, Jefe de Operaciones Bancaria, Jefe de Productos Bancarios y Jefe de Canales de Atención de la División del Departamento de Informática del Banco de la Nación, para verificar de qué manera se vienen administrando los elementos de configuración.

2.1 Materiales

2.1.1 Recursos Humanos

En el presente trabajo participaron cinco (05) personas, identificadas en la Tabla 03:

Tabla 03. Requerimiento de Recursos Humanos

Persona	Rol
Yofre Cortez	Líder de Proyecto / <i>Scrum</i> manager
Yofre Cortez / Liliana Díaz	2 Analistas de Procesos
Juan Peña	Desarrollador / Development
Analistas de pruebas	2 Usuarios

Elaboración: los autores

2.1.2 Hardware

En cuanto a las computadoras e impresora se requieren los equipos con las siguientes características de la Tabla 04:

Tabla 04. Requerimiento de hardware

Equipo	Características	Cantidad	Costo Unitario S/
Computadora	Intel (R) Core (TM) i5-3570 CPU @ 3.40GHz , Memoria (RAM) 4.00 GB, Tipo de Sistema 64 bits, Sistema Operativo <i>Windows 7</i> Enterprise	5	2,000.00
Impresora	Kyocera FS-9530DN	1	2,600.00

Elaboración: los autores

2.1.3 Software

En cuanto a los requerimientos de *software* se requiere lo siguiente de la Tabla 05:

Tabla 05. Requerimiento de *software*

N°	Software	Versión	Licencia
1	<i>Microsoft Office Word</i>	2010	<i>Microsoft</i>
2	<i>Microsoft Project</i>	2010	<i>Microsoft</i>
3	<i>Rational ClearQuest</i>	7.0.1.0	<i>IBM Rational</i>
4	<i>SVN by CollabNet</i>	4.0.14	<i>Software libre</i>
5	<i>Eclipse</i>	5.5	<i>Software libre</i>
6	<i>Microsoft Excel</i>	2010	<i>Microsoft</i>
7	<i>Oracle</i>	10G	<i>Oracle</i>
8	<i>Apache Tomcat</i>	6.0	<i>Software libre</i>
9	<i>Xampp Control Panel</i>	1.7.3	<i>Software libre</i>
10	<i>Mysql</i>	5.0	<i>Software libre</i>

Elaboración: los autores

2.1.4 Cronograma del Proyecto

El Planeamiento del Proyecto consta de tres (03) Fases que son las siguientes:

- Planificación de la iteración con una duración de 18 días útiles
- Ejecución de la iteración que tiene una duración de 103 días útiles
- Ejecución y adaptación que tiene una duración de 30 días

2.1.5 Presupuesto

De los requisitos mencionados en los puntos anteriores se excluyen los costos relacionados a *hardware*, ya que son proporcionados por la entidad bancaria.

En cuanto a los requerimientos de *software*, son herramientas libres de costo, ya que se pueden obtener a través de internet y las que tienen licencia del Banco de igual manera también son libres de costo pues son proporcionadas para el desarrollo del trabajo en la Sección.

La Tabla 06 muestra el costo del personal encargado del proyecto:

Tabla 06. Presupuesto

Material	Unidad	Costo unit. S/	Total S/.
Recurso Humano			
Líder del proyecto	300 hrs.	16.00	4,800.00
Analista de proceso 1	200 hrs.	15.00	3,000.00
Analista de proceso 2	100 hrs.	15.00	1,500.00
Desarrollador	500 hrs.	15.00	7,500.00
Analista de pruebas 1	20 hrs.	15.00	300.00
Analista de pruebas 2	20 hrs.	15.00	300.00
Hardware			
Computadora	5 unid.	2,000.00	10,000.00
Impresora	1 unid.	2,600.00	2,600.00
Software			
<i>Microsoft Office Word</i>			0.00
<i>Microsoft Project</i>			0.00
<i>Rational ClearQuest</i>			0.00
<i>SVN by CollabNet</i>			0.00
<i>Eclipse</i>			0.00
<i>Microsoft Excel</i>			0.00
<i>Oracle</i>			0.00
<i>Apache Tomcat</i>			0.00
<i>Xampp Control Panel</i>			0.00
<i>Mysql</i>			0.00
Total			30,000.00

Elaboración: los autores

2.2 Métodos

Para el desarrollo del sistema se ha empleado la metodología *Scrum*, que es un método de ágil gestión y a su vez permite maximizar la productividad de la Sección, ya que sus fases facilitaron su desarrollo y así se obtuvieron los resultados esperados para su elaboración.

Incluye junto con la descripción del ciclo de vida iterativo e incremental para el proyecto, los artefactos o documentos con los que se gestionan las tareas de adquisición y suministro: requisitos, monitorización y seguimiento del avance, así como las responsabilidades y compromisos de los participantes en el proyecto. Para cumplir con el plan de trabajo se elaboró un cronograma de la Implementación del Sistema de Gestión de la Configuración de *Software* (Ver Anexo 10).

La elección de la metodología se realizó en base a un análisis de valores cuantitativos por cada criterio, los cuales son referenciales y pueden variar según cada caso de proyecto y experiencia de las personas y las organizaciones.

Para cada uno de estos criterios se asignaron valores de 1 a 3, siendo Alto (1), Medio (2) y Bajo (3).

El resultado que se obtuvo se muestra en la tabla 07:

Tabla 07. Comparación de metodologías

Criterios	Metodología		
	RUP	Scrum	Métrica v3
Disponibilidad de recursos	3	3	3
Complejidad del proyecto	2	3	2
Entendimientos de requerimientos	2	1	3
Conocimiento del dominio del problema	2	2	2
Manejo de las perspectivas del riesgo	1	2	1
Tiempos de desarrollo	2	3	3
Costos de los proyectos	2	2	2
Calidad de <i>software</i>	1	2	1
Documentación	3	3	2
Resultado	18	21	19

Elaboración: los autores

Para el proyecto se decidió utilizar la metodología *Scrum* por las razones siguientes:

- Obtuvo el mayor puntaje en la evaluación después de realizar una comparación con las metodologías RUP y Métrica v3 como se muestra en la Tabla 07.
- El Sistema de Gestión de Configuración de *Software* planteado, es un sistema modular lo cual permite desarrollar una base funcional mínima y sobre ella ir incrementando las funcionalidades o modificando el comportamiento o apariencia de las ya implementadas.
- *Scrum*, permite entregas frecuentes y continuas al cliente de los módulos terminados, de forma que puede disponer de una funcionalidad básica en un tiempo mínimo y a partir de ahí un incremento y mejora continua del sistema.
- Es un modo de desarrollo adaptable, antes que predictivo. Es decir que se puede tomar decisiones para resolver problemas sobre la marcha y adaptar la manera de trabajo según demande el trabajo a realizar.
- Emplea el modelo de construcción incremental basado en iteraciones y revisiones.
- Entrega de un producto funcional al finalizar cada *Sprint*.
- Posibilidad de ajustar la funcionalidad en base a la necesidad de negocio del cliente.
- Alcance acotado y viable
- Equipos integrados y comprometidos con el proyecto, se auto-administran.

CAPÍTULO III

DESARROLLO DEL PROYECTO

A través de la metodología *Scrum* y la ejecución de las fases Planificación, Ejecución e Inspección y Adaptación, se logró llevar un proceso ágil en el desarrollo del sistema de gestión de la Configuración de *software*, para ello se utilizó el código de buenas prácticas brindado por la ISO 20000 Parte 2. Siendo el cambio, una característica constante en el desarrollo del *software*.

3.1 Planificación de la Iteración

En esta fase del método se realizó una entrevista a los jefes de las secciones del Departamento de Informática para conocer el proceso actual de SCM dentro del ciclo de vida de *software* del Banco de la Nación, lo que llevó a conocer las necesidades. Como resultado del diagnóstico la implementación de una metodología junto con el desarrollo de una plataforma que se integre con las herramientas existentes.

Para la ejecución de esta etapa se realizó la ejecución de los siguientes artefactos, así como la creación de un cronograma de actividades (Ver Anexo 10)

3.1.1 Pila de producto o *Product Backlog*

La pila del producto es la lista de requisitos priorizada que representa la visión del cliente respecto a los objetivos del proyecto.

Como es un documento que varía a lo largo de las iteraciones no es necesario que esté completo para proceder con la primera iteración.

La lista de iteraciones que contiene la pila del producto empleados para el desarrollo del proyecto se muestra en la Tabla 08 *Product Backlog*:

Tabla 08. Product backlog

Selección de requisitos
Registro de usuario
Autenticación de usuario
Traslado y versionado
Log de ejecución
Cruce de programas
Flujo de estado de los mantenimientos
Consulta de los mantenimientos registrados en <i>ClearQuest</i>
Planificación de las Iteraciones
Preparación del ambiente
Elaboración de Guía de instalación de <i>Subversion by CollabNet</i> (servidor y consola)
Instalación de <i>Subversion servidor CollabNet</i>
Configuración de <i>Subversion servidor CollabNet</i>
Desarrollo del enlace entre <i>ClearQuest</i> y Cliente <i>Subversion</i>
Registro y autenticación de usuario
Desarrollo de módulo de traslado – Cliente <i>Subversion</i>
Desarrollo del módulo de versionado <i>host</i> – Cliente <i>Subversion</i>
Desarrollo del módulo de programas en común – Cliente <i>Subversion</i>
Desarrollo del log de ejecución
Elaboración de guía de instalación del servidor de aplicaciones Tomcat
Elaboración de guía de despliegue del Cliente <i>Subversion</i>

Elaboración de guía de usuario del <i>Subversion by CollabNet</i> alineado al CVS del BN
Creación de repositorios
Creación de línea base
Elaboración de guía de usuario del Cliente <i>Subversion</i>
Elaboración de guía para la instalación de <i>Tortoise</i>
Realizar el registro de incidencias
Evaluar observaciones y/o mejoras
Resolución de observación y/o mejoras
Demostración – Exposición del sistema SCM – BN

Elaboración: los autores

3.1.2 *Sprint Backlog*

Lista de tareas a realizar durante la iteración. Son las tareas necesarias para construir un incremento (una parte completa del producto).

Esta lista, descompone el proyecto en unidades de tamaño adecuado para determinar el avance a diario e identificar riesgos y problemas sin necesidad de procesos complejos de gestión. También se aprovechó como herramienta de soporte para la comunicación directa del equipo, cubriendo todas las tareas identificadas por el equipo para conseguir el *sprint*.

La lista de *sprint backlog* del proyecto se muestra en la Tabla 09:

Tabla 09. *Sprint backlog*

<i>Sprint 1 Preparación, instalación y configuración</i>
Solicitar servidor con los recursos necesarios para soportar la instalación del servidor <i>SubversionEdge by CollabNet</i>
Solicitar los accesos para el servidor con los privilegios necesarios a la base de datos del <i>ClearQuest</i> así como a las librerías <i>host</i>
Elaboración de guía de instalación de <i>Subversion by CollabNet</i> (servidor y consola)
Instalación del servidor <i>SubversionEdge by CollabNet</i> , el cual instalará básicamente dos componentes, un servidor con toda la lógica del

control de versiones y un servidor web para publicar su consola de administración
Instalación del Servidor Web para el Cliente Web que contendrá todos los módulos que se vayan desarrollando
Configuración del servidor <i>SubversionEdge by CollabNet</i>
<i>Sprint 2 Desarrollo del enlace entre ClearQuest y Cliente Subversion</i>
Búsqueda del mantenimiento según el código registrado en el <i>ClearQuest</i>
Filtro por fecha de registro, prefijo de proyecto, prioridad, título del mantenimiento y estado
<i>Sprint 3 Registro y autenticación de usuario</i>
Creación del módulo para la creación de nuevos usuarios
Creación de pantalla de bienvenida y lógica de autenticación
<i>Sprint 4 Módulo de traslado – Cliente Subversion</i>
Creación de plantillas JCL para el traslado según el tipo de programa
Desarrollo del programa para el procedimiento de traslado entre Desarrollo y Certificación
<i>Sprint 5 Módulo de versionado host – Cliente Subversion</i>
Creación de plantillas JCL para convertir un programa seleccionado a texto plano
Creación de plantillas JCL para copiar un archivo en una carpeta temporal
Desarrollo del programa que administra los eventos siguientes: <ul style="list-style-type: none"> - Armar el JCL para convertir a texto plano - Armar el JCL para la copia del texto plano por FTP - Mover el archivo al repositorio que le corresponde - Ejecución de comandos SVN para el versionado
<i>Sprint 6 Módulo de programas en común – Cliente subversión</i>
Filtro de mantenimientos cuyo estado se encuentra en Certificación. (Anexo : Estados de mantenimientos)
Lista los programas según mantenimiento
Cruce de programas en común

<i>Sprint 7 Log de ejecución</i>
Desarrollo del Log de Ejecución que permita verificar las fechas y hora de los accesos, confirmación del éxito del proceso, entre otros
<i>Sprint 8 Flujo de estados de los mantenimientos</i>
Desarrollo de flujo de los estados de los mantenimientos entre ambientes
<i>Sprint 9 Módulo de consulta al ClearQuest</i>
Consulta de las actividades, historial, documentos y programas de los mantenimientos que se encuentran registrados en el <i>ClearQuest</i>
<i>Sprint 10 Elaboración de guías</i>
Elaboración de guía de instalación del servidor de aplicaciones Tomcat
Elaboración de guía de despliegue de Cliente <i>Subversion</i>
Elaboración de guía de usuario del <i>Subversion by CollabNet</i> alineado al CVS del BN: <ul style="list-style-type: none"> - Creación de usuarios - Creación de repositorios - Reglas de acceso a los repositorios - Log de cambios
Elaboración de guía de usuario de Cliente <i>Subversion</i>
Elaboración de guía para la instalación del <i>Tortoise</i>
<i>Sprint 11 Creación de repositorios</i>
Desarrollo y ejecución del JAR para crear un subdirectorio por cada proyecto registrado en el <i>ClearQuest</i>
<i>Sprint 12 Creación de línea base</i>
Creación de línea base
<i>Sprint 13 Pruebas del producto</i>
Pruebas de funcionalidad
Pruebas de adaptación
Pruebas de satisfacción
<i>Sprint 14 Demostración – Exposición</i>
Presentación del Sistema

Elaboración: los autores

3.1.3 *Sprint*

El *Sprint* es cada una de las iteraciones del ciclo de vida iterativo del proyecto.

Para el desarrollo del sistema de Gestión de Configuración de *Software* se trabajó en base a 14 *Sprints*, los cuales se muestran en la Tabla 10:

Tabla 10. Lista de *Sprint*

<i>Sprint 1</i> Preparación, instalación y configuración
<i>Sprint 2</i> Desarrollo del enlace entre <i>ClearQuest</i> y Cliente <i>Subversion</i>
<i>Sprint 3</i> Registro y autenticación de usuario
<i>Sprint 4</i> Módulo de Traslado – Cliente <i>Subversion</i>
<i>Sprint 5</i> Módulo de versionado <i>host</i> – Cliente <i>Subversion</i>
<i>Sprint 6</i> Módulo de programas en común – Cliente <i>subversión</i>
<i>Sprint 7</i> Log de Ejecución
<i>Sprint 8</i> Flujo de estados de los mantenimientos
<i>Sprint 9</i> Módulo de consulta al <i>ClearQuest</i>
<i>Sprint 10</i> Elaboración de guías
<i>Sprint 11</i> Creación de repositorios
<i>Sprint 12</i> Creación de línea base
<i>Sprint 13</i> Pruebas del producto
<i>Sprint 14</i> Demostración – Exposición

Elaboración: los autores

Se implementará un sistema / cliente web el cual integrará el *SubversionEdge by CollabNet* con las herramientas existentes y contará con cinco (5) módulos:

- Módulo de traslado y versionado
- Log de ejecución
- Módulo de cruce de programas
- Módulo de flujo de estado de los mantenimientos
- Módulo de consultas *ClearQuest*

Adicionalmente, con la instalación del servidor *SubversionEdge by CollabNet*, se instalará también una Consola de Administración con los módulos siguientes:

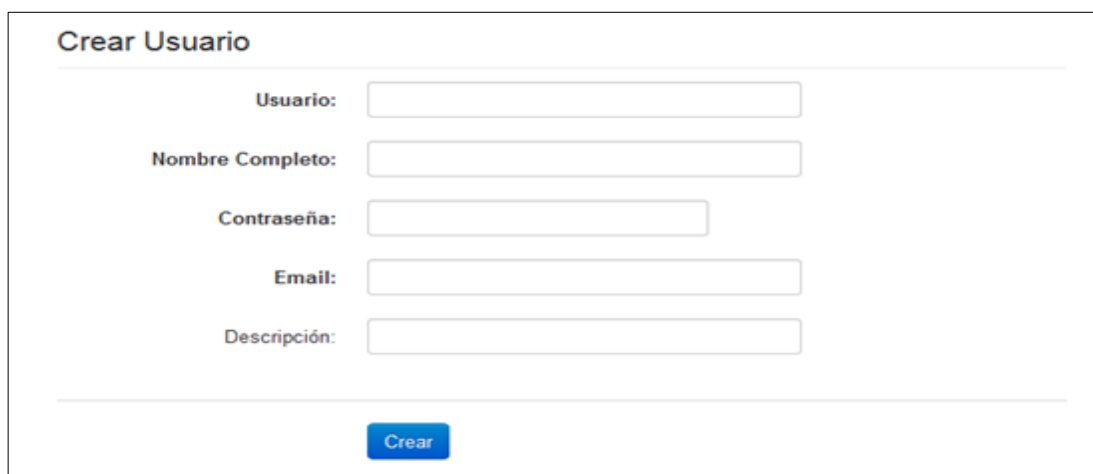
- Explorador de librerías
- Comparador de versiones

3.2 Ejecución de la iteración

Para cada una de las funcionalidades del sistema “Cliente *Subversion*” se crearon historias de usuarios (Ver Anexo 3), los que son utilizados para evaluar que las funcionalidades se están cumpliendo de acuerdo a lo solicitado.

3.2.1 Registro de usuario

Este módulo nos permite la creación de un nuevo usuario, como se muestra en la figura 04.



Crear Usuario

Usuario:

Nombre Completo:

Contraseña:

Email:

Descripción:

Figura 04. Registro de usuario
Elaboración: los autores

3.2.2 Autenticación de usuario

En la figura 05 se muestra la Pantalla de inicio del sistema Cliente *Subversion*.

Bienvenidos

CLIENTE SUBVERSION

Usuario:

Clave :

Figura 05. Autenticación de usuario
Elaboración: los autores

3.2.3 Traslado y versionado

Módulo para el traslado de Elementos de Configuración del ambiente de Desarrollo al ambiente de Certificación, realizando adicionalmente el versionado.

En la Figura 06 se muestra la búsqueda de un mantenimiento

Buscar

Mantenimiento: Fecha:

Registro: [Limpiar](#)

Proyecto: Prioridad:

Titulo: Estado:

Fecha: [Limpiar](#)

[Versionamientos](#) [Programas en comun](#)

Mantenimiento	Nombre	Titulo	Estado	Prioridad	Fecha	Ultimo	Dependencia	Versiona
2015-M0429	BDUC	LIMPIEZA DE DATOS - CLIENTE BDUc	Asignado pruebas	3-Media	2015-05-21 09:25:12	2015-05-27 14:45:41	CERTIFICACION	Versiona

« « Página 1 de 1 » » 10 ▼ Mostrando 1 - 1 de 1

Figura 06. Búsqueda de mantenimiento
Elaboración: los autores

En la Figura 07 se muestra el listado por tipo de programas del mantenimiento que se ha consultado.

2015-M0429		
TIPO	PROGRAMAS	ACCION
CICS	0	No hay programas
BATCH	1	Versionar
JCL	1	Versionar
MAPAS	0	No hay programas
TOLD APLICACION	0	No hay programas
TOLD PRODUCTO	0	No hay programas

Figura 07. Listado por tipo de programas
Elaboración: los autores

En la Figura 08 se muestra los programas que se encuentran en el tipo de programa al que se le ha indicado la acción Versionar.

2015-M0429 - BATCH					
TODOS	SELECCIONADOS	Nota:	COPIA CERMERGE		
#	PROGRAMA	ACCION	DETALLE	ESTADO	SISTEMA
<input checked="" type="checkbox"/>	SIUC5772	NUEVO	Limpieza de Magnitud del cliente	Registrado	SIUC-Sistema de Información Unica de Clientes

Figura 08. Traslado y versionado de programas
Elaboración: los autores

3.2.4 Log de Ejecución

En la Figura 09 se muestra el log de ejecución de los mantenimientos, el cual permite ver estado de cada actividad realizada en el proceso de traslado y versionado.

Se versiono el mantenimiento correctamente							
<u>Log versionamiento - 2015-M0428</u>							
Bandeja	Programa	Fecha Proceso	Resultado	Hora Inicio	Hora Fin	Log	Tipo
198	SIUC5774	20150605	OK	06:55:51 PM	06:55:57 PM	JOB09145 JOB09145 JOB DEL CERMERGE TERMINO CORRECTAMENTE	BATCH
198	Procedimiento de Copia	20150605	OK	06:55:57 PM	06:55:57 PM	Se genero el jcl para ejecutarse	BATCH
198	Procedimiento de Copia	20150605	OK	06:55:57 PM	06:55:57 PM	Se genero el jcl para ejecutarse	BATCH
198	Procedimiento de Copia	20150605	OK	06:55:57 PM	06:55:57 PM	SE AGREGO CORRECTAMENTE EL HEADER Y FOOTER DEL JCL COPIA	BATCH
198	Procedimiento de Copia	20150605	OK	06:55:57 PM	06:55:57 PM	SE AGREGO CORRECTAMENTE EL HEADER Y FOOTER DEL JCL FTP	BATCH
198	Procedimiento de Copia	20150605	OK	06:55:57 PM	06:56:02 PM	JOB09146 EL JOB DE COPIA SE EJECUTO CORRECTAMENTE	BATCH
198	Procedimiento de Copia	20150605	OK	06:56:02 PM	06:56:08 PM	JOB09147 EL JOB FTP SE EJECUTO CORRECTAMENTE	BATCH
198	SIUC5774	20150605	OK	06:56:08 PM	06:56:08 PM	1 file(s) moved.	BATCH
198	SIUC5774	20150605	OK	06:56:08 PM	06:56:08 PM	Se fuerza el commit y se verifica	BATCH

Figura 09. Log de ejecución
Elaboración: los autores

3.2.5 Cruce de programas

Permite verificar si uno o más Elementos de Configuración de un mantenimiento están incluidos en otros mantenimientos registrados en el *ClearQuest*, como se muestra en la figura 10.

Existen los siguientes mantenimientos con programas que se cruzan:

Programa	Mantenimiento	Estado	Fecha Modificacion	Mantenimiento	Estado	Fecha Modificacion
Consulta de cajero específico por a	2015-M0457	Asignado_pruebas	2015-06-05 11:20:19	2015-M0418	Asignado_pruebas	2015-06-05 11:20:11
Cosulta de cajero específico	2015-M0457	Asignado_pruebas	2015-06-05 11:20:19	2015-M0418	Asignado_pruebas	2015-06-05 11:20:11
binarios mdsdriver	2015-M0189	Asignado_pruebas	2015-05-06 10:29:34	2014-M0865	Asignado_pruebas	2015-05-06 10:56:32
binarios ndcdriver	2015-M0189	Asignado_pruebas	2015-05-06 10:29:34	2014-M0865	Asignado_pruebas	2015-05-06 10:56:32

Informacion actualizada al: 2015-06-05 06:57:05 PM
 Correos que seran notificados
 Juan Pe a jmpu0186@gmail.com
 Yofre Cortez yofrecortez@hotmail.com

[Notificar consulta](#)

Figura 10. Cruce de programas
Elaboración: los autores

3.2.6 Flujo de estado de los mantenimientos

En la figura 11 se puede ver de manera gráfica los estados por los que atravesó un mantenimiento registrado en el *ClearQuest*.

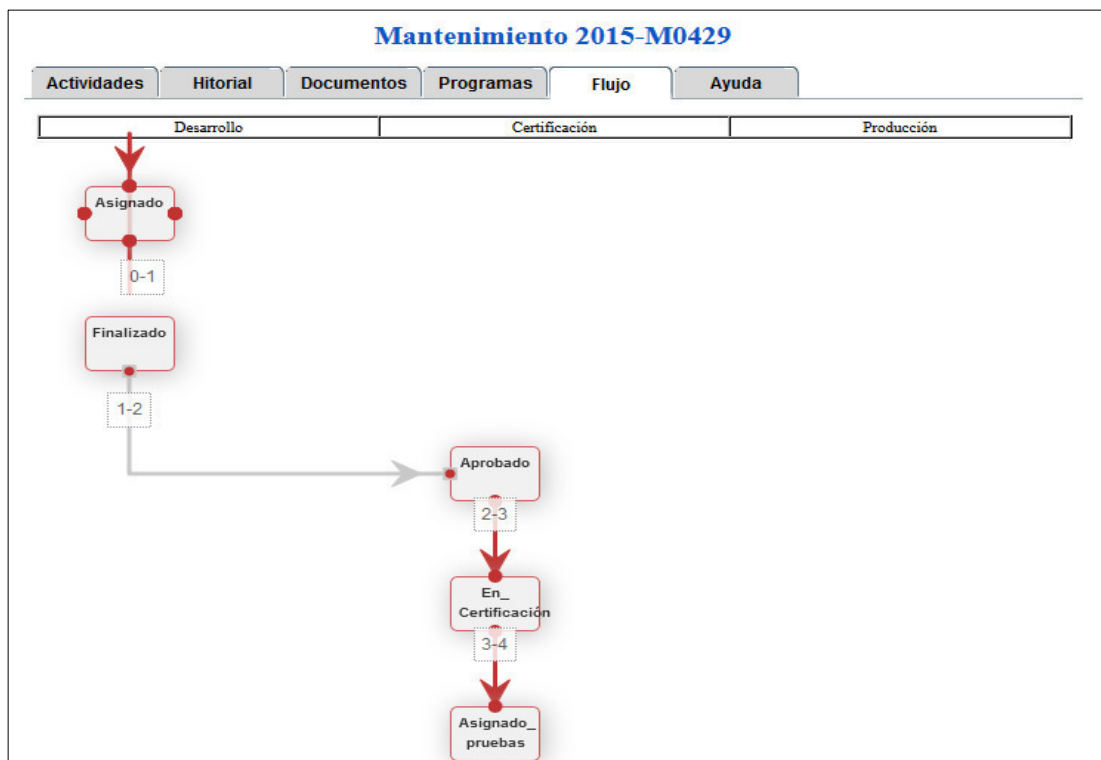


Figura 11. Flujo de estado de los mantenimientos
Elaboración: los autores

3.2.7 Consulta de los mantenimientos registrados en *ClearQuest*

Permite acceder a diferentes componentes incluidos en un mantenimiento registrado en el *ClearQuest*: Actividades, Historial, Documentos y Programas.

En la figura 12 se muestran las actividades de un mantenimiento.

Actividades	Hitorial	Documentos	Programas	Flujo	Ayuda
Actividad ID	Título	Estado	Area	Responsable	
2015-M0429-A001	LIMPIEZA DE DATOS - CLIENTE BDUC	Asignada	Desarrollo de Sistemas	ecardenas	
2015-M0429-A002	LIMPIEZA DE DATOS - CLIENTE BDUC	Finalizada	Desarrollo de Sistemas	lcampbell	
2015-M0429-A003	SC 2015-05-19 Limpieza BDUC	Finalizada	Calidad de Soluciones	kpoma	
2015-M0429-A004	SC 2015-05-19 Limpieza BDUC	Iniciada	Calidad de Soluciones	ycortez	

Figura 12. Actividades del mantenimiento
Elaboración: los autores

En la figura 13 se muestra el historial de las acciones realizadas por los usuarios para el mantenimiento seleccionado.

Mantenimiento 2015-M0429					
Actividades	Hitorial	Documentos	Programas	Flujo	Ayuda
Fecha y Hora	Accion	Estado - Antes	Estado - Despues	Usuario	
2015-05-21 09:25:49	Submit	no_value	Registrado	rarmas	
2015-05-21 09:30:59	Asignar	Registrado	Asignado	rarmas	
2015-05-21 18:33:38	Modificar	Asignado	Asignado	lcampbell	
2015-05-21 18:34:18	Modificar	Asignado	Asignado	lcampbell	
2015-05-21 18:34:58	Modificar	Asignado	Asignado	lcampbell	
2015-05-21 18:43:08	Modificar	Asignado	Asignado	lcampbell	
2015-05-22 11:45:47	Modificar	Asignado	Asignado	lcampbell	
2015-05-22 11:53:09	Modificar	Asignado	Asignado	lcampbell	
2015-05-22 11:55:36	Modificar	Asignado	Asignado	lcampbell	
2015-05-22 11:58:01	Modificar	Asignado	Asignado	lcampbell	
2015-05-22 12:02:32	Modificar	Asignado	Asignado	lcampbell	
2015-05-26 10:23:52	Finalizar	Asignado	Finalizado	rarmas	
2015-05-26 10:23:55	Aprobar	Finalizado	Aprobado	rarmas	
2015-05-27 14:33:10	Pasar_a_Certificación	Aprobado	En_Certificación	mdavila	
2015-05-27 14:45:41	Asignar_Pruebas	En_Certificación	Asignado_pruebas	mdavila	

Figura 13. Historial del mantenimiento
Elaboración: los autores

En la figura 14 se muestran los documentos del mantenimiento seleccionado.

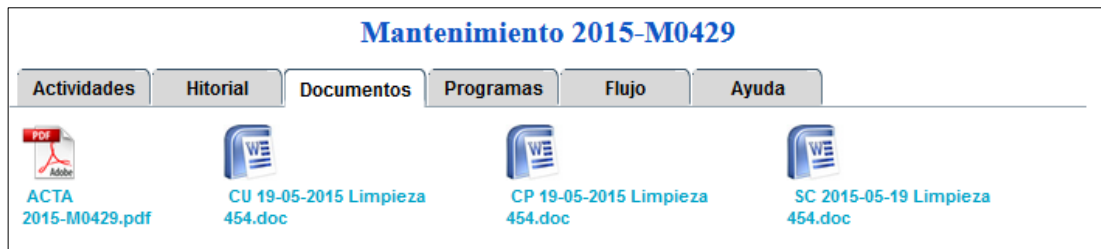


Figura 14. Documentos del mantenimiento
Elaboración: los autores

En la figura 15 se muestra la información de los programas de un mantenimiento.

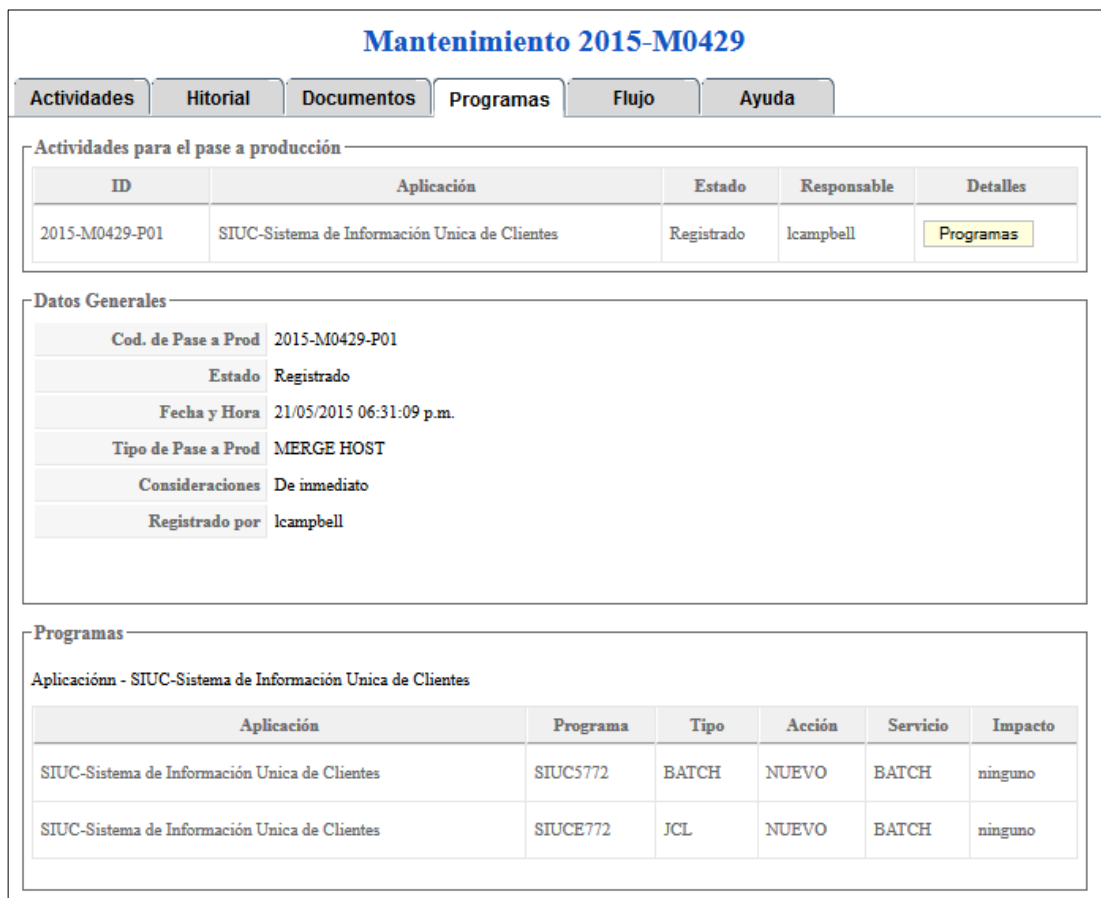


Figura 15. Programas del mantenimiento
Elaboración: los autores

3.3 Artefactos

3.3.1 Incremento

El incremento es la parte del producto desarrollado en un *Sprint* y tiene como características: documentación completamente terminada y operativa, en condiciones de ser entregada al cliente final.

La finalización de cada incremento es comunicada al equipo a través del correo electrónico.

Se crearon manuales de usuario para las funcionalidades principales.

3.3.2 Gráfica de producto

La gráfica del producto o *Burn up* es una representación gráfica del plan de producto previsto por el gestor de producto. En la figura 16 se muestra los temas del sistema en el orden que se desean, y el tiempo en el que se prevé su ejecución para realizar el seguimiento y cumplimiento de los avances.

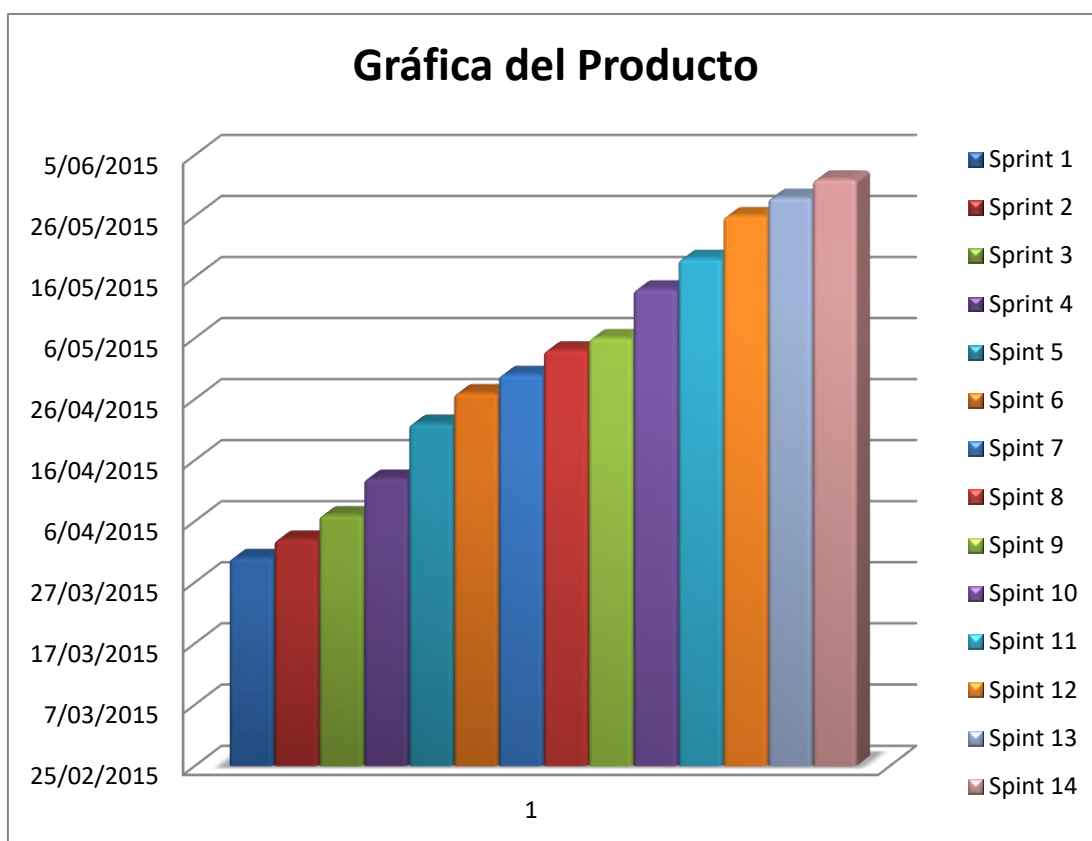


Figura 16. Gráfica del producto
Elaboración: los autores

3.3.3 Gráfica de avance

Se realiza la gráfica de avance o *Burn Down* la cual muestra el estado de avance del trabajo del *sprint* en curso.

En la figura 17 se muestra el tiempo planificado versus el tiempo real empleado para la ejecución de cada *sprint*.

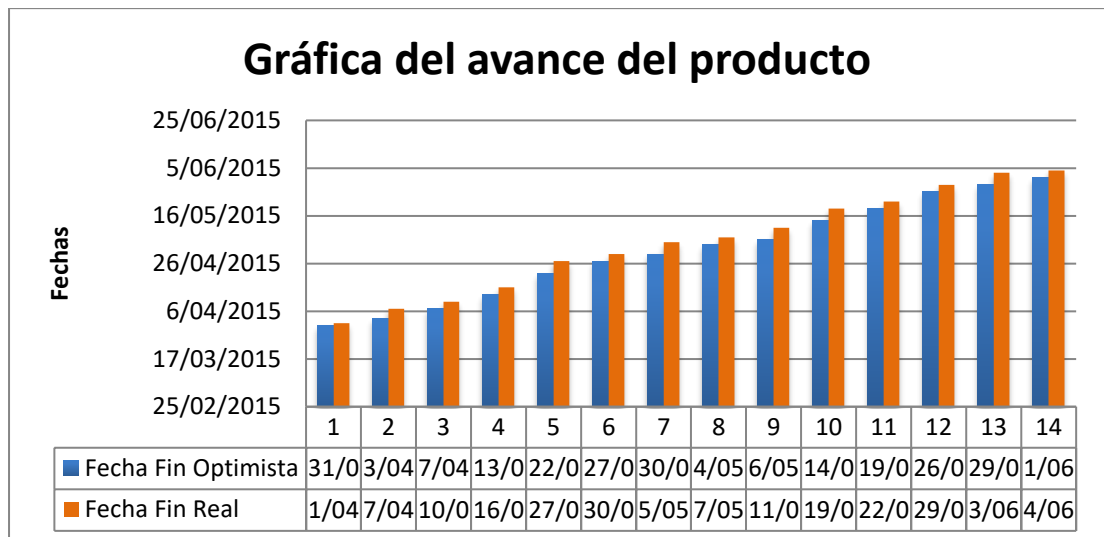


Figura 17. Gráfica de avance del producto
Elaboración: los autores

3.3.4 Reunión de inicio de *sprint*

La reunión de inicio de *Sprint* se utiliza para determinar las funcionalidades o historias de usuario que se van a incluir en el próximo incremento.

3.3.5 Reunión técnica diaria

Es la reunión que se realiza todos los días con el equipo con presencia del *Scrum Manager* con una duración aproximada de 10 minutos. Todos los roles empleados para el desarrollo del proyecto están descritos en la tabla 11: Requerimiento de Recursos Humanos.

Tabla 11. Requerimiento de Recursos Humanos

Persona	Rol
Yofre Cortez	Líder de proyecto / <i>Scrum manager</i>
Yofre Cortez / Liliana Díaz	2 Analistas de procesos
Juan Peña	Desarrollador / Development
Analistas de pruebas	2 Usuarios

Elaboración: los autores

3.3.6 Reunión de cierre de *sprint* y entrega del incremento

Se realiza cada vez que se va a entregar un incremento que está terminado para ser probado.

Como se mencionó líneas arriba, el incremento se envía a través del correo electrónico realizando las indicaciones necesarias para la prueba.

3.3.7 Arquitectura de la aplicación

Para tener una visión más clara sobre la arquitectura de la aplicación. Podemos observar los siguientes puntos en la figura 18:

- Tenemos dos tipos de Programas que podemos versionar *Host* (programas Cobol) y *Open* (JAVA, .NET, Scripts)
- Para Versionar programas *Host* usaremos el Sistema Cliente “Cliente *Subversion*”
- Para programas *Open* usaremos el Sistema Cliente *TortoiseSVN*
- Sea cual sea el sistema cliente que usemos estos versionarán en el Servidor *Subversion*
- El servidor *Subversion* será el encargado de administrar las revisiones en el repositorio.

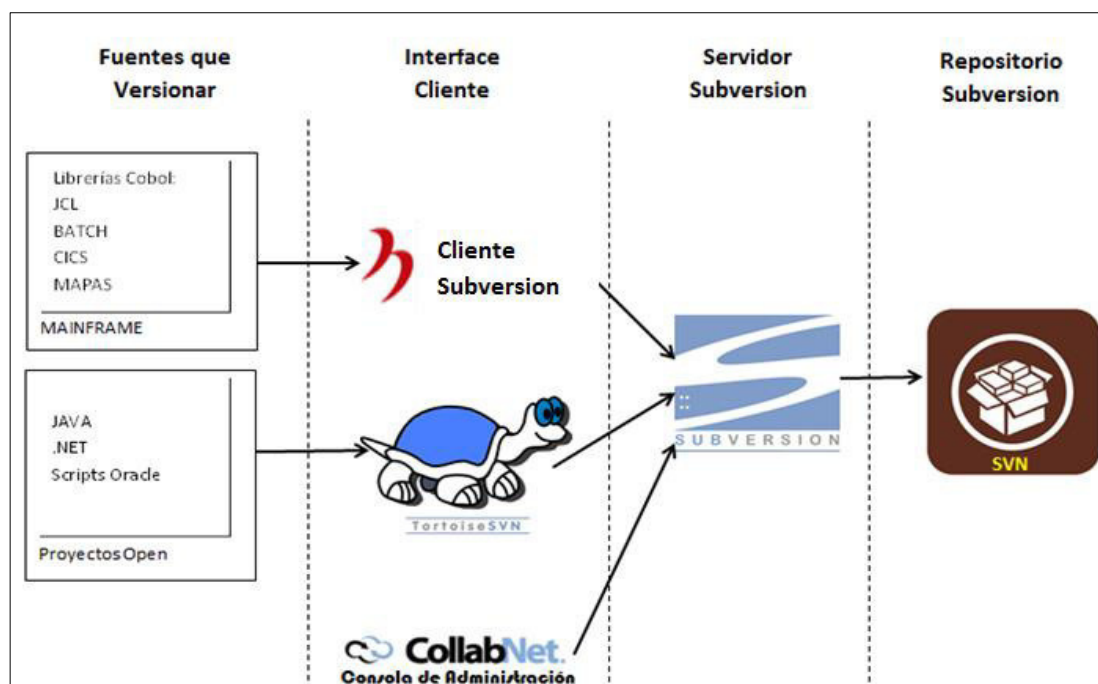


Figura 18. Gráfica de arquitectura de la aplicación
Elaboración: los autores

3.4 Inspección y adaptación

En este apartado se planea exponer el resultado de las pruebas preliminares y la capacitación del sistema dirigido a algunos analistas de la Sección Calidad de Soluciones.

Como resultado de esta capacitación se firmará un acta de conformidad por parte del jefe de la Sección Calidad de Soluciones aceptando la entrega del sistema. (Ver Anexo 9)

3.5 Factores Críticos de éxito

Dentro de los factores críticos de éxito identificados se tiene:

- Apoyo y participación de las jefaturas de las Secciones
- Resistencia al cambio por parte del personal
- Grado de conocimiento técnico y experiencia del personal
- Limitaciones presupuestarias

3.6 Lecciones aprendidas

- Existe una resistencia inicial a la mejora por parte del personal afectado, la cual se debe minimizar a través de políticas que ayuden a gestionar el cambio e incentiven el compromiso con la mejora.
- Es imprescindible que se designe un equipo de trabajo de la organización como responsable del monitoreo de la implementación del proyecto.
- Las jefaturas de las secciones involucradas deben dar muestras explícitas de su compromiso asistiendo a las reuniones más importantes e informar al personal involucrado.

CAPÍTULO IV

PRUEBAS Y RESULTADOS

Con la implementación del sistema de Gestión de la Configuración de *Software* se consiguió satisfacer las necesidades del departamento de Informática del Banco de la Nación; así como llevar un control sobre el impacto de pueda ocasionar el cambio de un elemento de configuración.

4.1 Funcionalidad

Para evaluar la funcionalidad del sistema se contrastó las historias de usuario con la funcionalidad de los entregables.

El usuario usó el sistema y podrá contrastar su historia de usuario con la funcionalidad que brinda Cliente Web y la Consola de Administración; y se podrá evaluar el porcentaje de la funcionalidad que brinda el sistema con respecto a las necesidades indicadas.

Es necesario indicar que para las pruebas se contó con el apoyo de los analistas de la sección Calidad de Soluciones, quienes procedieron a validar las historias de usuario según los criterios descritos a continuación.

En la Tabla 12 se muestra la cantidad de historias de usuario probadas con sus respectivas funcionalidades que se realizaron para dar la conformidad del sistema de gestión de servicio.

Tabla 12. Historias de usuario

Historias de usuario	% de Funcionalidad
Registro de usuario	100 %
Autenticación de usuario	100 %
Módulo de traslado y versionado – Cliente <i>Subversion</i>	100 %
Módulo de programas en común – Cliente <i>Subversión</i>	100 %
Log de ejecución	100 %
Módulo flujo de estados de los mantenimientos	100 %
Módulo de consulta al <i>ClearQuest</i>	100 %

Elaboración: los autores

4.2 Adaptación

Para la evaluación de la aceptación del sistema el usuario evaluó las historias de usuarios en cuatro (4) rangos de calificación diferentes.

La Tabla 13 muestra la lista de historias de usuario utilizadas para evaluar la adaptación del sistema con sus respectivas evaluaciones.

Tabla 13. Aceptación del sistema

Historias de usuario	Ineficiente	Regular	Buena	Excelente
Módulo de traslado – Cliente <i>Subversion</i>			X	
Módulo de versionado <i>host</i> – Cliente <i>Subversion</i>			X	
Módulo de programas en común – Cliente <i>Subversión</i>			X	
Log de ejecución		X		
Módulo flujo de estados de los mantenimientos			X	
Módulo de consulta al <i>ClearQuest</i>		X		

Elaboración: los autores

La Tabla 14 muestra el resultado obtenido de la evaluación a las personas que forman parte de la Sección Calidad de Soluciones a través de las calificaciones que otorgaron a las funcionalidades del sistema.

Tabla 14. Tabla de resultados de evaluación del sistema

Historias de usuario	Ineficiente	Regular	Buena	Excelente
Registro de usuario			10	
Autenticación de usuario		2	8	
Traslado y versionado		1	9	
Log de ejecución		4	6	
Cruce de programas		2	8	
Flujo de estado de los mantenimientos		2	8	
Consulta de los mantenimientos registrados en <i>ClearQuest</i>		1	9	

Elaboración: los autores

La figura 19 representa en forma gráfica el resultado de la evaluación de las funcionalidades implementadas en el sistema de Gestión de Configuración de *Software*.

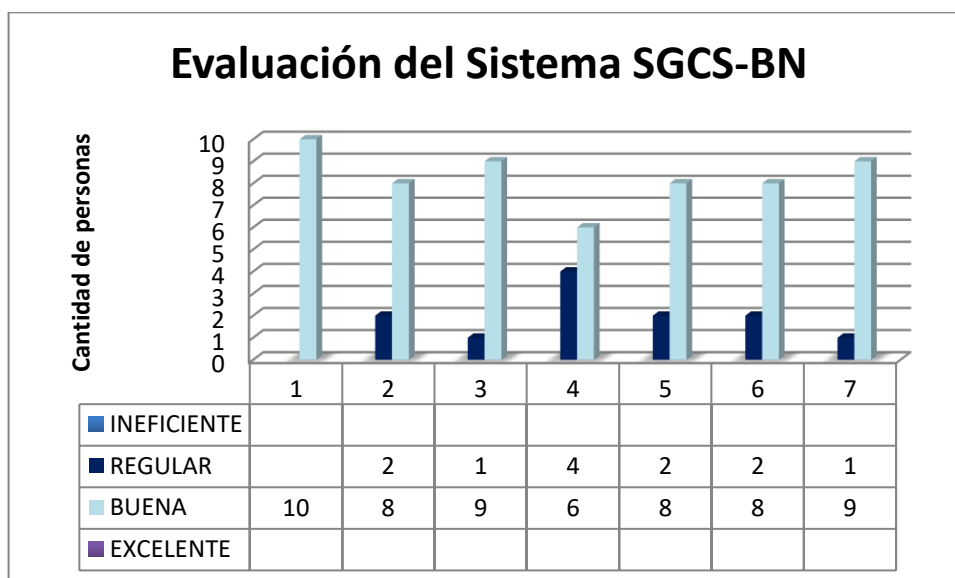


Figura 19. Evaluación de funcionalidades
Elaboración: los autores

4.3 Satisfacción

Para evaluar la satisfacción del usuario con respecto al sistema, el usuario deberá usar el Cliente *Subversion* y la Consola de Administración y contestar las preguntas con respecto a su satisfacción y se podrá concluir el nivel de satisfacción del usuario final.

La Tabla 15 muestra las preguntas utilizadas para medir la satisfacción del sistema:

Tabla 15. Satisfacción del sistema

Preguntas	Ineficiente	Regular	Buena	Excelente
¿El sistema se presenta de forma amigable?			X	
¿El sistema es fácil de usar?		X		
¿La información presentada por el sistema es lo que realmente espera ver?			X	
¿La navegación por las opciones del sistema es sencilla?		X		
¿La ejecución de cada proceso (registros, consultas, etc.) es complicada?		X		
¿Son razonables los tiempos de respuesta del sistema?			X	
¿La información presentada por el sistema es clara?			X	
¿La información presentada por el sistema es de calidad?			X	
¿El funcionamiento del sistema es entendible?			X	
¿La capacitación previa fue útil para usar el sistema?			X	

Elaboración: los autores

La Tabla 16 nos ofrece el resultado de la satisfacción del usuario con respecto al Cliente *Subversion*:

Tabla 16. Tabla de satisfacción del usuario

Preguntas	Ineficiente	Regular	Buena	Excelente
¿El sistema de presenta de forma amigable?		2	8	
¿El sistema es fácil de usar?		4	5	1
¿La información presentada por el sistema es lo que realmente espera ver?		1	5	4
¿La navegación por las opciones del sistema es sencilla?		3	7	
¿La ejecución de cada proceso (registros, consultas, etc.) es complicada?		2	8	
¿Son razonables los tiempos de respuesta del sistema?		1	9	
¿La información presentada por el sistema es clara?		1	9	
¿La información presentada por el sistema es de calidad?			9	1
¿El funcionamiento del sistema es entendible?		1	9	
¿La capacitación previa fue útil para usar el sistema?			10	

Elaboración: los autores

Como se puede observar el grado de satisfacción del usuario es bueno, lo que indica que la mayor cantidad de usuarios se encuentran satisfechos con la información y la utilidad del sistema. La Figura 20 demuestra en forma gráfica la satisfacción del usuario:



Figura 20. Satisfacción del usuario
Elaboración: los autores

CAPÍTULO V

DISCUSIONES Y APLICACIONES

En este capítulo se consiguió tener acceso a información referente al total de mantenimientos atendidos en lo que va del 2015 (enero - mayo), evidenciando que con la implementación de la gestión de la configuración de *software* a través del Cliente *Subversion* para la Sección Calidad de Soluciones del Banco de la Nación, se consigue automatizar el versionado de todos los programas *host*, reduciendo los tiempos de traslado y despliegue de los elementos de configuración del ambiente de Desarrollo a Certificación.

5.1 Discusiones

Se evaluó la operatividad del Cliente *Subversion* cuyos resultados se obtuvieron de analizar el procedimiento de traslado de programas *host* antes y después del uso del Cliente *Subversion* dando como resultado una mejora en tiempos de traslado, así como mejora en las herramientas a disposición de la sección Calidad de Soluciones para la devolución de mantenimientos.

5.1.1 Despliegue y traslado de elementos de configuración de *software*

Se realizó la evaluación de la cantidad de mantenimientos atendidos para los meses de enero en adelante del periodo 2015, información que muestra la alta demanda de certificación de programas *host*.

La Tabla 17 muestra el total de mantenimientos atendidos por la Sección Calidad de Soluciones por mes en lo que va del año 2015

(enero - mayo), detallando además la cantidad de programas *OPEN* y el total de programas *HOST*.

Tabla 17. Mantenimientos atendidos por mes

Mes	Total de mantenimientos	Total de programas <i>open</i>	Total de programas <i>host</i>
Enero	70	12	277
Febrero	78	7	115
Marzo	97	19	177
Abril	60	6	126
Mayo	68	31	180

Fuente: ITOP

La figura 21 presenta en forma gráfica la comparativa entre la cantidad de programas *Open* y de programas *Host* incluidos en los mantenimientos atendidos mes a mes.

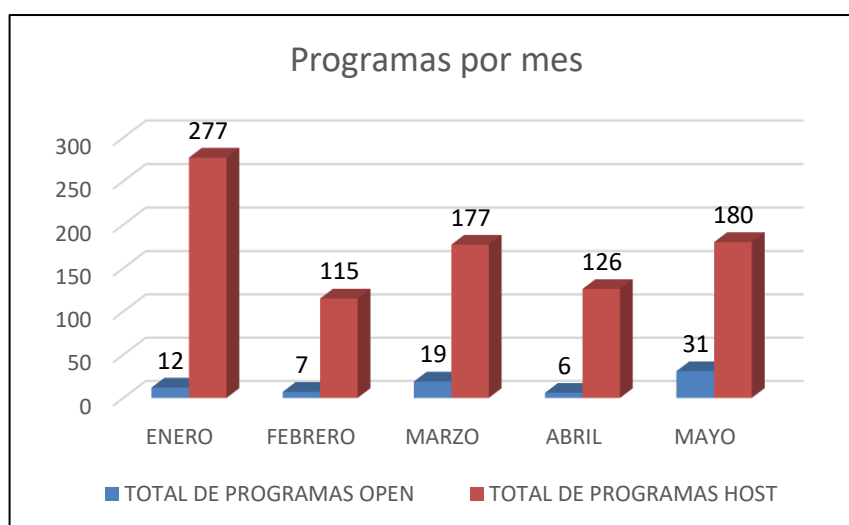


Figura 21. Programas por mes
Elaboración: los autores

5.1.2 Evaluación de los tiempos de despliegue

Se evaluaron los tiempos que los analistas planificadores invierten actualmente en sus actividades de traslado y se comparó los resultados obtenidos a través del Cliente Subversión.

La Tabla 18 muestra el tiempo invertido para el traslado de programas *Host* del ambiente de Desarrollo a Certificación en lo que va del año 2015 (enero – mayo) comparándolo con el tiempo en caso se hubiera utilizado el Cliente *Subversion*.

Tabla 18. Tiempos de traslado de programas *host* por mes

Mes	Total de mantenimientos	Total de programas <i>host</i>	Tiempos de traslado (min)	
			Método tradicional	Cliente <i>Subversion</i>
Enero	70	277	1385	350
Febrero	78	115	575	390
Marzo	97	177	885	485
Abril	60	126	630	300
Mayo	68	180	900	340

Elaboración: los autores

La figura 22 presenta en forma gráfica el tiempo de traslado de programas *Host* con el método tradicional comparándolo con el Cliente *Subversion*.

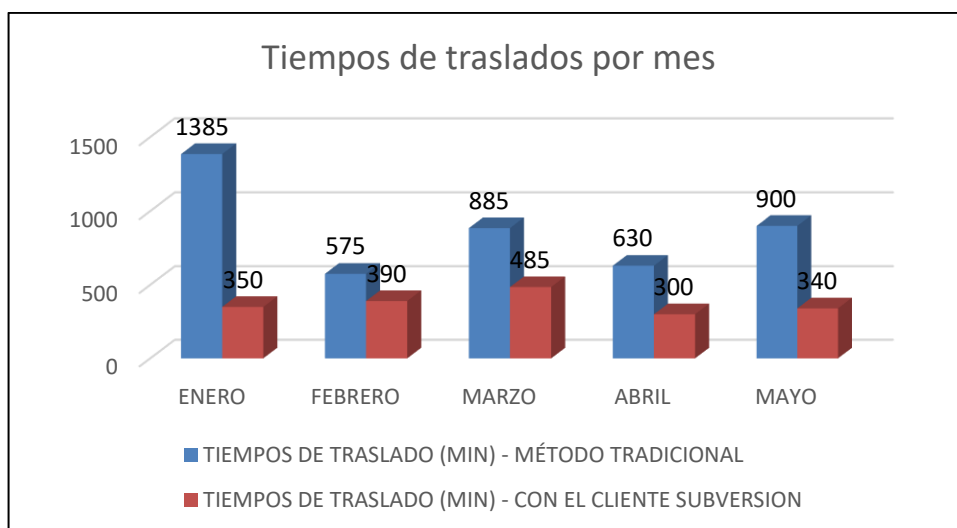


Figura 22. Tiempos de traslados por mes

Elaboración: los autores

La Tabla 19 muestra la reducción de los tiempos de traslado de los programas *Host* del ambiente de Desarrollo a Certificación en lo que va

del año 2015 (enero – mayo) comparándolo con el tiempo en caso se hubiera utilizado el Cliente *Subversion*.

Tabla 19. Reducción en tiempos (%) de traslado de programas *host*

Mes	Horas - Método Tradicional	Horas - Cliente <i>Subversion</i>	% Reducción por mes
Enero	23	6	25%
Febrero	10	7	68%
Marzo	15	8	55%
Abril	11	5	48%
Mayo	15	6	38%
		Promedio	47%

Elaboración: los autores

5.1.3 Identificación de cruces de programas entre mantenimientos

En todo el mes de junio se puso a prueba el cliente *Subversión* y al usar la opción “cruce de programas” se detectó que, de todos los mantenimientos atendidos en tres ocasiones, los desarrolladores aprobaron mantenimientos en donde uno mismo programa o eran incluidos en más de un mantenimiento lo cual ponía en riesgo la operatividad de las pruebas y la puesta en producción. Al detectarse se pudo proceder a la inmediata devolución de uno mantenimientos involucrados en el cruce luego de evaluarlo con desarrollo según su prioridad.

5.1.4 Evaluación de la implementación del sistema de gestión de la configuración de *software*

Para la evaluación sobre la implementación del sistema de gestión de la configuración de *software* se definieron cuatro frecuencias de uso para evaluar si el personal de la Sección de Calidad de Soluciones utiliza, según como se puede evidenciar en la Tabla 20.

Tabla 20. Uso del Sistema de gestión de la configuración del *software*

Historias de usuario	Frecuencia de uso			
	Nunca	Rara vez	Casi siempre	Siempre
Módulo de traslado y versionado – Cliente <i>Subversion</i>				X
Módulo de programas en común – Cliente <i>Subversión</i>				X
Log de ejecución			X	
Módulo flujo de estados de los mantenimientos			X	
Módulo de consulta al <i>ClearQuest</i>		X		

Fuente: Elaboración de los autores

De lo expuesto se demuestra que se implantó el sistema de gestión de la configuración de *software* en la sección Calidad de Soluciones.

5.2 Aplicaciones

- Sería ideal que la gestión de configuración de *software* se aplique a los tres ambientes (Desarrollo, Certificación y Producción). Gestión que deberá ser administrado por la División de Infraestructura.
- El concepto de ramas de *Subversion* provee una potente herramienta para el ambiente de desarrollo, al promover la edición colaborativa y la compartición de datos.

CONCLUSIONES

1. Se logró reducir en 47% el tiempo invertido en las actividades de traslado de programas *host*.
2. A través de la opción “cruce de programas” se redujo el riesgo de posibles colisiones de programas entre mantenimientos, antes de que ingresen al ambiente de Certificación.
3. La herramienta proporciona evidencia objetiva y concreta de la creación y evolución del producto.
4. Alinear las funcionalidades de un sistema dentro de un marco auditable ayuda mantener un control a nivel funcional y de seguridad.
5. Se logró implementar una herramienta totalmente integrada a las existentes en el Banco de la Nación.
6. La gestión de la configuración de *software* produjo una “pseudo burocratización” que va desapareciendo conforme se optimizan los procedimientos y se adquiere la cultura del control.
7. A través de diversos procedimientos y guías junto a la herramienta se logró una correcta administración de los elementos de configuración de *software*.

RECOMENDACIONES

1. Mantener actualizado constantemente las Políticas, Plan de Gestión de la Configuración, Plan de Pruebas, Casos de Pruebas y todos los procesos relacionados a la SCM según se vayan presentando nuevos requerimientos o cambios, de no registrar algún cambio importante podría causar deficiencia en los procesos que se encuentran trabajando actualmente con normalidad.
2. Motivar al personal de trabajo sobre la importancia de realizar correctamente los procesos de SCM e indicarles los beneficios que se obtienen para ellos, la empresa y los clientes.
3. Aplicar una auditoría de la configuración del *software* para que ayude a complementar la finalidad de los informes de estado y terminar por completo el proceso de la SCM, llevando un control amplio de todas sus fases.

FUENTES DE INFORMACIÓN

Bibliográficas

AENOR. (2007). Tecnología de la información. Gestión del Servicio. Parte 2: Código de buenas prácticas. *AENOR.*

AENOR. (2011). Tecnología de la información. Gestión del Servicio. Parte 1: Requisitos del Sistema de Gestión de Servicio. *AENOR.*

Business Innovatios. (2014). Probador Certificado ISTQB Nivel Básico - Conocimiento Esencial para Profesionales de Prueba.

De Pablos, C., Lopez, J. J., Romo, S. M., & Medina, S. (2012). Organización y transformación de los sistemas de información en la empresa. ESIC Editorial.

Jaibeer, M. (2013). *Agile Project Management with GreenHopper 6 Blueprints.* Packt Publishing.

López, F. J. T. (2013). Administración de proyectos de informática. Ecoe Ediciones.

Merchan Paredes, L., & Gomez Mosquera, D. A. (2011). Gestión de Configuración. Validación de un modelo liviano para pequeñas empresas de desarrollo de software. Entramado.

NTP. (2006). Tecnología de la Información. Procesos del ciclo de vida del software.

Pytel, P. (2011). Método de estimación de esfuerzo para proyectos de explotación de información. Herramienta para su validación. (Doctoral *dissertation*. Tesis de Magister en Ingeniería del *Software*). Convenio Universidad Politécnica de Madrid e Instituto Tecnológico Buenos Aires.

Spillner, A., Linz, T., & Schaefer, H. (2014). *Software Testing Foundations: A Study Guide for the Certified Tester Exam*. Rocky Nook, Inc.

Tabares, L. F. (2011). Personalización de RUP para proyectos académicos de desarrollo de *software*.

TECNOBIT. (2012). Sistemas de gestión de la configuración SW “El reto de la automatización.”

Hemerográficas

Anwar, A. (2014). *A Review of RUP (Rational Unified Process)*.

Christou, I., Ponis, S., & Palaiologou, E. (2010). *Using the Agile Unified Process in Banking*. *IEEE Software*, 27(3), 72–79. <http://doi.org/10.1109/MS.2009.156>

Daniele, M., Uva, M., Martelloto, P., & Picco, G. (2010). Aplicación de herramientas CASE a la enseñanza de Ingeniería de *Software*: Gestión de la Configuración de *Software* y *Testing* Funcional. In V Congreso de Tecnología en Educación y Educación en Tecnología.

Espinosa, M. M. (2012). Buenas prácticas sobre gestión de configuración en proyectos con metodología MDA, 3(2).

Fernandez, S., & Osso, M. (2010). Análisis de la gestión de configuración de *software* aplicada al modelo de espiral.

Lozada, J. (2014). Investigación Aplicada: Definición, Propiedad Intelectual e Industrial. CIENCIAMÉRICA, 3(1), 47-50.

Nazareno, R., Leone, H., & Gonnet, S. M. (2013). Trazabilidad de procesos ágiles: un modelo para la trazabilidad de procesos Scrum. In XVIII Congreso Argentino de Ciencias de la Computación.

Direcciones electrónicas:

Collins, B., Fitzpatrick, B. W., & Pilato, M. (2004). Control de versiones con Subversion. Retrieved from <http://svnbook.red-bean.com/en/1.0/svn-book.pdf>

Combodo. (2014). iTop *Documentation Wiki*. Retrieved June 15, 2015, from <https://wiki.openitop.org/doku.php>

Curso ITIL® *Foundation* > ITIL® *Foundation*. (n.d.). Retrieved April 28, 2015, from <http://itilv3.osiatis.es/itil.php>

Gestión de Cambios - Visión General. (n.d.). Retrieved May 2, 2015, from http://itil.osiatis.es/Curso_ITIL/Gestion_Servicios_TI/gestion_de_cambios/vision_general_gestion_de_cambios/vision_general_gestion_de_cambios.php

Ministerio de Administraciones Públicas. (2001). *Metrica V3 - Introducción*. Retrieved from http://administracionelectronica.gob.es/pae_Home/dms/pae_Home/documentos/Documentacion/Methodologias-y-guias/Metricav3/METRICA_V3_Introduccion.pdf

ProyectosAgiles.org. (n.d.). Qué es SCRUM | proyectos Ágiles. Retrieved September 6, 2014, from <http://www.proyectosagiles.org/que-es-Scrum>

ANEXOS

	Página
Anexo 1: Funciones del Banco de la Nación	62
Anexo 2: Entrevistas	64
Anexo 3: Historias de Usuario	66
Anexo 4: Accesos de <i>CollabNet</i>	70
Anexo 5: Ciclo de vida del <i>software</i>	72
Anexo 6: Guía para instalar <i>CollabNet Subversion Edge</i>	73
Anexo 7: Guía para instalar TortoiseSVN	78
Anexo 8: Lista de Aplicativos – Programas <i>Host</i>	82
Anexo 9: Acta de Conformidad	88
Anexo 10: Cronograma del Proyecto	89
Anexo 11: Memorándum de versionamiento de aplicaciones <i>open</i>	93

Anexo 1: Funciones del Banco de la Nación

El Banco de la Nación es una empresa de derecho público, integrante del Sector Economía y Finanzas, que opera con autonomía económica, financiera y administrativa.

Sus funciones son:

- Brindar servicios de pagaduría de acuerdo a las instrucciones que dicte la Dirección General del Tesoro Público.
- Brindar servicios de recaudación por encargo de los acreedores tributarios.
- Actuar como Agente Financiero del Estado
- Otorgar facilidades financieras al Gobierno Central y a los Gobiernos Regionales y Locales, en los casos en que estos no sean atendidos por el Sistema Financiero Nacional.
- Brindar servicios de Cuentas Corrientes a las entidades del Sector Público Nacional y a Proveedores del Estado.

Sección Calidad de Soluciones - Funciones

De acuerdo al «Manual de Organización y Funciones Departamento de Informática», (2014) la Sección Calidad de Soluciones se encarga de asegurar que las soluciones en Tecnologías de Información nuevos o los cambios en aquellos existentes, pasen a producción con la calidad e integridad necesaria para el correcto funcionamiento de las operaciones del Banco.

Funciones:

- a) Elaborar y ejecutar el Plan Estratégico del Banco; así como elaborar y ejecutar los proyectos del Plan Operativo Institucional – POI y los proyectos internos asignados a la Sección.
- b) Coordinar y participar en la planificación de las pruebas dentro de los planes de trabajo de cada equipo de proyecto.
- c) Ejecutar, documentar y supervisar la ejecución de las pruebas de certificación registrando las observaciones y comentarios técnicos y/o de los usuarios, clasificándolos para proponer los ajustes necesarios.

- d) Certificar los productos y soluciones informáticas desarrollos por el Banco o adquiridos a terceros.
- e) Proponer e implementar normas, procedimientos, estándares de calidad en Tecnologías de Información, para satisfacer los requisitos de los clientes y otras partes interesadas.
- f) Atender e implementar las medidas correctivas recomendadas por el Órgano de Control Institucional y los organismos de control externo.
- g) Garantizar un aseguramiento de la calidad de las soluciones de tecnologías de información, productos, *software*, hardware y procesos.

Anexo 2: Entrevistas

Entrevista: Rosario Armas	Respuesta
¿Cuentan algún control de versiones para sus elementos de <i>software</i> ?	Sí
De ser Sí, ¿cuál es y cómo funciona?	El Historial de versiones propio de <i>Libraryn</i> en Desarrollo y Producción
¿Se tiene determinado los responsables de este control? ¿Quiénes son?	Equipo de Infraestructura
¿En las auditorías realizadas se toca el tema de control de versiones?	Sí
¿Cuál es la frecuencia de las auditorías?	Anuales
¿Hay alguna observación por parte de auditoría con respecto al control de versiones?	No, debido a que no nos corresponde, sin embargo si se presentan observaciones que nosotros mismos (desarrollo), realizamos sobre el historial de versiones de las librerías.
¿Cuáles son?	Estas observaciones las enviamos a infraestructura. Las cuales normalmente son por pérdida de versiones.
¿Se han levantado estas observaciones?	La solución que nos da infraestructura es restaurar la última versión correcta que encuentran, sin embargo esto nos ocasiona problemas.
¿Cree que sería conveniente / útil implementar un sistema de configuración de <i>software</i> ?	Sí
¿Qué esperarías de este sistema de configuración de <i>software</i> ?	Log de modificaciones, secuencia de cambios, riesgo del cambio.

Entrevista: Martín Figueroa	Respuesta
¿Cuentan algún control de versiones para sus elementos de <i>software</i> ?	Sí
De ser Sí, ¿cuál es y cómo funciona?	Se lleva un control manual para evitar que pase una nueva versión de un programa que aún se encuentre en pruebas.
¿Se tiene determinado los responsables de este control? ¿Quiénes son?	Sí, el analista planificador.
¿En las auditorías realizadas se toca el tema de control de versiones?	Sí, pero no le corresponde a nuestra Sección.
¿Cuál es la frecuencia de las auditorías?	Anuales
¿Hay alguna observación por parte de auditoría con respecto al control de versiones?	No, debido a que no nos corresponde.
¿Cuáles son?	
¿Se han levantado estas observaciones?	
¿Cree que sería conveniente / útil implementar un sistema de configuración de <i>software</i> ?	Sí
¿Qué esperarías de este sistema de configuración de <i>software</i> ?	Log de Auditoría, Comparación entre versiones

Anexo 3: Historias de Usuario

Tabla Anexo 01. Historia de usuario Módulo de traslado y versionado

HISTORIA DE USUARIO	
Número: 1	
Nombre Historia:	Registro de usuario
Prioridad:	Muy alta
Desarrollador:	Juan Peña
Estimación:	8 hrs.
Descripción: El rol "Gestor de la Configuración" ingresará al módulo para realizar para crear un nuevo usuario.	
Prueba: El Gestor de la Configuración deberá ingresar al Cliente <i>Subversion</i> y elegir la opción crear un "nuevo usuario", además de elegir el perfil del mismo.	

Elaboración: los autores

Tabla Anexo 02. Historia de usuario Módulo de traslado y versionado

HISTORIA DE USUARIO	
Número: 2	
Nombre Historia:	Autenticación de usuario
Prioridad:	Muy alta
Desarrollador:	Juan Peña
Estimación:	8 hrs.
Descripción: El usuario ingresará a la pantalla de bienvenida del sistema, donde se le solicitará su usuario y contraseña.	
Prueba: El sistema solicitará usuario y contraseña, las cuales serán validadas. De no ser correctas mostrará el mensaje "Usuario y/o contraseña incorrecto(s)".	

Elaboración: los autores

Tabla Anexo 03. Historia de usuario Módulo de traslado y versionado

HISTORIA DE USUARIO	
Número: 3	
Nombre Historia:	Módulo de traslado y versionado
Prioridad:	Muy alta
Desarrollador:	Juan Peña
Estimación:	48 hrs.
Descripción: El rol "Analista Planificador" ingresará al módulo para realizar el traslado y versionado de los EC's registrados en los mantenimientos del <i>ClearQuest</i> .	
Prueba: El Analista Planificador deberá ingresar al Cliente <i>Subversion</i> y realizar la búsqueda de un mantenimiento que haya sido registrado en el <i>ClearQuest</i> , una vez encontrado procederá a versionar los elementos incluidos en el mismo.	

Elaboración: los autores

Tabla Anexo 04. Historia de usuario Módulo de auditoría

HISTORIA DE USUARIO	
Número: 4	
Nombre Historia:	Log de ejecución
Prioridad:	Alta
Desarrollador:	Juan Peña
Estimación:	16 hrs.
Descripción: El módulo deberá permitir la visualización de la ejecución de cada actividad realizada en el módulo de traslado y versionado.	
Prueba: Una vez terminada las actividades del módulo de traslado y versionado del Cliente <i>Subversion</i> , el analista planificador ingresará al módulo de auditoría para comprobar el éxito de cada uno.	

Elaboración: los autores

Tabla Anexo 05. Historia de usuario Módulo de programas en común

HISTORIA DE USUARIO	
Número: 5	
Nombre Historia:	Módulo de programas en común
Prioridad:	Medio
Desarrollador:	Juan Peña
Estimación:	8 hrs
Descripción: El analista planificador ingresará al módulo de programas en común para comprobar si existe uno o más EC's registrados en más de un mantenimiento del <i>ClearQuest</i> , cuyo estado sea de Certificación.	
Prueba: El analista planificador elegirá la opción programas en común para comprobar si existe algún cruce de programas entre mantenimientos.	

Elaboración: los autores

Tabla Anexo 06. Historia de usuario Módulo de flujo de estados de los mantenimientos

HISTORIA DE USUARIO	
Número: 6	
Nombre Historia:	Módulo flujo de estados de los mantenimientos
Prioridad:	Baja
Desarrollador:	Juan Peña
Estimación:	16 hrs
Descripción: El analista planificador podrá revisar mediante un diagrama de flujo los estados por lo que ha atravesado un mantenimiento.	
Prueba: El analista planificador ingresará a un mantenimiento del listado de la búsqueda para elegir la opción "flujo" y podrá visualizar un diagrama de flujo con los estados por cada ambiente (Desarrollo, Certificación y Producción)	

Elaboración: los autores

Tabla Anexo 07. Historia de usuario Módulo de consulta al *ClearQuest*

HISTORIA DE USUARIO	
Número: 7	
Nombre Historia:	Módulo de consulta al <i>ClearQuest</i>
Prioridad:	Media
Desarrollador:	Juan Peña
Estimación:	8 hrs
Descripción: El analista planificador podrá consultar las actividades, historial, documentos y programas de los mantenimientos registrados en el <i>ClearQuest</i> .	
Prueba: El analista planificador ingresará a un mantenimiento del listado de la búsqueda para elegir la opción “actividades”, “historial”, “documentos” o “programas” y podrá visualizar la información registrada en el <i>ClearQuest</i> .	

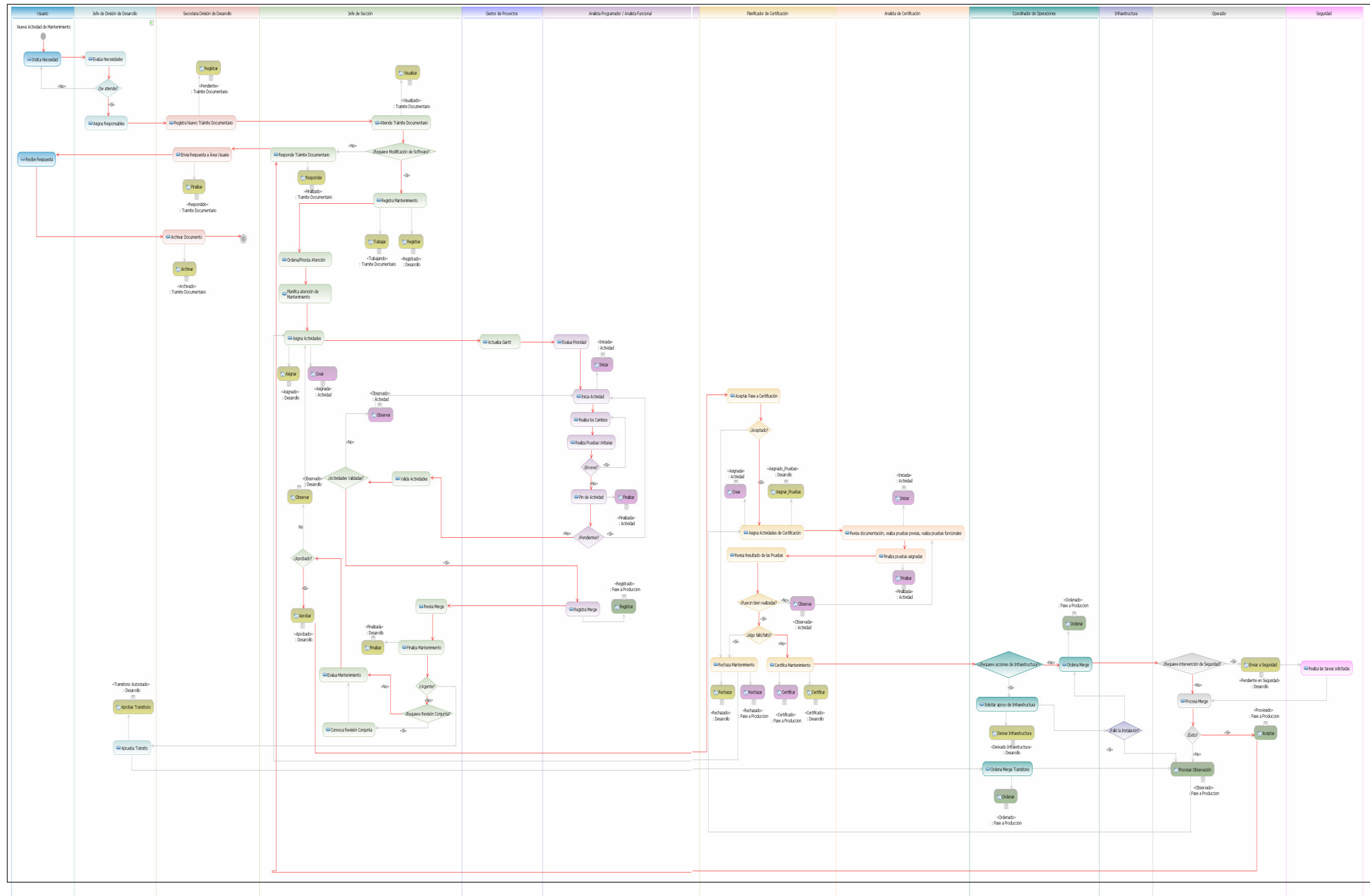
Elaboración: los autores

Anexo 4: Accesos de *CollabNet*

Role	Permissions
ROLE_ADMIN	Un administrador raíz tiene todos los permisos para el sitio. Al instalar <i>CollabNet Subversion</i> , el sitio tiene solamente este usuario.
ROLE_ADMIN_REPO	Administradores del repositorio tienen permisos para: <ul style="list-style-type: none"> * Creación de un repositorio * Conecte un repositorio existente * Editar las reglas de autorización de <i>Subversion</i> para dar a los usuarios acceso a todo un repositorio o cualquier ruta dentro del repositorio
ROLE_ADMIN_SYSTEM	Los administradores del sistema tienen permisos para administrar el servidor <i>Apache Subversion</i> desde la consola pueden <ul style="list-style-type: none"> * Opciones de configuración de actualización del servidor * Iniciar y detener el servidor * Registros de Monitor
ROLE_ADMIN_USER	Un administrador de usuario, por ejemplo, un líder de equipo, tiene permisos para: <ul style="list-style-type: none"> * Añadir (y eliminar) los usuarios * Activar cuentas de usuario * Actualización de los perfiles de usuario y las contraseñas para todas las cuentas de usuario
ROLE_ADMIN_HOOKS	Administradores de <i>script</i> gancho pueden crear, editar, renombrar, copiar, descargar o eliminar archivos de <i>script</i> gancho repositorio.
ROLE_USER	Esta es la función básica requerida para un usuario activo en el sitio de <i>CollabNet Subversion</i> . Sin embargo, incluso sin esta

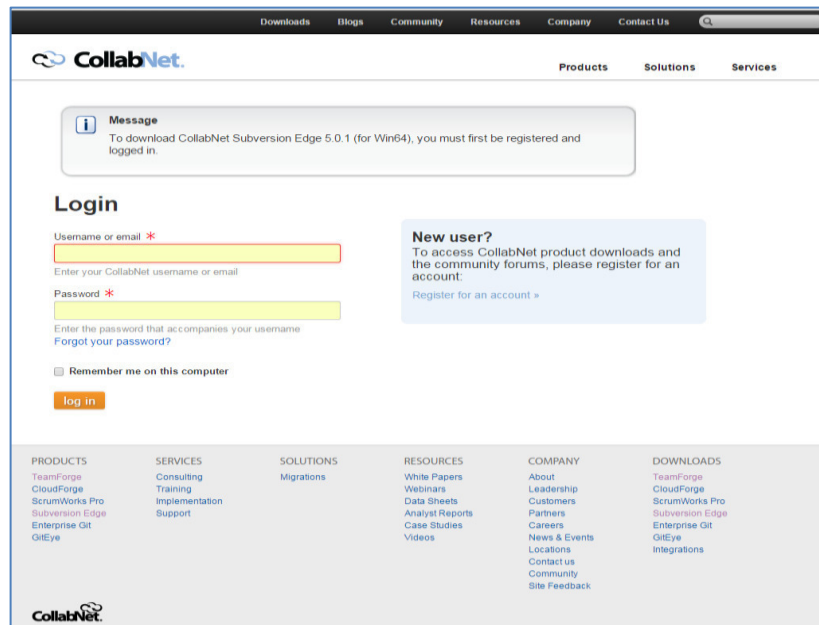
	<p>función, los usuarios pueden realizar la comprobación de repositorio y comprometerse operaciones. Los usuarios con este rol pueden editar su propia contraseña y el perfil de la información.</p>
--	--

Anexo 5: Ciclo de vida del software

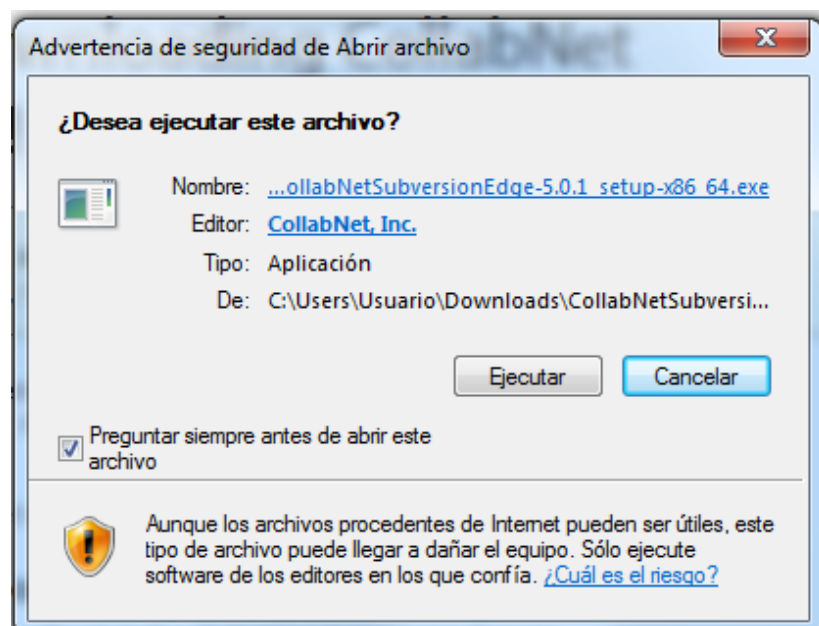


Anexo 6: Guía para instalar *CollabNet Subversion Edge*

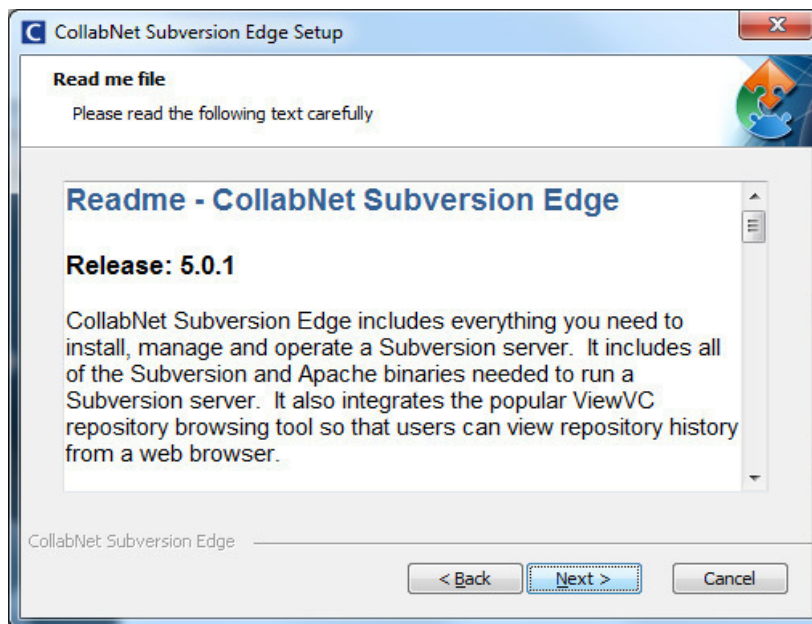
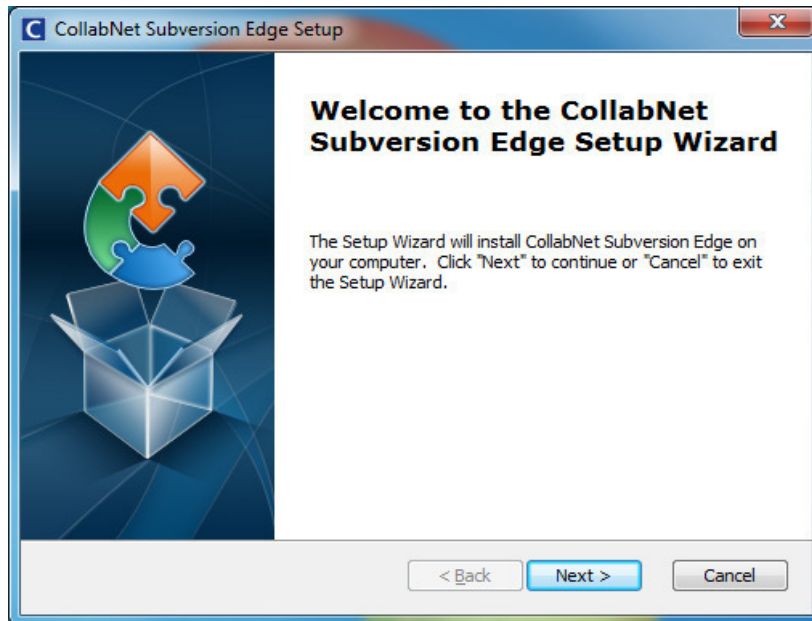
1. Ingresar a <http://www.collab.net/svnedge>.
2. Registrarse

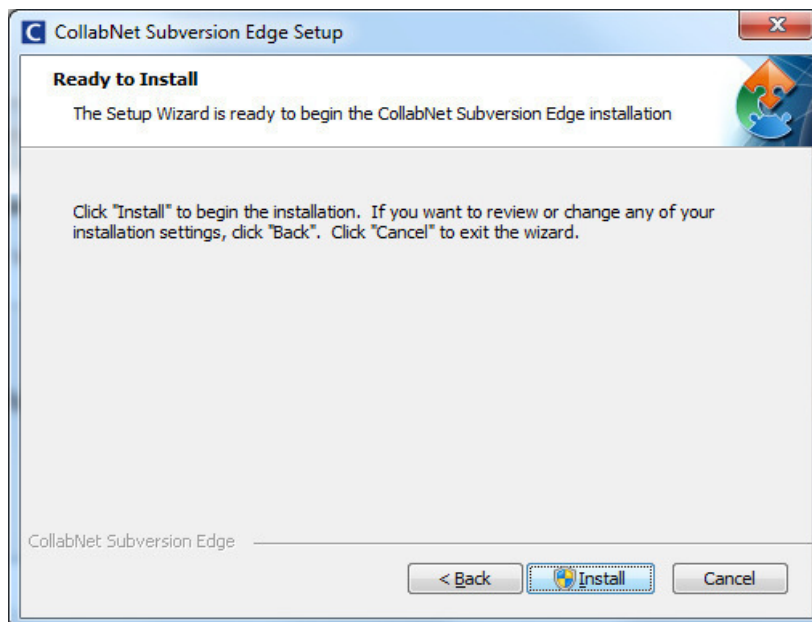
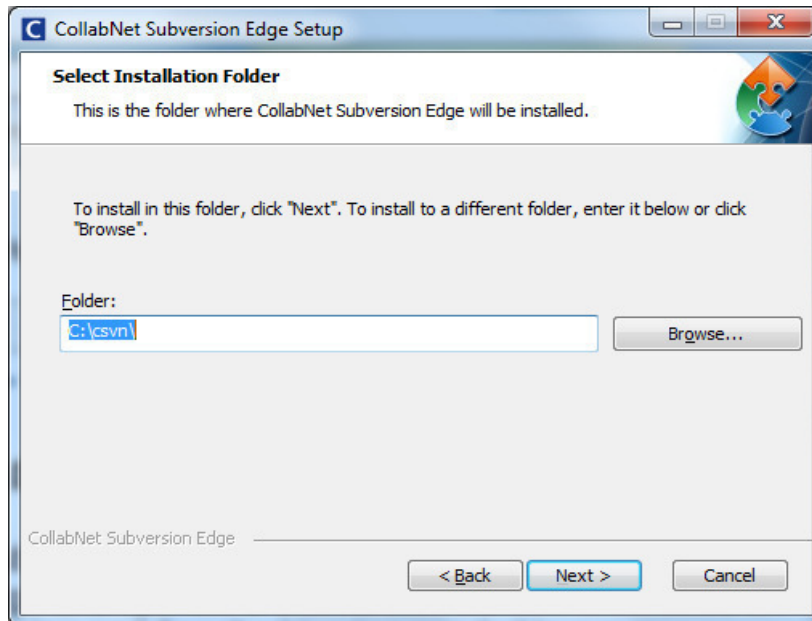


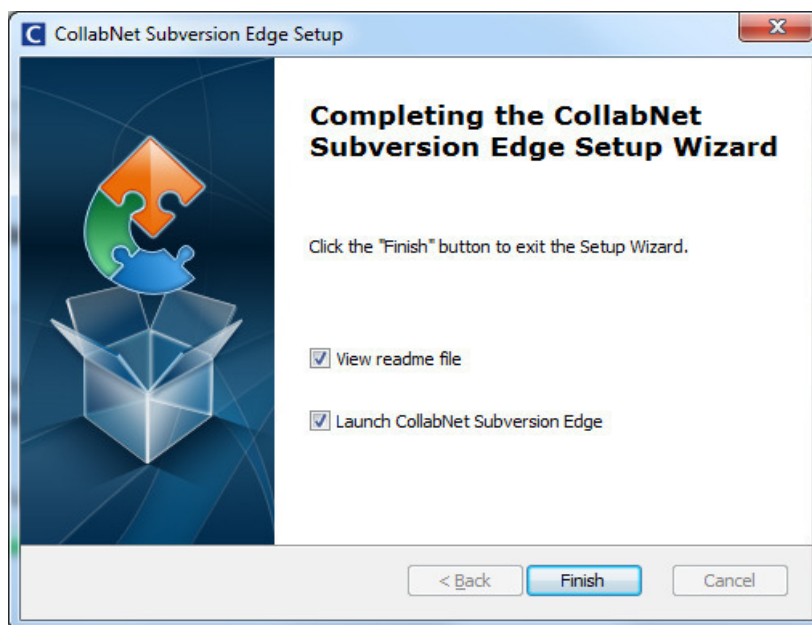
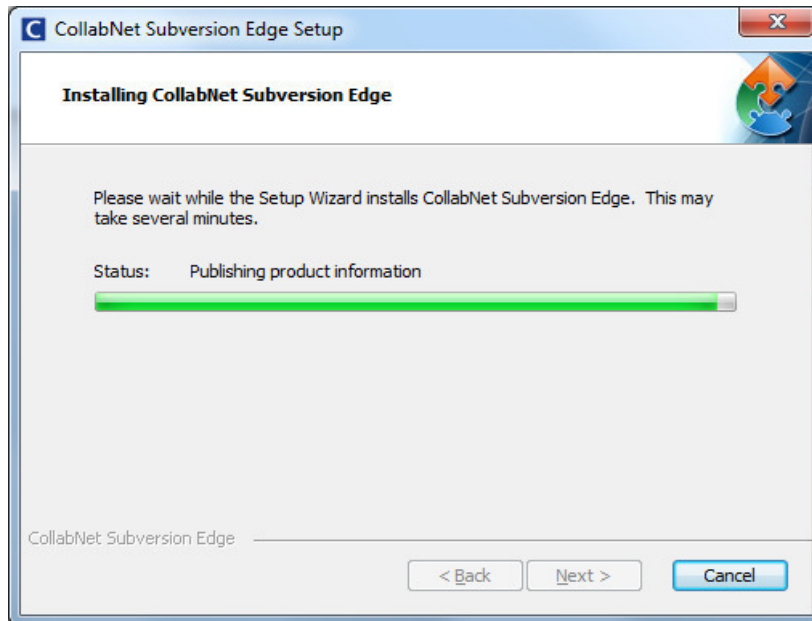
3. Ejecutar el archivo



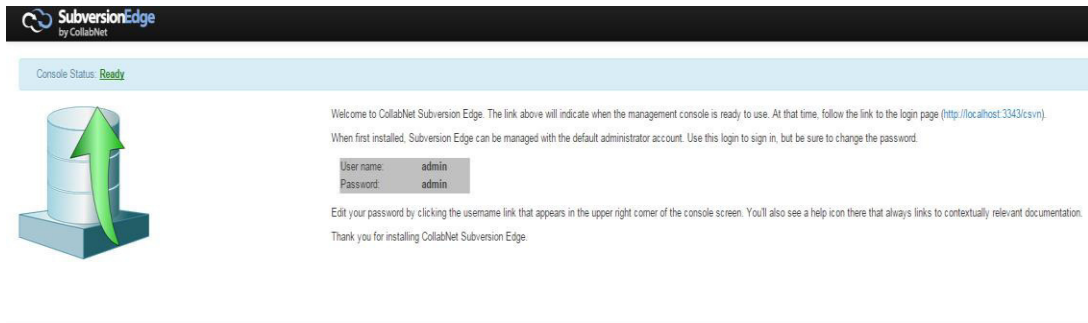
4. Seguir los pasos de instalación





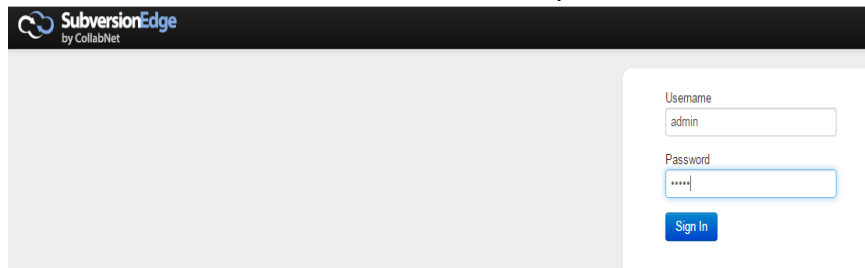


5. Se tiene la opción de iniciar la aplicación al final de la instalación. Si selecciona esta opción, el navegador abrirá una página local que contiene el enlace de la consola del inicio del servidor.

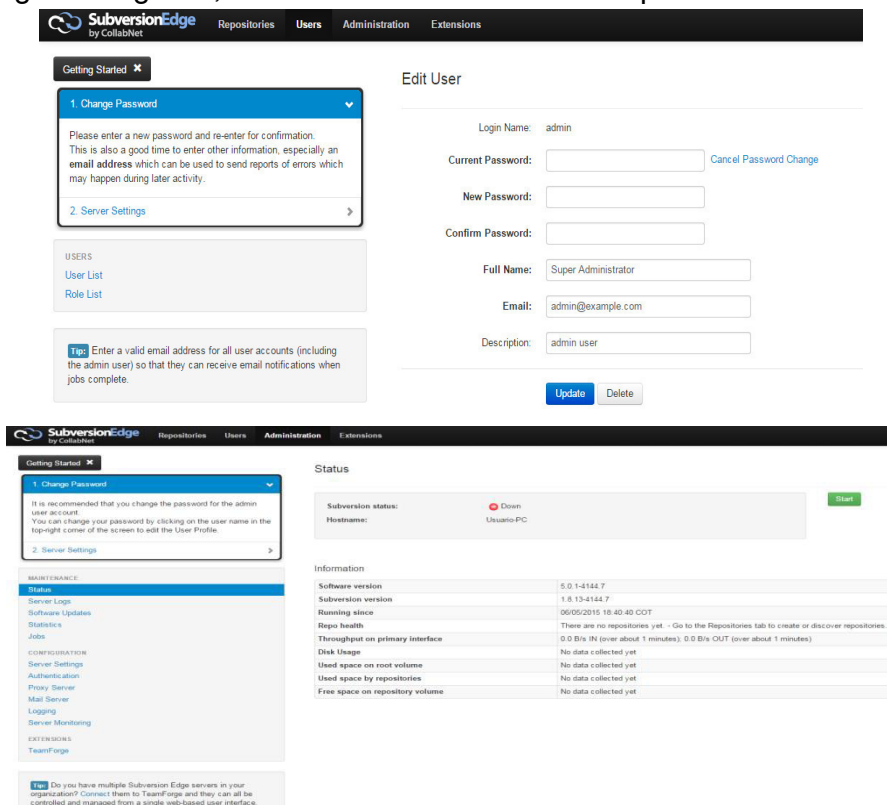


© 2015 CollabNet. CollabNet is a registered trademark of CollabNet, Inc.
 Apache Subversion is a trademark of the Apache Software Foundation.
 Subversion and the Apache Subversion logo are registered trademarks of the Apache Software Foundation.

6. Ingresar a la consola con las credenciales por defecto del administrador.

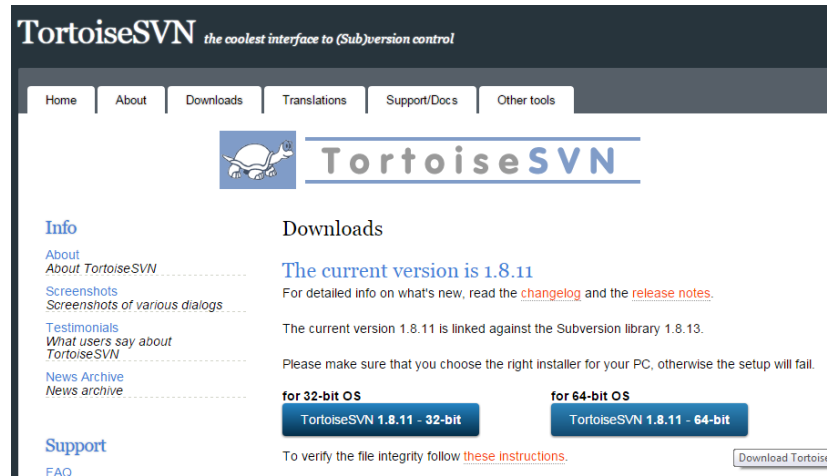


7. Luego del ingreso, cambiar la Contraseña en Super Administrator (admin)



Anexo 7: Guía para instalar TortoiseSVN

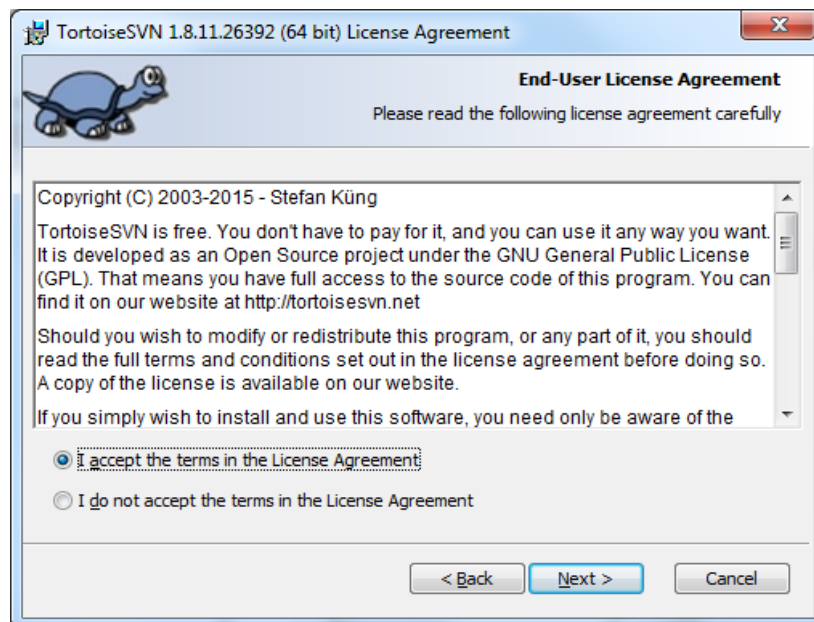
1. Ingresar a <http://tortoisesvn.net/downloads.html>

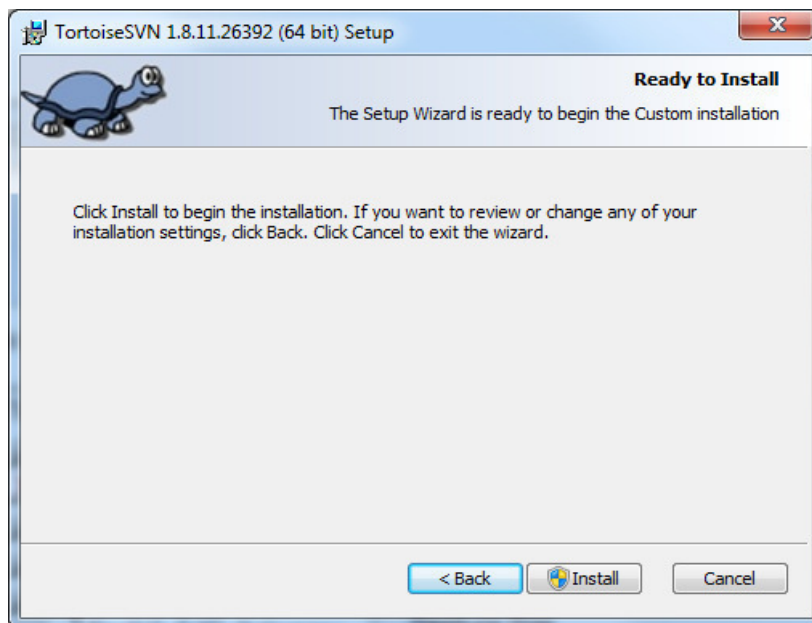
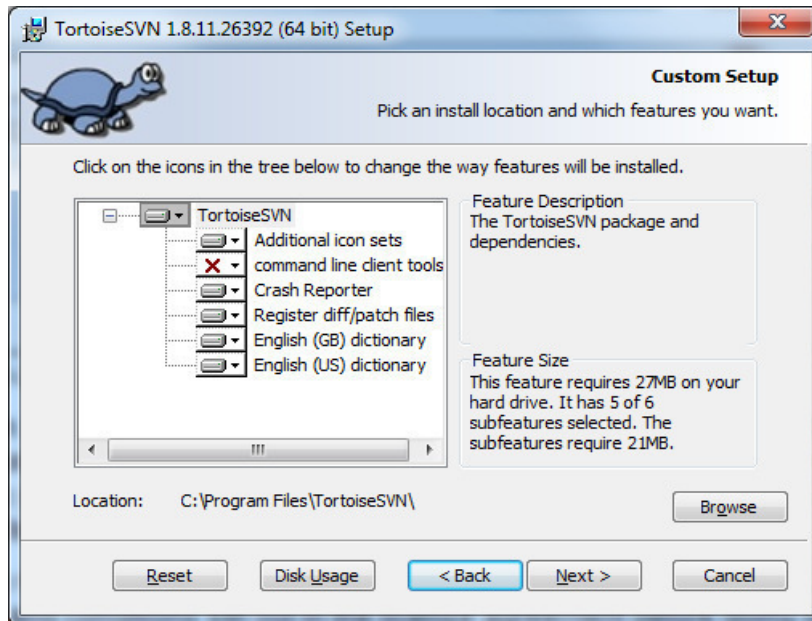


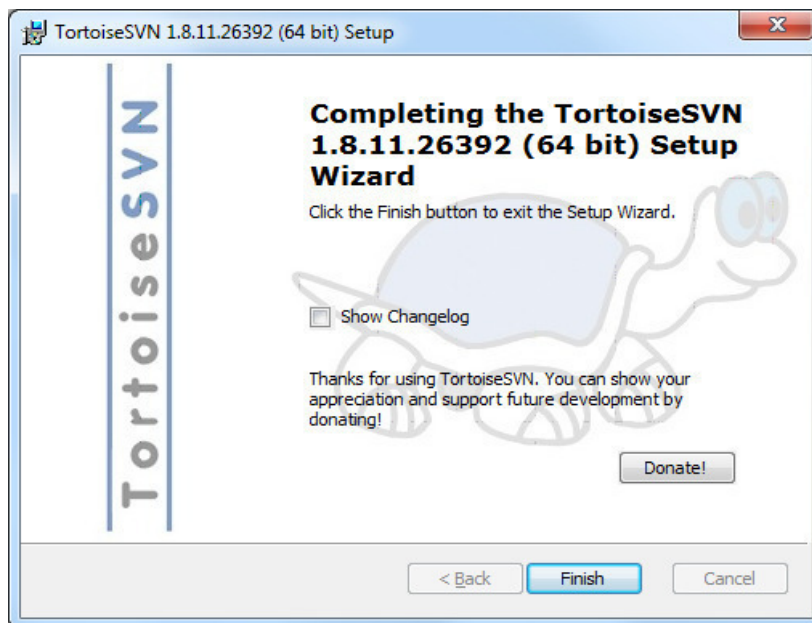
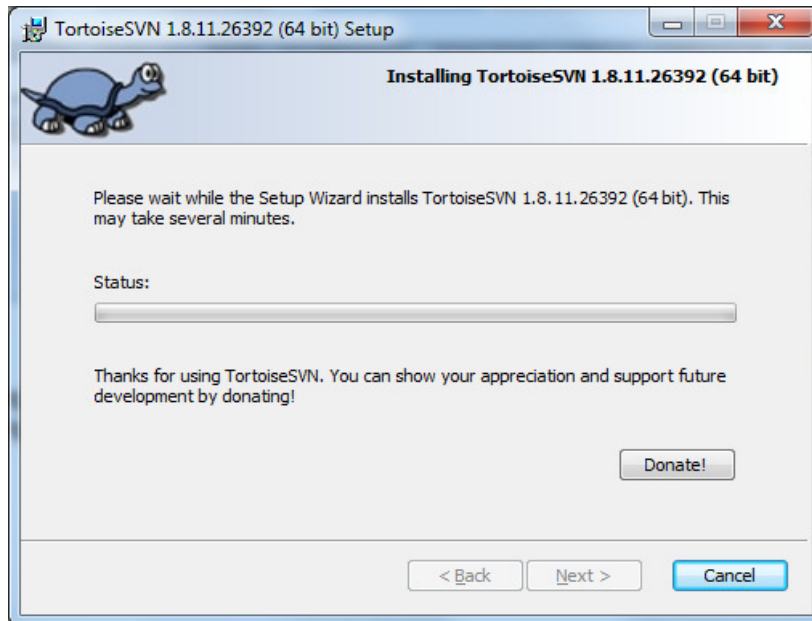
2. Ejecutar el archivo



3. Seguir los pasos de instalación







Anexo 8: Lista de Aplicativos – Programas Host

Prefijo	Nombre	Trx
ABAF	ACTIVO FIJO	AF00
ABAI	ADMINISTRACION DE INVENTARIO	AA01
ABCO	ABASTECIMIENTOS	AO00
ABDO	DOTACION BIENES	AA97
ABPE	PROCESO DE PEDIDOS	AA24
ABPL	PLANEAMIENTO Y ASIGNACION BI	AA34
ABPR	ADMINISTRACION DE PROVEEDORE	AA71
ABSI	ABASTECIMIENTOS	AB00
ABST	STOCKS	AA52
ABTA	TABLAS DEL SISTEMA	AA13
ACCI	ADM. CONTRATOS Y CONVENIOS	9999
AD01	CONTROL DE GARANTIAS	AD01
AEVM	ADM. EXPEDIENTE VENTANINTERN	AEVM
AFCS	ADM FORM CONSULTAS Y SUGEREN	AFCS
AHCO	ADMINISTRACION DE CONVENIOS	0
AHME	DEPOSITOS AHORROS ME	AE00
AHPL	PAGO DE PLANTILLAS SIAF	PL00
AHSB	REPORTES SBS	AHSB
AHTM	TRANSMISION MASIVA	AH00
AHTS	SISTEMA DEPOSITO CTS	TS00
AIEC	ENCUESTA CENSAL BN	AIEC
AIIA	INVESTIGACION ADMINISTRATIVA	AI11
AIMC	MEDIDAS CORRECTIVAS	AI00
AISC	SISTEMA SALDO CAJEROS AUTOMA	SG95
AISG	CONSULTA DE FIRMAS SIGN	SG95
AISR	SISTEMA DE RECLAMOS	SG95
ATCA	TARJETAS MULTIRED	AT01
ATCI	ADM.Y CONTROL DE TARJETAS	A760
BECO	CONCILIACIONES BCO.EXTERIOR.	BK01
BECT	CUENTAS DEL EXTERIOR	BC30
BEDG	DATOS GENERALES	BE02
BEXT	BANCOS DEL EXTERIOR	BEXT
BNBI	BONOS SITUACIONALES BN	BI00

CAVA	CONSULTAS DE AVALADOS Y AVAL	CAVA
CBCT	COBRANZAS REPUBLICA M.N.	CT00
CBDO	COBRANZAS DE LETRAS	DC00
CBME	COBRANZAS REPUBLICA M.E.	CO00
CCME	CHEQUES CERTIFICADOS M.E.	CE00
CCTC	CTAS CTES Y TARJETAS ANULADA	CCTC
CCTE	AUTORIZACIONES SIAF	TE00
CERT	UTILITARIOS CERTIFICACION	0
CGME	CHEQUES DE GERENCIA M. E.	CG80
CGMN	CHEQUES DE GERENCIA M.N	CH00
COAC	AUTOMATIZACION OFICINA CENTR	AC00
COBD	BALANCE DEUDORES	CD00
MECT	CTAS.CTES.M.E.	MC00
MEDH	DEPOSITOS	DH00
MEDU	DECRETO DE URGENCIA 217 CTS	DU00
MEPC	POSICION DE CAMBIOS	PC01
MEXJ	MONITOREO DE EXP. JUDICIALES	-
MPAM	COBRANZA POR ENCARGO MUNICIPAL	MPAM
OGCF	ORG. DE CARGOS	OF00
OPCF	CONCILIACION DE FACTURACION	IG00
OPOS	MONITOR DE CAJEROS	0
OPPE	OF.PERSONAL PENSIONES	PE00
OPSI	ADMINISTRACION PENSIONISTAS	OPSI
OXEC	COBRANZAS	OX00
OXEX	CREDITO DOCUMENTARIO	OE00
OXIC	COBRANZAS	OC00
OXIM	CREDITOS DOCUMENTARIOS	OI00
PADI	PAGOS DIVERSOS	PD00
PAPB	SISTEMA DE AUTORIZACION DE P	PAPB
PEQG	GRATIFICACION DICIEMBRE	PQ00
PFAC	PREFACT DE SERV EN ATMS	1
PGCA	TARIFARIO COMISIONES ADMINIS	PF10
PGCO	TARIFARIO DE COMISIONES	PG00
PGCU	APLICATIVO PAGO DE CUPONES	PGCU
PGFA	SOAT LA POSITIVA	PF5A

PGOL	PAGO DE FACTURAS EN LINEA	OL00
PITF	COBRO ITF	PI10
POAH	REGISTRO UNICO DE PODERES	1001
POEM	PORTAL DEL EMPLEADO	POEM
PRAH	PRESTAMOS MULTIRED	PRAH
PRCA	PRESTAMOD CORRESPONSALIA	PX00
PRHP	ADMINIS.CRE.HIPOTECARIO	PRHP
PRH1	PRESTAMOS ADMINISTRATIVOS	PRAH
PRMM	S. OTORGAMIENTO LINEA CREDIT	PRMM
PRMY	PRESTAMOS MYPE	PR00
PR01	PRESTAMOS MN.	PR01
PSJN	PROGRAMA SOCIAL JUNTOS	PS01
PTIV	PAGO DE TASAS DE INTERNET-VI	PT00
READ	PAGO DE ADUANAS	AR00
REAF	SISTEMA RECA-ADM.FILES	RF00
RECA	CONSULTA RENIEC	RC00
RECC	COBRANZA COACTIVA	RC20
RECL	RECAUDACION CARATULA DE LOTE	0
RECO	COBRANZA COACTIVA	RC01
RECQ	CONSULTA <i>CLEARQUEST</i>	CA00
REOA	OPERACIONES ADELANTADAS	RO00
REPE	REGISTRO DE PERFILES	CA00
RESI	SISTEMA INTEGRAL	RI00
RETS	TRANSMISION Y CARGA RECAUDAC	RIAC
RHAB	ASIGNACIONES Y BONIFICACIONE	RB01
RHAS	CONTROL DE ASISTENCIA	RA00
RHCG	CHEQUES DE GERENCIA	RG00
RHCP	CONTABILIZACION PAGOS PENSIO	RH80
RHDG	DATOS GENERALES	RD00
RHDL	DATOS LABORALES	RL00
RHDP	DATOS PERSONALES	RD00
RHES	MODULO DE ESCALON	RE00
RHE0	MODULO DE ESCALAFON-0	RE01
RHE2	MODULO DE ESCALAFON-2	RE01
RHHE	PGO.HORARIOS ESPEC.	RHHE
RHJD	ALIMENTISTAS PENSIONISTAS	RJ21

RHL1	MOVIM. DE PERSONAL	RL20
RHL2	HISTORICO MOVIM.	RL34
RHPC	PLLA. DE PENSIONES	RB10
RHPE	DATOS PENSIONARIOS	RR00
RHPP	PLANILLAS DE PAGOS	RP00
RHPT	PROCESOS TECNICOS	RN00
RHP1	PLANILLAS DE PAGOS-1	RP67
RHP2	PLANILLAS DE PAGOS-2	RP67
RHP3	PLANILLAS DE PAGOS-3	RP67
RHP4	PLANILLAS DE PAGOS-4	RP73
RHRV	R.H. ROL DE VACACIONES	RHNN
RHSG	CONTROL DE SEGUROS CLINICAS	RS00
RHTG	TABLAS GENERALES	RT00
RIAL	RECAUDACION	RIAL
RPC1	PLANILLA PENSIONES-1	RB11
RPE1	TPO. SERV. EN OTRAS ENT.	RR54
RPE2	PLLA. DE PAGOS PENSION-2	RB11
RPE3	PLLA. DE PAGOS PENSION-3	RB70
RRCD	REPORTE CREDITICIO DEUDORES	9999
RSIC	CONSULTA SIST. INTEG. PERSON	RHSI
RSIP	SISTEMA INTEGRAL DE PERSONAL	RH01
RV00	EVALUACION DEL PERSONAL	RV00
SADM	SIST. ADM. MULTI.	SADM
SAHE	ADMINISTRACION H.EXTRAS	SA00
SARO	SIST ADM DE RIESGOS OPERATIV	0
SATM	SISTEMA DE ADMINISTRACION DE	SA00
SCOP	ORDENES DE PAGO	SC60
SCRE	RED DE AGENCIA	SSS
SEDL	EVALUACION DESEMPEÑO LABORAL	SEDL
SGIR	AUTOMTZ. GESTION INTEG. RIES	0
SIAC	ARCHIVOS DE CONTROL	SI70
SIAI	AUDITORIA INTERNA	AI00
SIAN	ANEXOS SBS	AN00
SIBS	ADM. DATOS DE CUENTA	SIBS
SICA	CANJE NACIONAL CHEQUES	CJ00
SICE	SISTEMA CANJE SALIDA	SI11
SICF	CARTAS FIANZA	FC00

SICL	ACCESOS INTERNET USUARIOS	SC01
SICO	SISTEMA CORRESPONSALIA	SC00
SICR	POSICION CLIENTES	SICR
SICT	SECTOR ECONOMICOS	SI51
SIDG	DATOS GENERALES CLIENTES	SD00
SIEC	INGRESO DATOS PARA EVALUACIO	0
SIER	CONSULTAS DE ESTADO DE RED	ER00
SIGE	ADM.CALENDARIO DE FECHAS	SG00
SIL1	CREDITOS EN CUENTAS CORRIENT	LC01
SIMA	MANTENIMIENTO ARCHIVOS BASE	SIMA
SIMC	MODF.COLAS Y TIEMP.ESPERA	0
SIMD	ADM. TARIFARIO TABLA GESTION	SIMD
SINO	NOTAS DE CARGO Y ABONO INTRA	SINO
SINT	NOTAS DE CARGO Y ABONO	SN00
SION	SIS. OPERAC. A NIVEL NACIONAL	0
SIOP	SISTEMA INFORMAC. OPERACIONA	SI00
SIPG	PAGO DE COMPENSACIONES	S750
SIPT	POSICION CTA.PRINCIPAL TESOR	PT00
SIP1	POSICION DE CLIENTES	SP00
SIRC	MANTENIMIENTO RUC Y DNI	SM20
SIRD	RCD	SR00
SIRE	CTAS CTES MN TESORO PUBLICO	SIRE
SISP	SIS.INTEGRAL SELECCION PERSO	SI00
SITB	TRANSACCIONES BANCARIAS	TB00
SITC	TARJETA DE CREDITO	SITC
SITE	TRANSF.ELECTRONICA	CJ30
SITI	TRANSMISIONES INTRANET	SI00
SITP	AFECTACION DE SALDOS	SI81
SITR	CORRECCION DE TRANSACCIONES	SITR
SIUC	BASE DE DATOS UNICA DE CLIEN	SI00
SIU1	SIST DE INFO UNICA D CLIENTE	SIU1

SIVE	SISTEMA DE CREDITOS VENCIDOS	0
SI70	CONTORL DE CTAS CTES.	SI70
SPCL	SISTEMA DE POSICION DE CLIEN	SPCL
SPCN	PLAN CONTINUIDAD DE NEGOCIO	INT
SPC1	SISTEMA DE POSICION DE CLIEN	SPC1
SPMR	SIMU D PREST SECT. PUBLICO	0
SRCN	SISTEMA DE CONTRATOS	SN00
SRPM	SISTEMA DE GESTION INTEGRAL	0
STAI	SISTEMA DE TASAS E INTERESES	ST60
STCL	ADMINISTRACION DE CLAVES	CL01
STCT	INVENTARIO DE CINTAS MAGNETI	IC01
STSI	ADMINISTRACION DE SISTEMAS	ST10
SUPR	RECAUDACION DE PRICOS	PS00
SUTE	SUNAT TRANSF. ELECTR. DE FON	SU00
TAND	MOVIMIENTO DE CAJEROS	TA30
TBPV	TRN.BANC.PROVINCIAS	PV00
TBTF	TRANSF.EXT.-TELEFONDOS	TF00
TB00	TRANSACCIONES BANCARIAS	TB00
TDSI	SIST. DE TRASMISION DE DATOS	TD01
TERM	TERMINALES	TT00
TE00	AUTORIZACIONES	TE00
TMCC	TRANS.POR CANAL - CAJERO	TC06
TMIN	INTERFASE TARJETA MULTIRED	TM05
TOLD	DESPACHADOR TRANSACCIONES	DG00
TRAD	SISTEMA DE INVERSIONES	TRAD
TREA	SIMULADOR TASA RENDIM. EFE.A	TR00
VAAT	ADMINISTRACION DE TABLAS	VA00
VABO	CONSULTA BONO AGRARIOS	VB00
VAPA	PAGARES	VP00
VOEL	VOTACIONES ELECTRONICAS FEBA	VOEL
WSCR	WEB SERVICES CONSULTA DE PAG	WSCR

Anexo 9: Acta de Conformidad



ACTA DE CONFORMIDAD

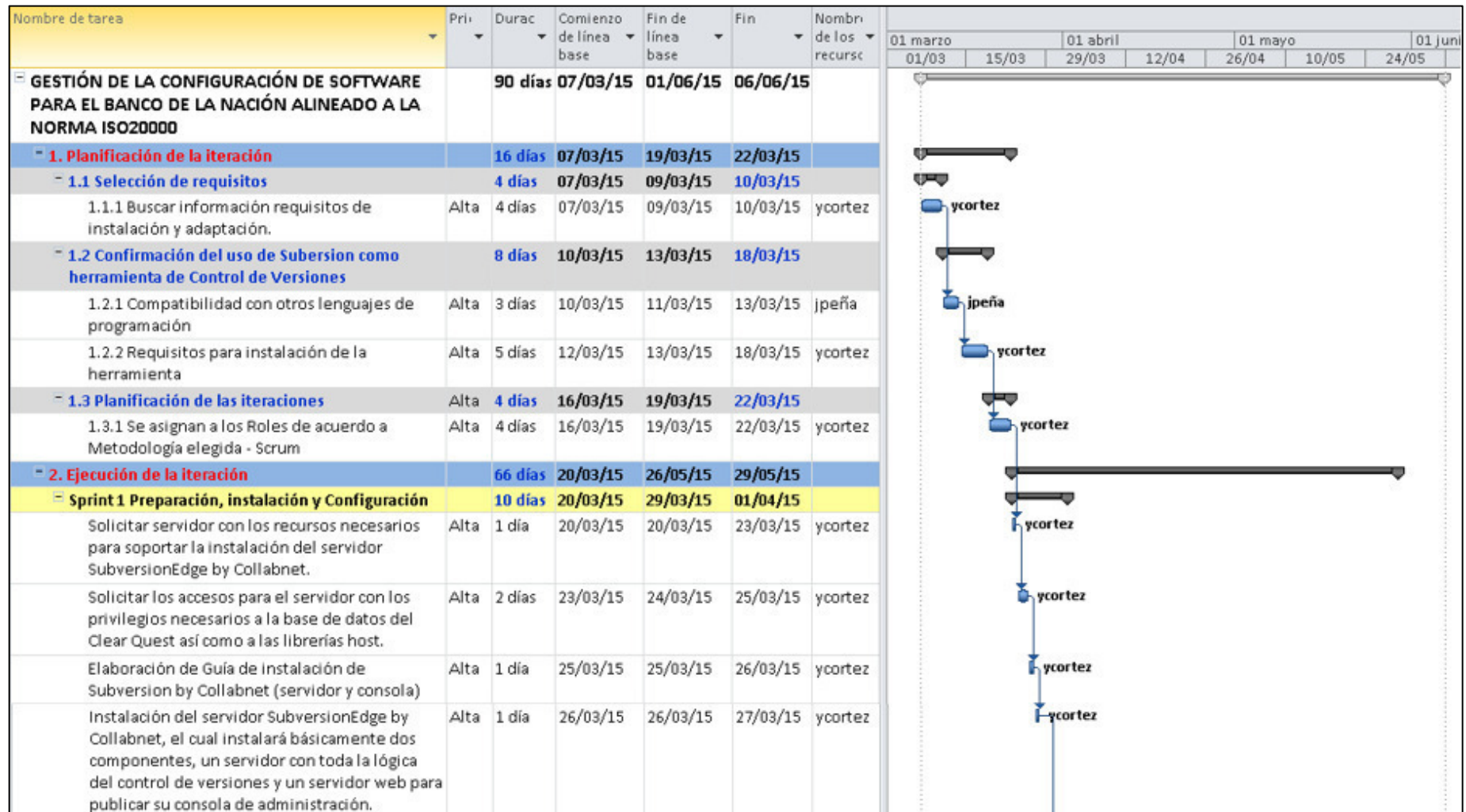
Cliente : Banco de la Nación

Mediante la presente, se da conformidad a la entrega e implantación del sistema "Cliente Subversion" junto con la metodología propuesta para la Gestión de Configuración de Software, que apoyará las actividades que se vienen realizando en relación de la Gestión de Configuración.

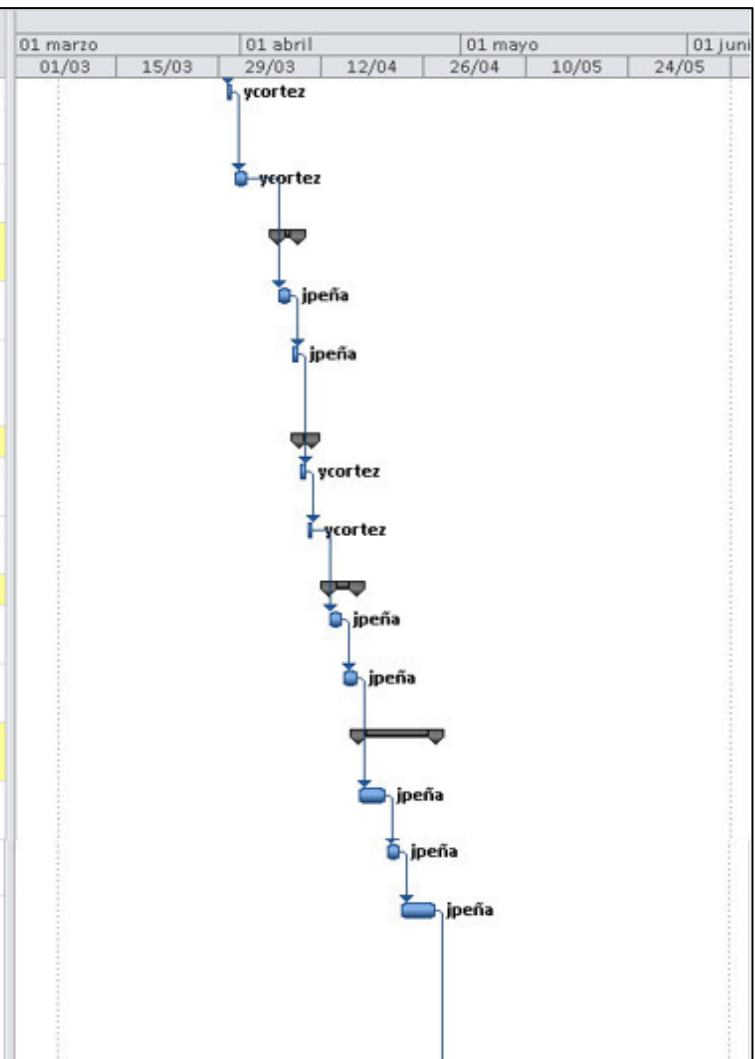
A handwritten signature in black ink, followed by a circular official stamp. The stamp contains the text: 'BANCO DE LA NACIÓN', 'Jorge M. Figueroa Revilla', 'Jefe', and 'Sección Calidad de Soluciones'.

Jorge Martín Figueroa Revilla
Jefe de la Sección Calidad de Soluciones

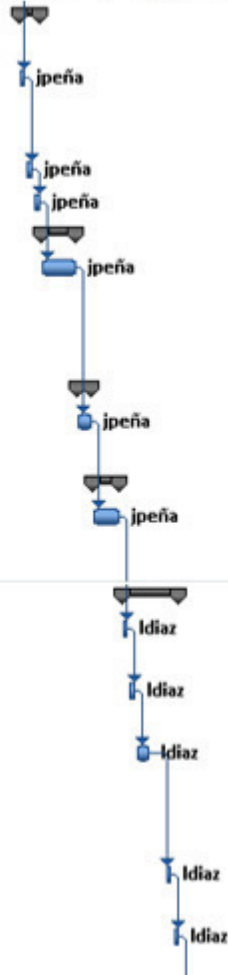
Anexo 10: Cronograma del Proyecto



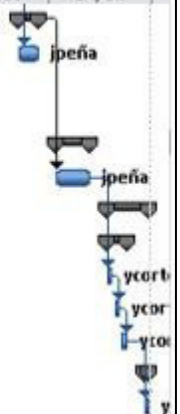
Nombre de tarea	Pri	Durac	Comienzo de línea base	Fin de línea base	Fin	Nombre de los recursos	01 marzo		01 abril		01 mayo		01 junio
							01/03	15/03	29/03	12/04	26/04	10/05	24/05
Instalación del Servidor Web para el Cliente Web que contendrá todos los módulos que se vayan desarrollando.	Alta	1 día	27/03/15	27/03/15	30/03/15	ycortez							
Configuración del servidor SubversionEdge by Collabnet.	Alta	2 días	28/03/15	29/03/15	01/04/15	ycortez							
Sprint 2 Desarrollo del enlace entre Clear Quest y Cliente Subversion		3 días	30/03/15	01/04/15	08/04/15								
Búsqueda del mantenimiento según el código registrado en el Clear Quest.	Alta	2 días	30/03/15	31/03/15	07/04/15	jpeña							
Filtro por fecha de registro, prefijo de proyecto, prioridad, título del mantenimiento y estado	Alta	1 día	01/04/15	01/04/15	08/04/15	jpeña							
Sprint 3 Registro y Autenticación de usuario		2 días	06/04/15	07/04/15	10/04/15								
Creación del módulo para la creación de nuevos usuarios.	Alta	1 día	06/04/15	06/04/15	09/04/15	ycortez							
Creación de pantalla de bienvenida y lógica de autenticación	Alta	1 día	07/04/15	07/04/15	10/04/15	ycortez							
Sprint 4 Módulo de Traslado – Cliente Subversion		4 días	08/04/15	13/04/15	16/04/15								
Creación de plantillas JCL para el traslado según el tipo de programa.	Alta	2 días	08/04/15	09/04/15	14/04/15	jpeña							
Desarrollo del programa para el procedimiento de traslado entre Desarrollo y Certificación.	Alta	2 días	10/04/15	13/04/15	16/04/15	jpeña							
Sprint 5 Módulo de versionado host – Cliente Subversion		11 días	14/04/15	22/04/15	27/04/15								
Creación de plantillas JCL para convertir un programa seleccionado a texto plano.	Alta	2 días	14/04/15	15/04/15	20/04/15	jpeña							
Creación de plantillas JCL para copiar un archivo en una carpeta temporal.	Alta	2 días	16/04/15	17/04/15	22/04/15	jpeña							
Desarrollo del programa que administra los siguientes eventos: -Armar el JCL para convertir a texto plano. -Armar el JCL para la copia del texto plano por FTP. -Mover el archivo al repositorio que le corresponde. -Ejecución de comandos SVN para el version	Alta	3 días	20/04/15	22/04/15	27/04/15	jpeña							



Nombre de tarea	Pri	Durac	Comienzo de línea base	Fin de línea base	Fin	Nombr de los recurs	01 marzo		01 abril		01 mayo		01 juni
							01/03	15/03	29/03	12/04	26/04	10/05	24/05
Sprint 6 Módulo de programas en común – Cliente subversión		3 días	23/04/15	27/04/15	30/04/15								
Filtro de mantenimientos cuyo estado se encuentra en Certificación. (Anexo : Estados de Mantenimientos)	Alta	1 día	23/04/15	23/04/15	28/04/15	jpeña							
Lista los programas según mantenimiento.	Alta	1 día	24/04/15	24/04/15	29/04/15	jpeña							
Cruce de programas en común.	Alta	1 día	27/04/15	27/04/15	30/04/15	jpeña							
Sprint 7 Log de Ejecución		5 días	28/04/15	30/04/15	05/05/15								
Desarrollo del Log de Ejecución que permita verificar las fechas y hora de los accesos, confirmación del éxito del proceso, entre otros.	Medi	3 días	28/04/15	30/04/15	05/05/15	jpeña							
Sprint 8 Flujo de estados de los mantenimientos		2 días	01/05/15	04/05/15	07/05/15								
Desarrollo de flujo de los estados de los mantenimientos entre ambientes	Medi	2 días	01/05/15	04/05/15	07/05/15	jpeña							
Sprint 9 Módulo de Consulta al Clear Quest		4 días	05/05/15	06/05/15	11/05/15								
Consulta de las actividades, historial, documentos y programas de los mantenimientos que se encuentran registrados	Medi	2 días	05/05/15	06/05/15	11/05/15	jpeña							
Sprint 10 Elaboración de guías		8 días	07/05/15	14/05/15	19/05/15								
Elaboración de guía de instalación del servidor de aplicaciones Tomcat	Medi	1 día	07/05/15	07/05/15	12/05/15	ldiaz							
Elaboración de guía de despliegue de Cliente Subversion	Medi	1 día	08/05/15	08/05/15	13/05/15	ldiaz							
Elaboración de guía de usuario del Subversion by Collabnet alineado al CVS del BN: -Creación de usuarios. -Creación de repositorios. -Reglas de acceso a los repositorios. -Log de cambios	Alta	2 días	11/05/15	12/05/15	15/05/15	ldiaz							
Elaboración de guía de usuario de Cliente Subversion	Medi	1 día	13/05/15	13/05/15	18/05/15	ldiaz							
Elaboración de guía para la instalación del Tortoise.	Medi	1 día	14/05/15	14/05/15	19/05/15	ldiaz							



Nombre de tarea	Pri.	Durac	Comienzo de línea base	Fin de línea base	Fin	Nombre de los recursos	01 marzo		01 abril		01 mayo		01 junio	
							01/03	15/03	29/03	12/04	26/04	10/05	24/05	
Sprint 11 Creación de repositorios		3 días	15/05/15	19/05/15	22/05/15									
Desarrollo y ejecución del JAR para crear un subdirectorio por cada proyecto registrado en el Clear Quest.	Alta	3 días	15/05/15	19/05/15	22/05/15	jpeña								
Sprint 12 Creación de Línea Base		5 días	20/05/15	26/05/15	29/05/15									
Creación de línea base	Alta	5 días	20/05/15	26/05/15	29/05/15	jpeña								
3. Inspección y adaptación		6 días	27/05/15	01/06/15	06/06/15									
Sprint 13 Pruebas del Producto		3 días	27/05/15	29/05/15	03/06/15									
Pruebas de Funcionalidad	Medi	1 día	27/05/15	27/05/15	01/06/15	ycortez								
Pruebas de Adaptación	Medi	1 día	28/05/15	28/05/15	02/06/15	ycortez								
Pruebas de Satisfacción	Medi	1 día	29/05/15	29/05/15	03/06/15	ycortez								
Sprint 14 Demostración - Exposición		1 día	01/06/15	01/06/15	06/06/15									
Presentación del sistema	Alta	1 día	01/06/15	01/06/15	06/06/15	ycortez								



Anexo 11: Memorandum de versionamiento de aplicaciones open


CARGO
Banco de la Nación

MEMORANDO

EF/92.2431 N° 017 - 2012

Al : **SR. HENRRY FERMÍN PAREDES OJEDA**
Sub Gerente
Jefe División Desarrollo de Sistemas de Información

Asunto : **Versionamiento de Aplicaciones – Línea Base Ambiente de Certificación**

Fecha : **San Isidro, 20 de Julio del 2012**




Es grato dirigirme a usted con la finalidad de comunicarle que la Sección Calidad de Soluciones viene implementando para el ambiente de Certificación el versionamiento de aplicativos OPEN, motivo por el cual, a fin de crear una línea base solicitamos que a partir del 01 de agosto del presente, conjuntamente con el archivo de despliegue nos remitan las fuentes del programa.

Sin otro particular, quedo a su disposición para cualquier coordinación adicional.

Atentamente,





23/6/12






Gerencia Dpto. de Informática
Div. Gestión de Tecnologías de Información
Sec. Operaciones Bancarias
Sec. Productos Bancarios
Sec. Servicios Administrativos
Sec. Canales de Atención

