

UNIVERSIDAD NACIONAL DEL ALTIPLANO**FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS****ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS**

**“SISTEMA INTEGRADO BASADO EN TECNOLOGÍAS OPEN SOURCE
HIBERNATE, SPRING Y JSF 2.0 APLICANDO PATRONES DATA ACCESS
OBJECT Y BUSINESS OBJECT PARA LA GESTIÓN DE LA INFORMACIÓN DE
COOPAIN CABANA – PUNO”**

TESIS**PRESENTADO POR:****HENRY FRANK CHILA CHOQUE****JOEL ZÓSIMO MAMANI ARENAS****TESIS PARA OPTAR EL TÍTULO PROFESIONAL DE:****INGENIERO DE SISTEMAS****PUNO – PERÚ****2014**

UNIVERSIDAD NACIONAL DEL ALTIPLANO
FACULTAD DE INGENIERÍA MECÁNICA ELÉCTRICA, ELECTRÓNICA Y SISTEMAS
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS

“SISTEMA INTEGRADO BASADO EN TECNOLOGÍAS OPEN SOURCE HIBERNATE, SPRING Y JSF 2.0 APLICANDO PATRONES DATA ACCESS OBJECT Y BUSINESS OBJECT PARA LA GESTIÓN DE LA INFORMACIÓN DE COOPAIN CABANA – PUNO”

TESIS PRESENTADA POR:

HENRY FRANK CHILA CHOQUE
JOEL ZÓSIMO MAMANI ARENAS

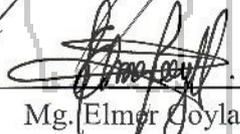
PARA OPTAR EL TÍTULO PROFESIONAL DE: INGENIERO DE SISTEMAS

APROBADA POR EL JURADO REVISOR CONFORMADO POR:

PRESIDENTE


M. Sc. Edgar Holguin Holguin

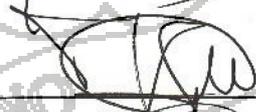
PRIMER MIEMBRO


Mg. Elmer Goyla Idme

SEGUNDO MIEMBRO


M. Sc. Elvis Augusto Aliaga Payehuanca

DIRECTOR


Ing. Hugo Yosef Gómez Quispe

ASESOR


Ing. Irenio Luis Chagua Aduviri

PUNO – PERÚ

2014

ÁREA: Informática

TEMA: Sistemas de información tradicionales y expertos

DEDICATORIA

A mi queridos padres Carmen Rosa y Manuel, no me equivoco si digo que son los mejores padres del mundo, gracias por todo su esfuerzo, su apoyo y la confianza que depositaron en mí, este es un logro que quiero compartir con ustedes. Gracias por ser mis padres y por creer en mí. Quiero que sepan que ocupan un lugar muy especial dentro de mi corazón. Los quiero mucho.

A toda mi familia, en especial a mi tía Filomena, a mi tío Hugo a mis primos Edwin, Nataly, Luis y Miguel Ángel quienes son la alegría de mi vida. A mis abuelos Aleja, Juan, Berta y Daniel.

Frank

Dedico el presente trabajo de investigación. A mis padres Fredy Mamani y Ofelia Arenas, por ser el pilar fundamental en todo lo que soy, en toda mi educación, tanto académica, como de la vida, por su incondicional apoyo perfectamente mantenido a través del tiempo, por ser artífices en la culminación de mis estudios superiores quienes con sus consejos y ayuda me dieron un impulso para salir adelante. A mis hermanos Anahí y Aldo a quienes les debo muchas cosas, quienes han vivido de cerca los distintos procesos de mi vida, por estar conmigo y apoyarme siempre.

Joel

AGRADECIMIENTOS

En primer lugar quiero agradecer a **Dios** por haberme dado la sabiduría, el entendimiento y la fortaleza para poder llegar al final de mi carrera, por no haber dejado que me rindiera en ningún momento e iluminarme para salir adelante.

Un agradecimiento muy especial a mis padres **Manuel** y por haberme dado la vida y la **Carmen Rosa** oportunidad de llegar hasta donde estoy. A todos mis amigos por su compañía y los buenos momentos que pasamos juntos, en especial a **Joel, Ramiro** y **Guido** quienes participaron en la preparación del proyecto.

Frank

A **Dios**, por haberme dado sabiduría, fortaleza, salud, coraje y no dejarme solo en los momentos difíciles, y haberme permitido llegar a la meta en este gran trabajo.

A **mis Padres**, que siempre me han dado su apoyo incondicional y a quienes debo este triunfo profesional, por todo su trabajo y dedicación para darme una formación académica y sobre todo humanista y espiritual, de ellos es este triunfo y para ellos es todo mi agradecimiento.

Para mis hermanos, **Anahí** y **Aldo**, para que también continúen superándose. A toda mi familia, muy en especial a mis tíos, **Theko** y **Oscar** por todo su apoyo.

A mis amigos **Guido, Ramiro, Frank** compañero de tesis y todas aquellas personas que con su ayuda han colaborado en la realización del presente trabajo de investigación.

Joel

Al Ing. Hugo Yosef Gómez Quispe y al Ing. Irenio Luis Chagua Aduviri por su valioso tiempo prestado en la dirección y asesoramiento de la tesis. Y de manera especial al Presidente y a los miembros del jurado, **M. Sc. Edgar Holguin Holguin, Mg. Elmer, Coyla Idme, M. Sc. Elvis Augusto, Aliaga Payhuanca** respectivamente, agradecemos su paciencia y capacidad de guiarnos durante todo el proceso de control del trabajo de investigación.

ÍNDICE

ÍNDICE DE FIGURAS.....	viii
ÍNDICE DE TABLAS.....	x
RESUMEN.....	xi
ABSTRACT	xii
INTRODUCCIÓN.....	xiii

CAPÍTULO I

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA.....	1
1.2. DEFINICIÓN DEL PROBLEMA.....	3
1.2.1. DEFINICIÓN DEL PROBLEMA GENERAL.....	3
1.2.2. DEFINICIÓN DEL PROBLEMA ESPECÍFICO.....	3
1.3. JUSTIFICACIÓN.....	3
1.4. OBJETIVOS DE LA INVESTIGACIÓN.....	4
1.4.1. OBJETIVO GENERAL.....	4
1.4.2. OBJETIVOS ESPECÍFICOS.....	4
1.5. HIPÓTESIS.....	5
1.5.1. HIPÓTESIS GENERAL.....	5
1.5.2. HIPÓTESIS ESPECÍFICOS.....	5
1.6. ALCANCES.....	5

CAPÍTULO II

MARCO TEÓRICO

2.1. ANTECEDENTES DE LA INVESTIGACIÓN.....	7
2.1.1. ANTECEDENTES NACIONALES.....	7
2.1.2. ANTECEDENTES INTERNACIONALES.....	9
2.2. MARCO CONCEPTUAL.....	10
2.2.1. SISTEMA INTEGRADO.....	10
2.2.2. TECNOLOGÍA OPEN SOURCE.....	11
2.2.3. HIBERNATE.....	13
2.2.4. SPRING.....	17
2.2.5. JAVA SERVER FACES.....	21
2.2.6. PATRONES.....	23
2.2.7. DATA ACCESS OBJET.....	25
2.2.8. BUSINESS OBJECT.....	27
2.2.9. GESTIÓN DE LA INFORMACIÓN.....	27
2.2.10. COOPAIN CABANA.....	28

2.2.11. PROGRAMACIÓN EXTREMA.....	30
2.2.12. MySQL.....	33
2.2.13. AJAX.....	33
2.2.14. JASPERS REPORTS.....	34
2.3.DEFINICIÓN DE TÉRMINOS BÁSICOS.....	35
2.3.1. SISTEMA.....	35
2.3.2. JSF.....	35
2.3.3. SPRING.....	36
2.3.4. HIBERNATE.....	36
2.3.5. GESTIÓN.....	36
2.3.6. COOPAIN CABANA.....	36
2.3.7. IoC.....	36
2.3.8. PATRÓN MVC.....	37

CAPÍTULO III MATERIALES Y MÉTODOS

3.1. TRABAJO EXPERIMENTAL.....	38
3.1.1. METODOLOGÍA DE LA INVESTIGACIÓN.....	38
3.1.2. DISEÑO DE LA INVESTIGACIÓN.....	38
3.1.3. POBLACIÓN Y MUESTRA.....	39
3.2. SISTEMA DE VARIABLES.....	40
3.2.1. DEFINICIÓN DE VARIABLES.....	40
3.2.2. OPERACIONALIZACIÓN DE VARIABLES.....	41
3.3. MATERIAL EXPERIMENTAL.....	42
3.3.1. TÉCNICAS DE RECOLECCIÓN DE DATOS.....	42
3.3.2. TRATAMIENTO DE DATOS.....	42
3.3.3. ANÁLISIS DE DATOS.....	42
3.4. METODOLOGÍA DE DESARROLLO DEL SISTEMA.....	42
3.4.1. PLANEACIÓN.....	43
3.4.2. DISEÑO.....	58
3.4.3. CODIFICACIÓN.....	76
3.4.4. PRUEBAS.....	83
3.5. MATERIAL APLICATIVO.....	83
3.5.1. HARDWARE.....	83
3.5.2. SOFTWARE.....	84
3.5.3. SERVICIOS.....	84

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS DE LAS ENCUESTAS	85
4.2. PRUEBA DE HIPÓTESIS.....	91
CONCLUSIONES.....	93
RECOMENDACIONES Y SUGERENCIAS	95
BIBLIOGRAFÍA.....	96
ANEXOS	99



ÍNDICE DE FIGURAS

Figura N° 1: Integración de JSF – Spring - Hibernate	10
Figura N° 2: Roles de las interfaces de Hibernate.....	16
Figura N° 3: Arquitectura Spring	20
Figura N° 4: Código DAO.....	26
Figura N° 5: Organigrama de la Cooperativa Agroindustrial Cabana – Puno	30
Figura N° 6: Programación Extrema XP	43
Figura N° 7: Plan de entrega de las iteraciones	47
Figura N° 8: Cronograma: Módulo Producción.....	49
Figura N° 9: Cronograma: Módulo Acopio y Almacén	50
Figura N° 10: Cronograma: Módulo Procesamiento.....	51
Figura N° 11: Cronograma: Módulo Comercialización	52
Figura N° 12: Cronograma: Módulo Control Documentario	53
Figura N° 13: Cronograma: Módulo Caja.....	54
Figura N° 14: Cronograma: Módulo Gerencia.....	55
Figura N° 15: Cronograma: Módulo Personal Administrativo	56
Figura N° 16: Esquema de la interfaz del Sistema.....	59
Figura N° 17: Interfaz final del Sistema.....	59
Figura N° 18: Interfaz Gestionar Productor	60
Figura N° 19: Interfaz Gestionar Inspecciones	60
Figura N° 20: Interfaz Generar Reporte Productor	61
Figura N° 21: Interfaz Gestionar Campañas	61
Figura N° 22: Interfaz Gestionar Campañas	61
Figura N° 23: Interfaz Gestionar Acopios.....	62
Figura N° 24: Interfaz Gestionar Lotes de Compra.....	62
Figura N° 25: Interfaz Generar Reporte Acopio	63
Figura N° 26: Interfaz Gestionar Procesamiento.....	63
Figura N° 27: Interfaz Gestionar Lotes de Venta.....	64
Figura N° 28: Interfaz Gestionar Empresas de Transporte	64
Figura N° 29: Interfaz Gestionar Clientes	65
Figura N° 30: Interfaz Gestionar Ventas	65



Figura N° 31: Interfaz Gestionar Productos	66
Figura N° 32: Interfaz Gestionar Documentos	66
Figura N° 33: Interfaz Gestionar Ingresos.....	67
Figura N° 34: Interfaz Gestionar Egresos	67
Figura N° 35: Interfaz Gestionar Pagos al Personal	68
Figura N° 36: Interfaz Generar Reportes de Caja	68
Figura N° 37: Interfaz Gestionar precios de Producción.....	69
Figura N° 38: Interfaz Gestionar precios de Comercialización.....	69
Figura N° 39: Interfaz Gestionar Personal Administrativo	70
Figura N° 40: Interfaz Generar Reporte de Personal Administrativo	70
Figura N° 41: Modelo Relacional de la Base de Datos	71
Figura N° 42: Tarjeta CRC Producción.....	73
Figura N° 43: Tarjeta CRC Acopio y Almacén.....	73
Figura N° 44: Tarjeta CRC Procesamiento	74
Figura N° 45: Tarjeta CRC Comercialización y Gerencia	74
Figura N° 46: Tarjeta CRC Control Documentario.....	75
Figura N° 47: Tarjeta CRC Caja	75
Figura N° 48: Vistas del Sistema.....	78
Figura N° 49: Paquetes del Sistema	79
Figura N° 50: Código de configuración de JSF.....	80
Figura N° 51: Código de integración de JSF y Spring	80
Figura N° 52: Código de conexión con la Base de Datos	81
Figura N° 53: Configuración de DataSource.....	81
Figura N° 54: Configuración integración de Hibernate y Spring.....	81
Figura N° 55: Configuración de seguridad Spring.....	82
Figura N° 56: Opinión con respecto a la Confiabilidad del Sistema.....	86
Figura N° 57: Opinión con respecto a la Usabilidad del Sistema	87
Figura N° 58: Opinión con respecto al tiempo de ejecución de procesos	88
Figura N° 59: Opinión con respecto a los costos de operación con el Sistema.....	89
Figura N° 60: Opinión con respecto a control de procesos con el Sistema.....	90
Figura N° 61: Distribución T.....	92

ÍNDICE DE TABLAS

Tabla N° 1: Población del Personal Administrativo.....	39
Tabla N° 2: Operacionalización de Variables	41
Tabla N° 3: Velocidad del Proyecto	45
Tabla N° 4: División en Iteraciones	46
Tabla N° 5: Planificación de Tareas: Módulo Producción	49
Tabla N° 6: Planificación de Tareas: Módulo Acopio y Almacén	50
Tabla N° 7: Planificación de Tareas: Módulo Procesamiento.....	51
Tabla N° 8: Planificación de Tareas: Módulo Comercialización	52
Tabla N° 9: Planificación de Tareas: Módulo Control Documentario	53
Tabla N° 10: Planificación de Tareas: Módulo Caja.....	54
Tabla N° 11: Planificación de Tareas: Módulo Gerencia.....	55
Tabla N° 12: Planificación de Tareas: Módulo Personal Administrativo.....	56
Tabla N° 13: Distribución de roles XP	57
Tabla N° 14: Estándares de Codificación.....	77
Tabla N° 15: Resultado general de la Encuesta.....	85
Tabla N° 16: Opinión respecto a la Confiabilidad	86
Tabla N° 17: Opinión respecto a la Usabilidad del Sistema	87
Tabla N° 18: Opinión respecto al tiempo de ejecución de procesos	88
Tabla N° 19: Opinión respecto a los costos de operación con el Sistema.....	89
Tabla N° 20: Opinión respecto al control de procesos con el Sistema.....	90

RESUMEN

El presente trabajo de investigación tuvo como objetivo principal mejorar la gestión de la información en la Cooperativa Agroindustrial (COOPAIN) Cabana - Puno. El Sistema Integrado Basado en Tecnologías Open Source aplicando Patrones Data Access Object y Business Object fue desarrollado para la Cooperativa referida, donde la gestión de la información conlleva manejar una considerable cantidad de información en sus diferentes áreas. La demora, inexactitud, redundancia y falta de información digitalizada precisa evidencias para contar con el apoyo de tecnologías de información, constituyendo una vía muy importante para el crecimiento empresarial, aumento de la productividad, reducción de costos, tiempo y mejora constante en la gestión de la Información. Para el desarrollo de la investigación se utilizó la metodología Programación Extrema XP. La selección de la muestra, no probabilística se obtuvo por la técnica de muestreo por conveniencia. Se concluyó que el Sistema mejora la Gestión de la Información ya que en su mayoría los usuarios califican como Excelente, con una ponderación de 23 en base a 25; lo que comprueba que el Sistema Integrado basado en Tecnologías Open Source mejora en un 92% en los procesos de gestión de información, permitiendo realizar los diferentes procesos con mayor control, reducción de costos y tiempo con un alto grado de productividad, en consecuencia se recomienda el uso de tecnologías Open Source ya que se ahorra en costos tales como licencia de sistema operativo entre otras ventajas y la aplicación de Patrones tales como Data Access Object y Business Object ya que facilitan la implementación de sistemas robustos que están sujetas a cambios futuros y reutilización de código.

Palabras clave: mejorar procesos, gestión de la información, patrones de diseño.

ABSTRACT

The present research had as main objective to improve the management of information in the Agroindustrial Cooperativa (COOPAIN) Cabana – Puno. Integrated System Based on Open Source Technologies Patterns using Data Access Object and Business Object was developed for the cooperative referred, where the management of information has to handle a considerable amount of information in different areas. The delay, inaccuracy, redundancy and lack of digitized information precise evidence to be supported by information technologies, constitute a very important way for business growth, increased productivity, cost reduction, time and constant improvement in the management of Information. For the development of the research methodology was used Extreme Programming XP. The selection of the sample was obtained by non-probabilistic sampling technique for convenience. It was concluded that the system improves the information management because mostly users rate as Excellent, with a weighting of 23 based on 25, which proves that the integrated system based on Open Source Technologies improves by 92% in the information management processes, allowing for different processes with greater control, cost reduction and time with a high degree of productivity, consequently the use of Open Source technologies is recommended as it will save on costs such as operating system license among other advantages and application of patterns such as Data Access Object and Business Object and facilitating the implementation of robust systems that are subject to future changes and code reuse.

Key words: process improvement, information management, design patterns.

INTRODUCCIÓN

Cuando se desarrolla un sistema, finalmente lo que se logra es que mejoren los procesos de una organización, hoy en día el manejo adecuado de la información en una organización, es un impacto estratégico y la oportunidad de tener una ventaja competitiva frente a otras organizaciones. Y teniendo en cuenta que el funcionamiento en el entorno ayuda a producir un cambio realmente significativo. El crecimiento exponencial del volumen de información que se produce en todos los sectores económicos y el consiguiente crecimiento en la complejidad de la gestión de dicho volumen de información, explica por qué los sistemas brindan mejores soluciones.

En COOPAIN Cabana, se gestiona toda la información relacionada con los diferentes procesos de cada una de sus áreas, las tareas de gestión de información como llevar el control de Producción, Procesamiento, Comercialización, Caja, Gerencia, Control Documentario y Personal Administrativo, se realiza de manera aislada y poco coordinada, dificultándose la gestión de información, ocasionando pérdida de tiempo y dinero durante estos procesos, por dichas razones se decidió desarrollar un Sistema Integrado Basado en Tecnologías Open Source, Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object para mejorar la Gestión de la información de COOPAIN Cabana – Puno.

El presente trabajo de investigación contiene los siguientes aspectos:

En el primer capítulo se detalla el planteamiento del problema, justificación, los objetivos de la investigación y posteriormente se formula la hipótesis de la investigación que se quiere demostrar y se establece el escenario de investigación.

En el segundo capítulo se desarrolla el marco teórico, se constituyen los antecedentes considerados en el trabajo de investigación, el marco conceptual comprende todo lo relacionado con los términos utilizados en la investigación, describiéndose sintéticamente algunos de los principales conceptos.

En el tercer capítulo se detalla los métodos e instrumentos que se utilizó en la investigación; también se determina el tipo de investigación cuantitativa con diseño cuasi experimental; sistema de variables, material experimental, métodos de recopilación de datos, método de tratamiento de datos y metodología de desarrollo.

En el cuarto capítulo, denominado resultados y discusión, está constituido por la prueba de hipótesis.

Finalmente se tiene las conclusiones alcanzadas en la investigación, las recomendaciones respectivas y los anexos.



CAPÍTULO I.

PLANTEAMIENTO DE LA INVESTIGACIÓN

1.1. PLANTEAMIENTO DEL PROBLEMA.

La Cooperativa Agroindustrial Cabana - Puno (COOPAIN Cabana), de gran prestigio a nivel regional, nacional e internacional que agrupa a profesionales de las diferentes especialidades además de contar con productores asociados del lugar. En COOPAIN Cabana, existen múltiples deficiencias en el manejo e integración de la información como llevar el control de Producción, Procesamiento, Comercialización, Caja, Gerencia, Control Documentario y Personal Administrativo.

En el área de Producción se realiza el registro manual de asociaciones, productores, sectores, parcelas, unidades productivas, inspecciones, acopios y fichas de unidades productivas. Se tiene dificultades para saber el historial de un productor y a qué asociación pertenece. No se cuenta con información en tiempo real, debido a la carencia de información digitalizada en el área referida. No se cuenta con un historial de registro de entradas y/o salidas (planta de procesamiento) de cantidad de quinua.

En el área de Procesamiento se realiza el registro manual de procesos, lotes de venta, partes de salida y muestras de pre embarque. Se tiene dificultades en: contar con un historial de la cantidad de quinua procesada, peso del lote acopiado, fecha del proceso, color de quinua procesada, almacén proveniente, fecha de envío, cantidad enviada y/o sobrante del lote, descripción de la unidad de transporte, detalles de envío, si se hizo en

sacos o en bolsas, peso bruto, peso neto y entre otros datos. En conclusión se tiene dificultades para el manejo y control de dichos procesos.

En el área de Comercialización se realiza el registro manual de Clientes, Ventas de productos. Se tiene dificultades para tener el control de ventas a un cliente, cantidad vendida en Kilogramos, peso de sacos en Kilogramos, si la venta ha sido iniciada o finalizada, no se cuenta con la información del stock de un determinado producto perteneciente a un almacén, carece de un historial de clientes clasificados por tipo de persona (persona natural o jurídica).

En el área de caja se hace un registro digitalizado en Excel, pero se tienen algunos inconvenientes en el registro de ingresos, egresos, pago personal, aporte de socios y desembolso de requerimientos, en el sentido de que se tiene un pésimo manejo de datos.

En el área de Gerencia se determina el precio de productos en comercialización y/o producción por lo que no se cuenta con un historial digitalizado de precios de productos de años anteriores. También se tiene dificultades en el manejo de documentos (oficios, informes, cartas, memorándums, solicitudes y requerimientos) ya que es un proceso engorroso, adicionalmente demanda mucho tiempo en la recepción y/o revisión de los mismos, algunas veces se extravían en el trayecto y no se cuenta con un historial de documentos enviados y/o recepcionados.

En el área de Personal Administrativo no se tiene un registro digitalizado del personal, no se cuenta con un historial de dichos personales de años anteriores ya sean datos como fecha de contrato, salarios, cargo etc.

1.2. DEFINICIÓN DEL PROBLEMA.

1.2.1. DEFINICIÓN DEL PROBLEMA GENERAL.

¿En qué medida el Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y JSF 2.0 aplicando Patrones Data Access Object y Business Object mejora la Gestión de la Información de COOPAIN Cabana - Puno?.

1.2.2. DEFINICIÓN DEL PROBLEMA ESPECÍFICO.

- ¿En qué medida la obtención de requerimientos afecta el desarrollo del Sistema?.
- ¿En qué medida las Tecnologías Open Source facilita el desarrollo del Sistema?.
- ¿Cómo los patrones Data Access Object y Business Object ayuda a la implementación del Sistema?.
- ¿En qué medida se mejora en tiempo, costo y control en la Gestión de la Información?.

1.3. JUSTIFICACIÓN.

El gran avance de la tecnología y las bondades de las tecnologías de información nos obliga a integrarnos a ellos, puesto que en la actualidad las instituciones públicas y privadas van adaptándolas, del cual COOPAIN Cabana no debe ser ajena a ello.

Mejorar los procesos de Gestión de la información es un factor clave, ya que influye directamente en la calidad de organización. La demora, inexactitud, redundancia y la falta de información digitalizada precisa evidencias claras para contar con el apoyo de tecnologías de información constituyendo una vía muy importante para el crecimiento empresarial, aumento de la productividad, reducción de costos, tiempo y mejora constante en el manejo organizado de la Información de COOPAIN Cabana. Por lo que

se tiene por propósito automatizar e integrar las diferentes áreas haciendo uso de las Tecnologías Open Source que cubre con las necesidades de COOPAIN Cabana entre sus primordiales beneficios su escalabilidad y soporte para múltiples sistemas operativos.

Por las razones expuestas anteriormente es indispensable desarrollar un Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y JSF 2.0 aplicando Patrones Data Access Object y Business Object para mejorar la Gestión de la Información de COOPAIN Cabana. Cuyo desarrollo fortalecerá el logro óptimo de los objetivos Empresariales.

1.4. OBJETIVOS DE LA INVESTIGACIÓN.

1.4.1. OBJETIVO GENERAL.

Desarrollar un Sistema Integrado Basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object para mejorar la Gestión de la información de COOPAIN Cabana – Puno.

1.4.2. OBJETIVOS ESPECÍFICOS.

- Obtener los requerimientos de COOPAIN Cabana – Puno.
- Analizar y Diseñar el Sistema Integrado Basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0.
- Implementar el Sistema Integrado aplicando Patrones Data Access Object y Business Object, además de MySQL como Gestor de Base de Datos para COOPAIN Cabana – Puno.
- Evaluar la mejora de los procesos de Gestión de la información de COOPAIN Cabana – Puno con ayuda del Sistema.

1.5. HIPÓTESIS.

1.5.1. HIPÓTESIS GENERAL.

El Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object mejora la Gestión de la Información de COOPAIN Cabana – Puno.

1.5.2. HIPÓTESIS ESPECÍFICOS.

- Los requerimientos permiten el desarrollo del Sistema Integral para mejorar la Gestión de la Información de COOPAIN Cabana.
- El Análisis y Diseño Basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 facilita el desarrollo del Sistema Integral para mejorar la Gestión de la Información de COOPAIN Cabana.
- La implementación aplicando Patrones Data Access Object y Business Object facilita el desarrollo del Sistema Integral para mejorar la Gestión de la Información de COOPAIN Cabana.
- Las pruebas realizadas al Sistema Integral mejora en tiempo, costo y control en los procesos de Gestión de la Información de COOPAIN Cabana – Puno.

1.6. ALCANCES.

El sistema es propuesto para ser utilizado en Cooperativas Agroindustriales por que se adecuan al caso de estudio de la investigación, también la investigación trata de dar a conocer, de cierta manera que el uso de tecnologías Open Source es adecuado para el desarrollo de aplicaciones empresariales por sus estándares y especificaciones.

Otro de los alcances más próximos de la implementación de este sistema, es que permitirá que se ahorre una gran cantidad de tiempo y esfuerzo a medida de que lo que actualmente se hace de manera mecánica pueda automatizarse, de forma que se obtenga un control mucho más extenso y eficiente para su administración.



CAPÍTULO II.

MARCO TEÓRICO

2.1. ANTECEDENTES DE LA INVESTIGACIÓN.

2.1.1. ANTECEDENTES NACIONALES.

1. FLORES, J. (2004)¹. *Portal Web Académico utilizando Herramientas con Licencia GNU GPL para optimizar la Gestión Académica de la Universidad Nacional del Altiplano*. Universidad Nacional del Altiplano - Puno. “Hipótesis: El Portal Web Académico utilizando Herramientas con Licencia GNU GPL optimiza la gestión Académica en la Universidad Nacional del Altiplano de Puno; llegando a las siguientes conclusiones: El Desarrollo de un Portal Académico utilizando Herramientas con Licencia GNU GPL optimiza la gestión Académica en el Universidad Nacional del Altiplano, los requerimiento del análisis se diseñó del portal Web académico empleando human computer Interface y conceptos de usabilidad para permitir interfaces amigables para el usuario y que responder a los requerimientos de este. Para la Implementación del Portal se utilizó herramientas con Licencia GNUGPL las cuales no tienen Costo de licenciamiento. De acuerdo a las pruebas y encuestas se obtuvo que los estudiantes y Docentes se adaptan al Uso de Portal Web Académico en las Gestiones que realiza”.

¹ FLORES MOROCO, Juan Antonio: UNAP, FIMEES, 2004.

2. POCOHUANCA, G., LUQUE, D. (2011)². *Sistema basado en la Plataforma JEE para optimizar la gestión de la Oficina de Administración Tributaria y Rentas de la Municipalidad Provincial de Huancané*. Universidad Nacional del Altiplano - Puno. “Hipótesis: El Sistema basado en la Plataforma JEE optimizará los procesos de gestión de la Oficina de Administración Tributaria y Rentas de la Municipalidad Provincial de Huancané; llegando a las siguientes conclusiones: Con el Sistema basado en la Plataforma JEE optimiza en un 90% en los procesos de gestión de la Oficina de Administración Tributaria y Rentas de la Municipalidad Provincial de Huancané, De acuerdo a las pruebas y encuestas se obtuvo que los trabajadores Administrativos y Contribuyentes se benefician y adaptan al uso del Sistema. La aceptación del Trabajador Administrativo fue total y aplicando la prueba de Hipótesis con significancia de 5% se obtuvo que $t=16.78 > t_{0.05} = 2.132$ concluyéndose que el sistema Optimiza en los procesos de gestión de la Oficina de Administración Tributaria y Rentas”.

² POCOHUANCA CHOQUEHUANCA, Guido, LUQUE CUTIPA, Diana Ruth: UNAP, FIMEES, 2011.

2.1.2. ANTECEDENTES INTERNACIONALES.

1. HERNÁNDEZ, P. (2004)³. *Herramienta Interactiva de Apoyo en Procesos Administrativos basado en Tecnología NORAS*. Universidad de las Américas Puebla. “El objetivo principal del trabajo propuesto es aplicar las tecnologías existentes de acceso y recuperación de información en ambientes distribuidos como una extensión más de los conocidos e-services aplicados al desarrollo de sistemas de apoyo a las actividades administrativas, en general a través de un conjunto de servicios disponibles en la WEB que trabajan sobre infraestructura de intercambio de información entre un conjunto de usuarios distribuidos. El principal alcance y aportación de la presente tesis es la generación de un software de carácter administrativo que permita cumplir con los requerimientos y que permita satisfacer las necesidades expuestas. Llegando a las siguientes conclusiones: El llevar a cabo esta tesis implica la generación de un software de propósito administrativo que permita reducir las tareas que hasta ahora son de carácter mecánico, lo que se reflejara directamente en la automatización de procesos y la reducción de tiempos. Es importante darse cuenta de las limitaciones y retos que surgen al realizar este proyecto, pues hay que tomar en cuenta que se tienen que construir módulos que solucionan tareas específicas y que tratan de satisfacer las necesidades de la mayor cantidad de usuarios posibles”.

³ HERNÁNDEZ JESÚS, Pablo Gustavo: Universidad de las Américas Puebla, Departamento de Ingeniería en Sistemas Computacionales, 2004.

2.2. MARCO CONCEPTUAL.

2.2.1. SISTEMA INTEGRADO.

Un Sistema Integral de Información (SII), es un sistema de información que hace uso intensivo y extensivo de las Tecnologías de la Información y las Comunicaciones (TIC) para integrar o centralizar la gestión de la información dentro de una organización. Un Sistema Integral de Información soporta todos los procesos de negocio y de soporte de la organización.

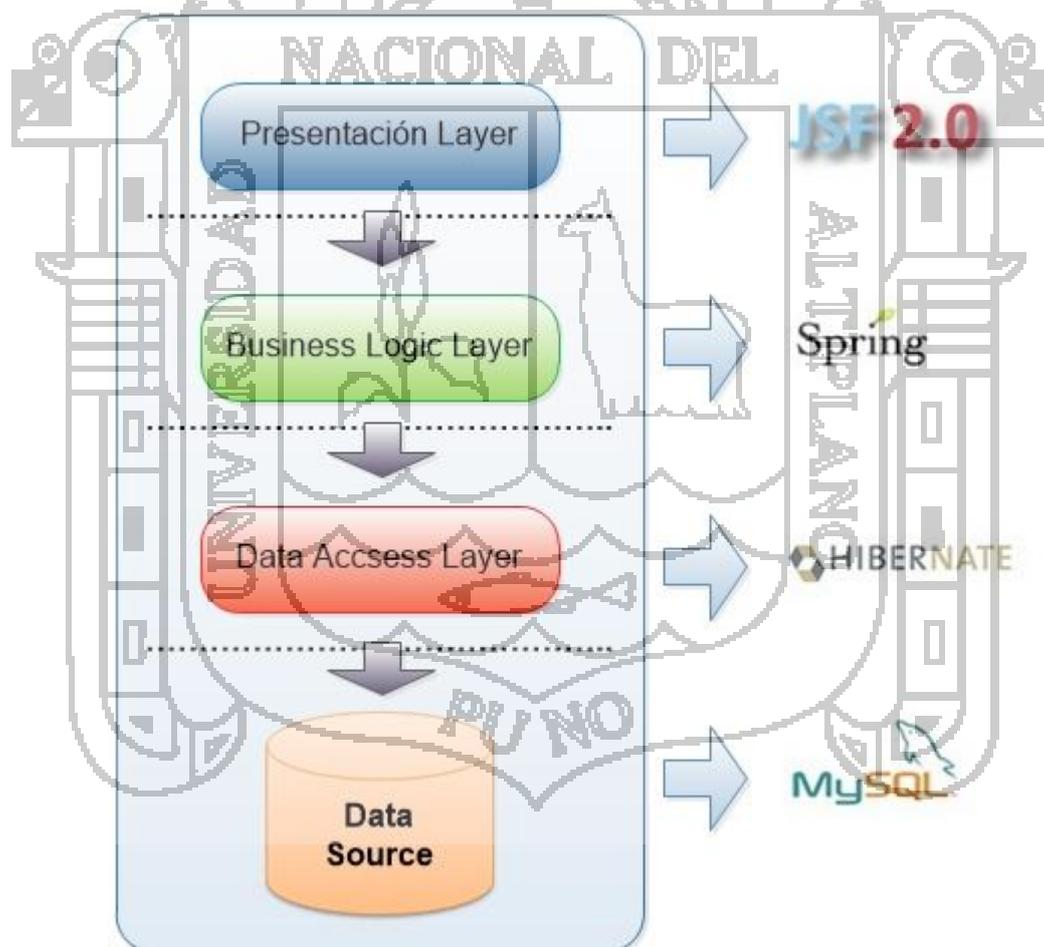


Figura N° 1: Integración de JSF – Spring - Hibernate

Fuente: (TAPIA, 2011)

2.2.2. TECNOLOGÍA OPEN SOURCE.

Código abierto es la expresión con la que se conoce al software distribuido y desarrollado libremente. Se focaliza en los beneficios prácticos (acceso al código fuente) que en cuestiones éticas o de libertad que tanto se destacan en el software libre.

Código abierto (en inglés Open Source) es el término con el que se conoce al software distribuido y desarrollado libremente.

El Código abierto tiene un punto de vista más orientado a los beneficios prácticos de compartir el código que a las cuestiones morales y/o filosóficas las cuales destacan en el llamado software libre.

Su uso nació por primera vez en 1998 de la mano de algunos usuarios de la comunidad del software libre, tratando de usarlo como reemplazo al ambiguo nombre original en inglés del software libre (free software). Free en inglés significa dos cosas distintas dependiendo del contexto: gratuidad y libertad. Lo cual implica, para el caso que nos ocupa, “software por el que no hay que pagar” (software gratuito) y además software libre, según la acepción española de libertad.

Se prefiere el uso del término software libre para referirse a programas que se ofrecen con total libertad de modificación, uso y distribución bajo la regla implícita de no modificar dichas libertades hacia el futuro.

La idea bajo el concepto de código abierto es sencilla: cuando los programadores (en internet) pueden leer, modificar y redistribuir el código fuente de un programa, éste

evoluciona, se desarrolla y mejora. Los usuarios lo adaptan a sus necesidades, corrigen sus errores a una velocidad impresionante, mayor a la aplicada en el desarrollo del software convencional o cerrado, dando como resultado la producción de un mejor software.

La idea del código abierto se centra en la premisa de que al compartir el código, el programa resultante tiende a ser de calidad superior al software propietario, es una visión técnica. Por otro lado, el software libre tiene tendencias filosóficas e incluso morales: el software propietario, al no poder compartirse, es “antiético” dado que prohibir compartir entre seres humanos va en contra del sentido común.

Al igual que el software libre, el código abierto u Open Source tiene una serie de requisitos necesarios para que un programa pueda considerarse dentro de este movimiento, éstos son:

- Libre redistribución: el software debe poder ser regalado o vendido libremente.
- Código Fuente: el código fuente debe estar incluido u obtenerse libremente.
- Trabajos derivados: la redistribución de modificaciones debe ser permitida.
- Integridad del código fuente del autor: las licencias pueden requerir que las modificaciones sean redistribuidas solo como parches.
- Sin discriminación de personas o grupos: nadie puede dejarse fuera.

Sin discriminación de áreas de iniciativa: los usuarios comerciales no pueden ser excluidos (KAPUR, 2010).

2.2.3. HIBERNATE.

Hibernate es una herramienta de Mapeo objeto-relacional (ORM) para la plataforma Java (y disponible también para .Net con el nombre de NHibernate) que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones.

Como todas las herramientas de su tipo, Hibernate busca solucionar el problema de la diferencia entre los dos modelos de datos coexistentes en una aplicación: el usado en la memoria de la computadora (orientación a objetos) y el usado en las bases de datos (modelo relacional). Para lograr esto permite al desarrollador detallar cómo es su modelo de datos, qué relaciones existen y qué forma tienen.

Con esta información Hibernate le permite a la aplicación manipular los datos de la base operando sobre objetos, con todas las características de la POO. Hibernate convertirá los datos entre los tipos utilizados por Java y los definidos por SQL.

Hibernate genera las sentencias SQL y libera al desarrollador del manejo manual de los datos que resultan de la ejecución de dichas sentencias, manteniendo la portabilidad entre todos los motores de bases de datos con un ligero incremento en el tiempo de ejecución.

Hibernate está diseñado para ser flexible en cuanto al esquema de tablas utilizado, para poder adaptarse a su uso sobre una base de datos ya existente. También tiene la funcionalidad de crear la base de datos a partir de la información disponible.

Hibernate ofrece también un lenguaje de consulta de datos llamado HQL, al mismo tiempo que una API para construir las consultas programáticamente (conocida como Criteria).

Hibernate para Java puede ser utilizado en aplicaciones Java independientes o en aplicaciones Java EE, mediante el componente HibernateAnnotations que implementa el estándar JPA, que es parte de esta plataforma (BAUER, 2004).

CARACTERÍSTICAS:

- No intrusivo (estilo POJO- Plain Old Java Object)
- Muy buena documentación (forums para ayuda, libro)
- Comunidad activa con muchos usuarios
- Transacciones, caché, asociaciones, polimorfismo, herencia, lazy loading, persistencia transitiva, estrategias de fetching.
- Potente lenguaje de consulta (HQL): subqueries, outer joins, ordering, proyección (report query), paginación.
- Fácil testeo.
- No es estándar.

Hibernate Query Language HQL

Hibernate nos proporciona además un lenguaje para el manejo de consultas a la base de datos.

Este lenguaje es similar a SQL y es utilizado para obtener objetos de la base de datos según las condiciones especificadas en el HQL.

El uso de HQL nos permite usar un lenguaje intermedio que según la base de datos que usemos y el dialecto que especifiquemos será traducido al SQL dependiente de cada base de datos de forma automática y transparente.

Arquitectura

El API de Hibernate es una arquitectura de dos capas (Capa de persistencia y Capa de Negocio).

En la siguiente Figura se muestran los roles de las interfaces Hibernate más importantes en las capas de persistencia y de negocio de una aplicación JEE. La capa de negocio está situada sobre la capa de persistencia, debido a que actúa como un cliente de la capa de persistencia (BAUER, 2004).

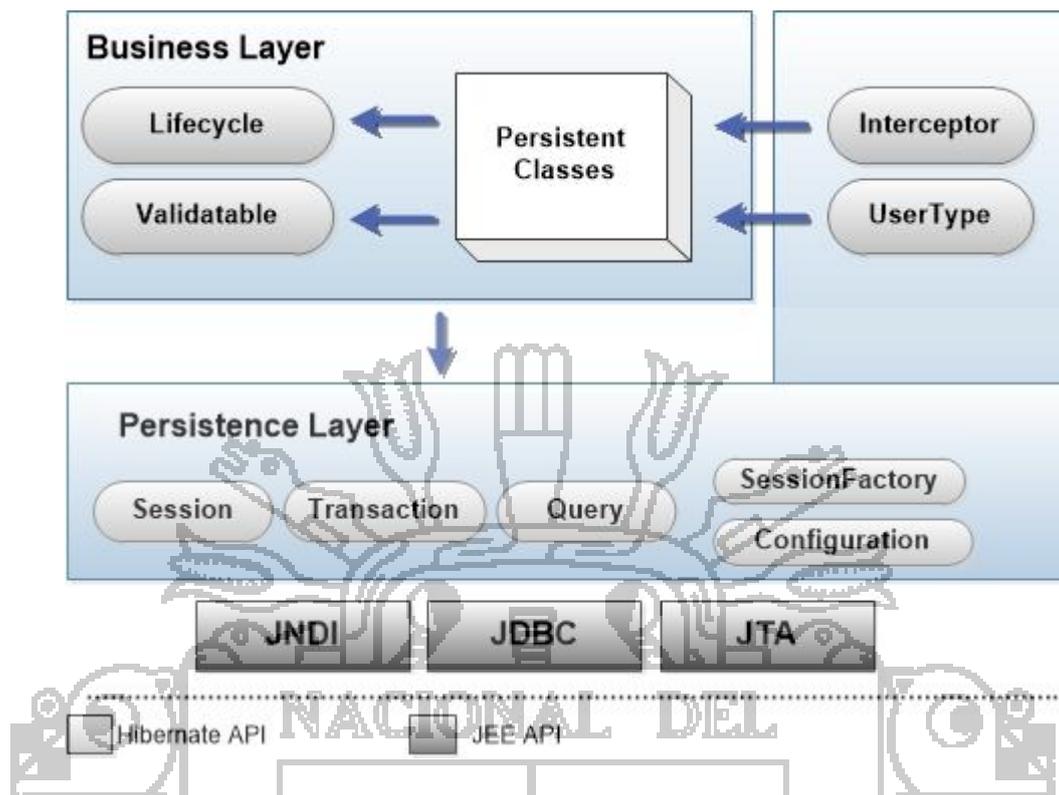


Figura N° 2: Roles de las interfaces de Hibernate

Fuente: (BAUER, 2004)

Las Interfaces mostradas se clasifican de la siguiente forma:

- Interfaces llamadas por la aplicación para realizar operaciones básicas:
 - ✓ Session: interfaz primaria utilizada en cualquier aplicación Hibernate (SessionFactory).
- Transaction
- Query: permite realizar peticiones a la base de datos y controla cómo se ejecuta dicha petición (query). Las peticiones se escriben en HQL o en el dialecto SQL nativo de la base de datos que estamos utilizando. Una instancia Query se utiliza para enlazar los parámetros de la petición, limitar el número de resultados devueltos por la petición y para ejecutar dicha petición.

- Interfaces llamadas por el código de la infraestructura de la aplicación para configurar Hibernate. La más importante es la clase Configuration: se utiliza para configurar y "arrancar" Hibernate. La aplicación utiliza una instancia de Configuration para especificar la ubicación de los documentos que indican el mapeado de los objetos y propiedades específicas de Hibernate, y a continuación crea la Session Factory.
- Interfaces callback que permiten a la aplicación reaccionar ante determinados eventos que ocurren dentro de la aplicación, tales como **Interceptor, Lifecycle, y Validatable.**
- Interfaces que permiten extender las funcionalidades de mapeado de Hibernate, como por ejemplo **UserType, CompositeUserType, e IdentifierGenerator.**

Además, Hibernate hace uso de APIs de Java, tales como JDBC, JTA (Java Transaction Api) y JNDI (Java Naming Directory Interface) (BAUER, 2004) .

2.2.4. SPRING.

Spring es un framework de aplicaciones Java/JEE desarrollado usando licencia de OpenSource. Se basa en una configuración a base de javabeans bastante simple. Es potente en cuanto a la gestión del ciclo de vida de los componentes y fácilmente ampliable. Es interesante el uso de programación orientada a aspectos (IoC). Tiene plantillas que permiten un más fácil uso de Hibernate, iBatis, JDBC., se integra "de fábrica" con Quartz, Velocity, Freemarker, Struts, Webwork2 y tienen un plugin para eclipse.

Ofrece un ligero contenedor de bean para los objetos de la capa de negocio, DAOs y repositorio de Datasources JDBC y sesiones Hibernate. Mediante un XML definimos el

contexto de la aplicación siendo una potente herramienta para manejar objetos Singleton o “factorias” que necesitan su propia configuración.

El objetivo de Spring es no ser intrusito, aquellas aplicaciones configuradas para usar beans mediante Spring no necesitan depender de interfaces o clases de Spring, pero obtienen su configuración a través de las propiedades de sus beans. Este concepto puede ser aplicado a cualquier entorno, desde una aplicación JEE a un applet (WALLS, 2011).

Spring proporciona:

- Una potente gestión de configuración basada en JavaBeans, aplicando los principios de Inversión de Control (IoC). Esto hace que la configuración de aplicaciones sea rápida y sencilla. Ya no es necesario tener singletons ni ficheros de configuración, una aproximación consistente y elegante. Estas definiciones de beans se realizan en lo que se llama el contexto de aplicación.
- Una capa genérica de abstracción para la gestión de transacciones, permitiendo gestores de transacción añadibles (pluggables), y haciendo sencilla la demarcación de transacciones sin tratarlas a bajo nivel. Se incluyen estrategias genéricas para JTA y un único JDBC DataSource. En contraste con el JTA simple o EJB CMT, el soporte de transacciones de Spring no está atado a entornos JEE.
- Una capa de abstracción JDBC que ofrece una significativa jerarquía de excepciones (evitando la necesidad de obtener de SQLException los códigos que cada gestor de base de datos asigna a los errores), simplifica el manejo de errores, y reduce considerablemente la cantidad de código necesario.

- Integración con Hibernate, JDO e iBatis SQL Maps en términos de soporte a implementaciones DAO y estrategias con transacciones. Especial soporte a Hibernate añadiendo convenientes características de IoC, y solucionando muchos de los comunes problemas de integración de Hibernate. Todo ello cumpliendo con las transacciones genéricas de Spring y la jerarquía de excepciones DAO.
- Funcionalidad AOP, totalmente integrada en la gestión de configuración de Spring. Se puede aplicar AOP a cualquier objeto gestionado por Spring, añadiendo aspectos como gestión de transacciones declarativa. Con Spring se puede tener gestión de transacciones declarativa sin EJB, incluso sin JTA, si se utiliza una única base de datos en un contenedor Web sin soporte JTA.
- Un framework MVC (Model-View-Controller), construido sobre el núcleo de Spring. Este framework es altamente configurable vía interfaces y permite el uso de múltiples tecnologías para la capa vista como pueden ser JSP, Velocity, Tiles, iText o POI. De cualquier manera una capa modelo realizada con Spring puede ser fácilmente utilizada con una capa web basada en cualquier otro framework MVC, como Struts, WebWork o Tapestry.

Toda esta funcionalidad puede usarse en cualquier servidor JEE, y la mayoría de ella ni siquiera requiere su uso. El objetivo central de Spring es permitir que objetos de negocio y de acceso a datos sean reutilizables, no atados a servicios JEE específicos. Estos objetos pueden ser reutilizados tanto en entornos JEE (Web o EJB), aplicaciones “standalone”, entornos de pruebas, etc., sin ningún problema (WALLS, 2011).

ARQUITECTURA DE SPRING

La arquitectura en capas de Spring ofrece mucha flexibilidad. Toda la funcionalidad está construida sobre los niveles inferiores. Por ejemplo se puede utilizar la gestión de configuración basada en JavaBeans sin utilizar el framework MVC o el soporte AOP.

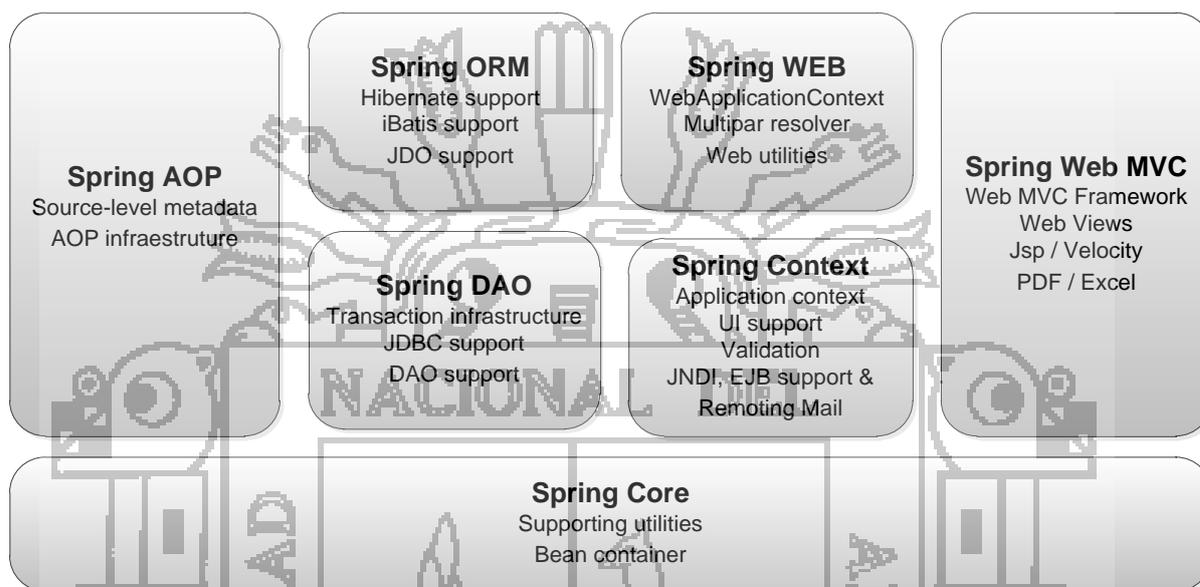


Figura N° 3: Arquitectura Spring

Fuente: (WALLS, 2011)

2.2.4.1. Spring Security.

Spring Security es un subproyecto del framework Spring, que permite gestionar completamente la seguridad de nuestras aplicaciones Java, y cuyas ventajas principales son las siguientes:

- Es capaz de gestionar seguridad en varios niveles: URLs que se solicitan al servidor, acceso a métodos y clases Java, y acceso a instancias concretas de las clases.

- Permite separar la lógica de nuestras aplicaciones del control de la seguridad, utilizando filtros para las peticiones al servidor de aplicaciones o aspectos para la seguridad en clases y métodos.
- La configuración de la seguridad es portable de un servidor a otro, ya que se encuentra dentro del WAR⁴ o el EAR⁵ de nuestras aplicaciones.
- Soporta muchos modelos de identificación de los usuarios (HTTP BASIC, HTTP Digest, basada en formulario, LDAP, OpenID, JAAS y muchos más). Además podemos ampliar estos mecanismos implementando nuestras propias clases que extiendan el modelo de Spring Security (WALLS, 2011).

2.2.5. JAVA SERVER FACES.

La tecnología JSF constituye un marco de trabajo (framework) de interfaces de usuario del lado de servidor para aplicaciones web basadas en tecnología Java y en el patrón MVC (Modelo Vista Controlador).

COMPONENTES:

- Una API y una implementación de referencia.
- Representar componentes de interfaz de usuario (UI-User Interface) y manejar su estado.
- Manejar eventos, validar en el lado del servidor y convertir datos.
- Definir la navegación entre páginas.
- Soportar internacionalización y accesibilidad.

⁴ De sus Siglas en Inglés de Web-Archives

⁵ De sus Siglas en Inglés de Enterprise-Archives

- Proporcionar extensibilidad para todas estas características.
- Una librería de etiquetas JSP personalizadas para dibujar Tutorial de JSF.
- Componentes UI dentro de una página JSP.

BENEFICIOS DE LA TECNOLOGÍA JSF

Una de las grandes ventajas de la tecnología JSF es que ofrece una clara separación entre el comportamiento y la presentación. Las aplicaciones Web construidas con tecnología JSP conseguían parcialmente esta separación. Sin embargo, una aplicación JSP no puede mapear peticiones HTTP al manejo de eventos específicos de los componentes o manejar elementos UI como objetos con estado en el servidor. La tecnología JSF nos permite construir aplicaciones Web que implementan una separación entre el comportamiento y la presentación tradicionalmente ofrecida por arquitectura UI del lado del cliente.

La separación de la lógica de la presentación también le permite a cada miembro del equipo de desarrollo de una aplicación Web enfocarse en su parte del proceso de desarrollo, y proporciona un sencillo modelo de programación para enlazar todas las piezas. Por ejemplo, los Autores de páginas sin experiencia en programación pueden usar las etiquetas de componentes UI de la tecnología JSF para enlazar código de la aplicación desde dentro de la página Web sin escribir ningún script.

Pero lo más importante, la tecnología JSF proporciona una rica arquitectura para manejar el estado de los componentes, procesar los datos, validar la entrada del usuario, y manejar eventos.

2.2.6. PATRONES.

Los patrones de diseño son la base para la búsqueda de soluciones a problemas comunes en el desarrollo de software y otros ámbitos referentes al diseño de interacción o interfaces.

Un patrón de diseño resulta ser una solución a un problema de diseño. Para que una solución sea considerada un patrón debe poseer ciertas características. Una de ellas es que debe haber comprobado su efectividad resolviendo problemas similares en ocasiones anteriores. Otra es que debe ser reutilizable, lo que significa que es aplicable a diferentes problemas de diseño en distintas circunstancias.

2.2.6.1. OBJETIVOS DE LOS PATRONES.

Los patrones de diseño pretenden:

- Proporcionar catálogos de elementos reusables en el diseño de sistemas software.
- Evitar la reiteración en la búsqueda de soluciones a problemas ya conocidos y solucionados anteriormente.
- Formalizar un vocabulario común entre diseñadores.
- Estandarizar el modo en que se realiza el diseño.
- Facilitar el aprendizaje de las nuevas generaciones de diseñadores condensando conocimiento ya existente.

Asimismo, no pretenden:

- Imponer ciertas alternativas de diseño frente a otras.
- Eliminar la creatividad inherente al proceso de diseño.

No es obligatorio utilizar los patrones, solo es aconsejable en el caso de tener el mismo problema o similar que soluciona el patrón, siempre teniendo en cuenta que en un caso particular puede no ser aplicable. "Abusar o forzar el uso de los patrones puede ser un error".

2.2.6.2. CATEGORÍAS DE PATRONES.

Según la escala o nivel de abstracción:

- Patrones de arquitectura: Aquellos que expresan un esquema organizativo estructural fundamental para sistemas de software.
- Patrones de diseño: Aquellos que expresan esquemas para definir estructuras de diseño (o sus relaciones) con las que construir sistemas de software.
- Dialectos: Patrones de bajo nivel específicos para un lenguaje de programación o entorno concreto.

Además, también es importante reseñar el concepto de "antipatrón de diseño", que con forma semejante a la de un patrón, intenta prevenir contra errores comunes de diseño en el software. La idea de los antipatrones es dar a conocer los problemas que acarrear ciertos diseños muy frecuentes, para intentar evitar que diferentes sistemas acaben una y otra vez en el mismo callejón sin salida por haber cometido los mismos errores.

Además de los patrones ya vistos actualmente existen otros patrones como el siguiente:

- **Interacción:** Son patrones que nos permiten el diseño de interfaces web.

2.2.7. DATA ACCESS OBJET.

En software de computadores, un Data Access Object (DAO, Objeto de Acceso a Datos) es un componente de software que suministra una interfaz común entre la aplicación y uno o más dispositivos de almacenamiento de datos, tales como una Base de datos o un archivo. El término se aplica frecuentemente al Patrón de diseño Object.

DAO de la capa es una parte esencial de la arquitectura de buena aplicación.

Las aplicaciones de negocios casi siempre necesitan tener acceso a los datos de bases de datos relacionales u objeto y la plataforma Java ofrece muchas técnicas para acceder a estos datos. La técnica más antigua, es usar el JDBC de la API, que proporciona la capacidad de ejecutar consultas SQL contra una base de datos y obtener los resultados, una columna a la vez. A pesar de esta API proporciona todo lo que un desarrollador necesita para acceder a los datos y para conservar el estado de aplicación, es una API para desarrollar engorroso contra lo que hace que un generador de código de especial utilidad.

GENERAR DAOs:

JEE, ofrece un framework de persistencia más reciente en forma de beans de entidad, un subconjunto de la Enterprise JavaBean (EJB) marco. Aunque ha habido muchas mejoras en la más reciente especificación EJB 2.0, muchos desarrolladores están buscando ahora a los marcos de persistencia alternativas para sus objetos de acceso a datos, tales como Spring DAO y DAO Hibernate.

LOS BENEFICIOS DE DAO:

DAO patrón de diseño proporciona una técnica para la separación de la persistencia de objetos y la lógica de acceso a datos de cualquier mecanismo de persistencia en particular o de la API. Hay claros beneficios de este enfoque desde una perspectiva arquitectónica. El método de acceso a datos de objetos proporciona flexibilidad para cambiar de un mecanismo de aplicación de la persistencia en el tiempo sin la necesidad de la lógica de aplicación re-ingeniería que interactúa con el objeto de acceso a nivel de datos. Por ejemplo, puede haber beneficios en el rendimiento en el cambio de un mecanismo de aplicación el rendimiento del uso de beans de entidad al uso de las llamadas directas JDBC de un bean de sesión, o incluso un traslado a un framework de persistencia, alternativas, como el Spring Framework. Sin un acceso a objetos de datos de nivel en su lugar, este tipo de transición requeriría una amplia re-ingeniería de código existente.

DAO también proporciona una API simple y consistente para el acceso a datos que no requiere el conocimiento de JDBC, EJB, Hibernate, Spring o interfaces.

Un acceso de datos típica interfaz de objetos se muestra a continuación:

```
1. package com.generic.dao;
2. import java.util.List;
3. public interface AbstractDao <DomainObject, KeyType> {
4.     public DomainObject load(KeyType id);
5.     public void update(DomainObject object);
6.     public void save(DomainObject object);
7.     public void delete(DomainObject object);
8.     public void deleteById(KeyType id);
9.     public List<DomainObject> getList();
10.    public void deleteAll();
11.    public int count();
12. }
```

Figura N° 4: Código DAO

Es importante señalar que DAO no sólo se aplica a las asignaciones simples de un objeto a una tabla relacional, sino que también permite consultas complejas a realizar y permite procedimientos almacenados y vistas de base de datos a ser identificado en las estructuras de datos Java.

2.2.8. BUSINESS OBJECT.

El Business Object representa el cliente de datos. Es el objeto que requiere el acceso a la fuente de datos para obtener y almacenar los datos. Un Business Object puede ser implementado como un bean de sesión, bean de entidad, o algún otro objeto de Java, además de un servlet o ayudante beans que accede a la fuente de datos.

Business Object implementa el modelo de dominio conceptual utilizando un modelo de objetos. Business Objects se separan los datos de negocio y la lógica en una capa independiente de la aplicación. Business Objects suelen representar objetos persistentes y pueden ser transparente persistieron con dominio tienda. El patrón de Business Object separa los datos de negocio de la lógica.

2.2.9. GESTIÓN DE LA INFORMACIÓN.

La Gestión de la información (GI) es la denominación convencional de un conjunto de procesos por los cuales se controla el ciclo de vida de la información, desde su obtención (por creación o captura), hasta su disposición final (su archivo o eliminación). Tales procesos también comprenden la extracción, combinación, depuración y distribución de la información a los interesados. El objetivo de la gestión de la información es garantizar la integridad, disponibilidad y confidencialidad de la información.

La Gestión de Información se define como un término impreciso que sirve para designar un conjunto de actividades orientadas a la generación, coordinación, almacenamiento o conservación, búsqueda y recuperación de la información tanto interna como externa contenida en cualquier soporte (PRYTHERCH, 2000).

La Gestión de Información tiene como objetivo optimizar la utilidad y contribución de los recursos de información con el fin de alcanzar los objetivos de la organización. En este sentido, la práctica de la gestión de información se traduce en la creación de canales y medios para transmitir y acceder a la información, así como, en añadirle valores a ésta (CHOO, 2002).

2.2.10. COOPAIN CABANA.

COOPAIN, Cooperativa Agroindustrial Cabana, ubicado en el distrito de Cabana tierra hermosa que produce la mejor quinua del sur del país, grano de oro, que contribuye a la alimentación del hombre desde tiempos inmemoriales. En la actualidad los descendientes de los antiguos productores, agrupados en comunidades constituimos de manera libre y voluntaria la Asociación Central de Productores Multisectoriales -ASCENPROMUL- hoy organizados como empresa comunal decidieron convertirse en Cooperativa Agroindustrial – COOPAIN Cabana, para producir, quinua pelada de alta calidad y luego promover la diversificación y sostenibilidad de la empresa.

En COOPAIN Cabana, perseguimos la calidad para llegar a la excelencia, buscando siempre justicia social, responsabilidad universal y sobre todo respeto por la naturaleza.

2.2.10.1. VISION

Para el 2015 COOPAIN Cabana es una cooperativa agroindustrial fortalecida y moderna. Genera los mayores beneficios y prosperidad para sus clientes y socios. La mayoría de dirigentes, trabajadores y profesionales que trabajan en la planta industrial son los mejores del distrito. Se ha concluido con la construcción de la planta de procesamiento, con más y mejores maquinarias y equipos modernos. Tiene un mercado regional, nacional e internacional amplio para la comercialización de productos orgánicos certificados diversos. Se ha contribuido en mejorar las condiciones de vida y del medio ambiente del distrito de Cabana.

2.2.10.2. MISIÓN

COOPAIN Cabana tiene como propósito mejorar permanentemente la calidad de la producción y comercialización de los diversos productos andinos orgánicos, contribuyendo al progreso, mejor alimentación del hombre, la calidad de vida de los productores y la generación de fuentes de trabajo para los pobladores de Cabana. Estamos orientados por los valores de la solidaridad entre todos, responsabilidad en nuestro trabajo, el respeto a las personas y el medio ambiente, la puntualidad en nuestros compromisos, la equidad entre los hombres y mujeres y la transparencia de nuestros actos. Los beneficiarios de nuestro trabajo son los socios, trabajadores y dirigentes de la empresa cooperativa; así como los pobladores del distrito de Cabana, la región Puno y nuestro país.

2.2.10.3. ESTRUCTURA ORGANIZACIONAL

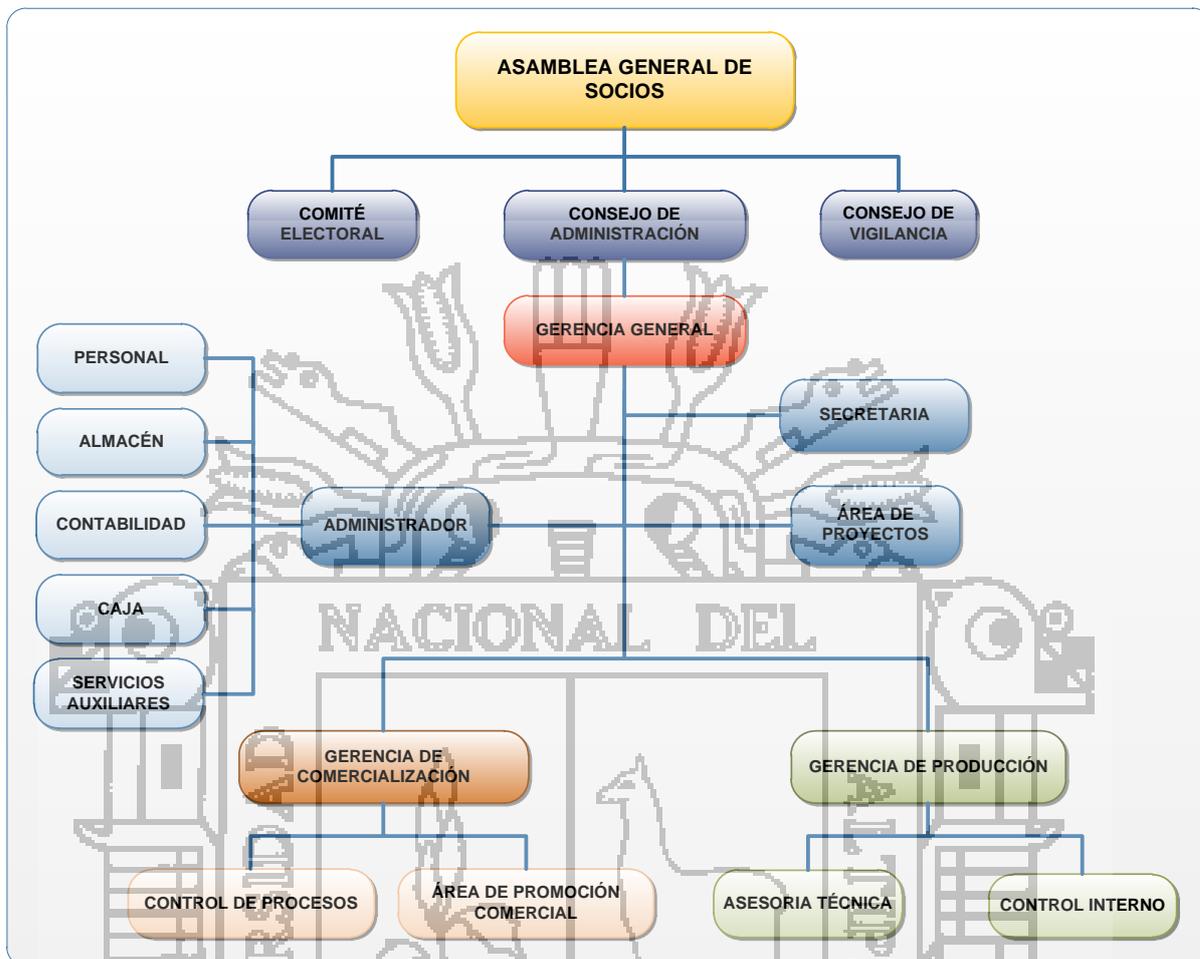


Figura N° 5: Organigrama de la Cooperativa Agroindustrial Cabana – Puno

2.2.11. PROGRAMACIÓN EXTREMA.

A pesar de que los primeros trabajos sobre las ideas y los métodos asociados con la programación extrema (XP) se realizaron a finales de la década de 1980, el trabajo fundamental sobre la materia, escrito por Kent Beck, se publicó en 1999. La programación extrema (XP) utiliza un enfoque orientado a objetos como su paradigma de desarrollo preferido. La programación extrema (XP) abarca un conjunto de reglas y prácticas que ocurren en el contexto de cuatro actividades del marco de trabajo: planeación, diseño, codificación y pruebas (PRESMAN, 2009).

2.2.11.1. PLANEACIÓN

La actividad de planeación comienza creando una serie de historias (también llamadas *historias de usuario*) que describen las características y la funcionalidad requeridas para el software que se construirá. Cada historia (similar a los casos de uso) la escribe el cliente y se coloca en una carta índice. El cliente le asigna un valor (es decir, una prioridad) a la historia basándose en los valores generales del negocio respecto de la característica o la función. Los miembros del equipo de la programación extrema evalúan entonces cada historia. Los clientes y el equipo trabajan juntos para decidir cómo agrupar las historias hacia el próximo lanzamiento (el siguiente incremento de software) para que el equipo las desarrolle (PRESMAN, 2009).

2.2.11.2. DISEÑO

El diseño de la programación extrema sigue de manera rigurosa el principio MS (mantenerlo simple). Siempre se prefiere un diseño simple respecto de una presentación más compleja. Además, el diseño ofrece una guía de implementación para una historia como está escrita, ni más ni menos. Se desapruaba el diseño de funcionalidad extra (porque el desarrollador supone que se requerirá más tarde).

La programación extrema apoya el uso de tarjetas CRC como un mecanismo efectivo para pensar en el software en un contexto orientado a objetos. Las tarjetas CRC (colaborador-responsabilidad-clase) identifican y organizan las clases orientadas al objeto que son relevantes para el incremento del software actual.

La programación extrema apoya la refabricación, una técnica de construcción que también lo es de diseño. Fowler describe la refabricación de la siguiente manera: *Refabricación es el proceso de cambiar un sistema de software de tal manera que no altere el comportamiento externo del código y que mejore la estructura interna. En esencia, se mejora el diseño del código después de que sea escrito.* De hecho, la actividad de construcción misma le proporcionara al equipo una guía sobre cómo mejorar el diseño (PRESMAN, 2009).

2.2.11.3. CODIFICACIÓN

Un concepto clave durante la actividad de codificación es la programación en pareja. La PE recomienda que dos personas trabajen juntas en una estación de trabajo de computadora para crear el código de una historia. Esto proporciona un mecanismo para la resolución de problemas en tiempo real y el aseguramiento de la calidad en las mismas condiciones. También alienta que los desarrolladores se mantengan centrados en el problema que se tiene a la mano (PRESMAN, 2009).

2.2.11.4. PRUEBAS

Las pruebas de aceptación de la PE, también llamadas pruebas del cliente, las especifica el cliente y se enfocan en las características generales y la funcionalidad del sistema, elementos visibles y revisables por el cliente. Las pruebas de aceptación se derivan de las historias del usuario que se han implementado como parte de un lanzamiento de software.

2.2.12. MySQL.

Es un sistema de administración de base de datos relacional (RDBMS) se trata de un programa capaz de almacenar una enorme cantidad de datos de gran variedad y de distribución para cubrir la necesidad de cualquier tipo de organización, desde pequeños establecimientos comerciales a grandes empresas y organismos administrativos, MySQL compite con sistemas RDBMS propietarios conocidos como Oracle, SQL Server y DB2.

MySQL, es un desarrollo de código abierto, lo que significa que el código fuente está disponible gratuitamente para todo el mundo que lo quiera. Como el código fuente de MySQL está disponible, hay formas disponibles de instalar MySQL.

2.2.13. AJAX.

AJAX⁶ es el uso sistemático de JavaScript y XML (y derivados) para hacer el navegador más interactiva con el usuario, usando solicitudes asincrónicas de información. AJAX no es sólo un nuevo modelo, es también una iniciativa para crear aplicaciones web más dinámicas y creativas. AJAX no es una tecnología, son realmente muchas tecnologías trabajando juntos, cada uno haciendo su parte al ofrecer nuevas características. AJAX incorpora en su modelo:

- Presentación basada en estándares usando XHTML y CSS;
- La exposición y la interacción dinámica con el DOM;
- Intercambio y manipulación de datos XML y XSLT;
- Recuperación asíncrona de datos utilizando XMLHttpRequest;
- Javascript unirse todos juntos a ellos.

⁶ Por sus Siglas en Inglés de *Asynchronous Javascript and XML*

El modelo clásico de aplicaciones Web funciona de esta manera: La mayoría de las acciones del usuario interfaz activa una petición HTTP al servidor web. El servidor procesa algo que la recuperación de datos, procesa números, hablando con varios sistemas y devuelve una página HTML para el cliente. Es un modelo adaptado del uso original de la web como un agente de hipertexto, pero lo que hace a la Web buena para el hipertexto no significa necesariamente que sea bueno para las aplicaciones de software.

2.2.14. JASPERS REPORTS.

JasperReports es la mejor herramienta de código libre en Java para generar reportes. Puede entregar ricas presentaciones o diseños en la pantalla, para la impresora o para archivos en formato PDF, HTML, RTF, XLS, CSV y XML.

Está completamente escrita en Java y se puede utilizar en una gran variedad de aplicaciones de Java, incluyendo JEE o aplicaciones Web, para generar contenido dinámico.

FUNCIONAMIENTO DE JASPER REPORTS

JasperReports trabaja en forma similar a un compilador y a un intérprete, el usuario diseña el reporte codificándolo en XML de acuerdo a las etiquetas y atributos definidos en un archivo llamado jasperreports.dtd (parte de JasperReports). Usando XML el usuario define completamente el reporte, describiendo donde colocar texto, imágenes, líneas, rectángulos, cómo adquirir los datos, como realizar ciertos cálculos para mostrar totales, etc.

El archivo fuente XML debe ser compilado para obtener un reporte real. La versión compilada del fuente es nombrada "archivo jasper" (este termina con .jasper). Un Archivo jasper es el compilado de un código fuente. Cuando tenemos un archivo jasper, necesitamos otra cosa para producir un reporte: necesitamos datos. Esto no siempre es cierto. En algunos casos querríamos generar un reporte que no mostrara datos dinámicos, solo texto estático por ejemplo, pero esto puede simplificarse a un reporte que tiene solamente un registro vacío. Para proporcionar estos registros al "jasper engine" necesitamos presentarlos usando una interfaz especial específica llamada JRDataSource. Una fuente de datos más un Archivo jasper = un "archivo print". Un "archivo print" puede exportarse en muchos formatos como PDF, HTML, RTF, XML, XLS, CVS, etc. La exportación se puede realizar utilizando clases especiales para implementar exportadores específicos.

2.3. DEFINICIÓN DE TÉRMINOS BÁSICOS.

2.3.1. SISTEMA.

Se trata de la coordinación de diversos subconjuntos con un objetivo común. Estos subconjuntos son los recursos con los que se cuenta: personal, presupuesto, tecnología, locales, información. De la dotación adecuada de estos recursos, su organización y su relación interdependiente, a través de los procesos y tareas efectuados, resultan los servicios y productos que la biblioteca pone a disposición de sus usuarios o clientes.

2.3.2. JSF

Es un framework web que implementa el patrón MVC.

2.3.3. SPRING

Es un framework modular no intrusivo, diseñado como una solución ligera que cubra los requerimientos que son comunes en aplicaciones empresariales.

2.3.4. HIBERNATE

Es una herramienta ORM (Object Relational Mapping), sirve de puente entre un mundo orientado a objetos y registros de una base de datos.

2.3.5. GESTIÓN.

Gestión hace referencia a la acción y a la consecuencia de administrar o gestionar algo. Al respecto, hay que decir que gestionar es llevar a cabo diligencias que hacen posible la realización de una operación o de un anhelo cualquiera. La noción de gestión, por lo tanto, se extiende hacia el conjunto de trámites que se llevan a cabo para resolver un asunto o concretar un proyecto.

2.3.6. COOPAIN CABANA.

Cooperativa Agroindustrial Cabana.

2.3.7. IoC.

Spring se basa en IoC. IoC es lo que nosotros conocemos como "El Principio de Inversión de Dependencia, Inversion of Control" (IoC) consiste en:

- Un Contenedor que maneja objetos por ti.

- El contenedor generalmente controla la creación de estos objetos. Por decirlo de alguna manera, el contenedor hace los “new” de las clases java para que no los realices tú.
- El contenedor resuelve dependencias entre los objetos que contiene.

2.3.8. PATRÓN MVC

El modelo es el objeto que representa los datos del programa. Maneja los datos y controla todas sus transformaciones. El Modelo no tiene conocimiento específico de los Controladores o de las Vistas, ni siquiera contiene referencias a ellos. Es el propio sistema el que tiene encomendada la responsabilidad de mantener enlaces entre el Modelo y sus Vistas, y notificar a las Vistas cuando cambia el Modelo.

La vista es el objeto que maneja la presentación visual de los datos representados por el Modelo. Genera una representación visual del Modelo y muestra los datos al usuario. Interactúa con el Modelo a través de una referencia al propio Modelo.

El controlador es el objeto que proporciona significado a las órdenes del usuario, actuando sobre los datos representados por el Modelo. Cuando se realiza algún cambio, entra en acción, bien sea por cambios en la información del Modelo o por alteraciones de la Vista. Interactúa con el Modelo a través de una referencia al propio Modelo.

CAPÍTULO III.

MATERIALES Y MÉTODOS

3.1. TRABAJO EXPERIMENTAL.

3.1.1. METODOLOGÍA DE LA INVESTIGACIÓN.

Esta tesis, de acuerdo con las características de la hipótesis, los objetivos y la pregunta de investigación, se enmarca dentro del enfoque cuantitativo correlacional es decir, una investigación que pretende analizar la relación entre variables y que se realiza a través de un diseño de investigación y nos permite de manera objetiva, seleccionar y evaluar alternativas de acción al problema planteado.

3.1.2. DISEÑO DE LA INVESTIGACIÓN.

El diseño de la investigación es cuasi experimental porque permite establecer una relación causal entre una o más variables denominada dependiente (Y) y otras variables independiente (X) en una situación estrictamente controlada. Los diseños cuasi-experimentales manipulan por lo menos una variable independiente para observar su efecto y relación con una o más variables dependientes.

3.1.3. POBLACIÓN Y MUESTRA.

3.1.3.1. POBLACIÓN.

La población a investigar para el desarrollo del presente proyecto estuvo conformada por el personal administrativo de COOPAIN Cabana – Puno.

Tabla N° 1: Población del Personal Administrativo

PERSONAL ADMINISTRATIVO	CANT
Gerente General	1
Gerente de Comercialización	1
Planta	2
Acopio	2
Almacén	1
Caja	1
Asistente Administrativo	1
Asistente De Gerencia	1
Producciones e Inspecciones	4
Auditor Interno	1
Presidente de Administración	1
Seguridad	1
Control de Calidad	1
Contabilidad	1
Administrador Web	1
Soporte Técnico	1
TOTAL(N)	21

Fuente: Elaboración Propia

3.1.3.2. MUESTRA.

La selección de la muestra no probabilística o dirigida se obtuvo por la técnica de muestreo por conveniencia para el personal administrativo que serán los usuarios directos, siendo el tamaño de muestra (MORALES, 2012):

$n = N-1$, Dónde:

$N= 21$; Tamaño de la Población.

$n= 20$; Tamaño de Muestra.

$e=0.05$ (error de tolerado del 5% y 95% de Nivel de Confianza)

3.2. SISTEMA DE VARIABLES.

3.2.1. DEFINICIÓN DE VARIABLES.

Independiente: Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object.

Dependiente: Gestión de la Información.

3.2.2. OPERACIONALIZACIÓN DE VARIABLES

Tabla N° 2: Operacionalización de Variables

VARIABLE	DIMENSIÓN	INDICADOR	ÍNDICE	ESCALA	INSTRUMENTO
INDEPENDIENTE	Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object	Escalabilidad	Capacidad de adaptarse a las circunstancias cambiantes. Cantidad de registros Cantidad de usuarios Cantidad de módulos.	Si No	Test y/o Pruebas
		Usabilidad	Facilidad de usar las interfaces	Excelente Bueno Regular Malo Muy Malo	
		Confiabilidad	Precisión con la que el sistema proporciona confianza, tolerancia a fallas	Excelente Bueno Regular Malo Muy Malo	
DEPENDIENTE	Gestión de la Información	Eficiencia	Se tiene buen desempeño en sus procesos.	Si No	Encuestas
		Tiempo	Se realizan en tiempos reducidos	Segundos Minutos Horas Días	
		Costos	Se reducen los costos en los procesos.	Excelente Bueno Regular Malo Muy Malo	
		Control	Se tiene un mejor control en el manejo de procesos.	Excelente Bueno Regular Malo Muy Malo	

Fuente: Elaboración Propia

3.3. MATERIAL EXPERIMENTAL.

3.3.1. TÉCNICAS DE RECOLECCIÓN DE DATOS.

La técnica que se utilizó para la demostración de la hipótesis se realizó mediante encuestas y la observación directa.

3.3.2. TRATAMIENTO DE DATOS.

Para el tratamiento de datos se utilizó lo siguiente:

- Tabla de Distribución de Frecuencia.
- Desviación estándar.
- Distribución T de Student.

Para el procesamiento de los datos se utilizara la hoja de Cálculo de Excel.

3.3.3. ANÁLISIS DE DATOS.

Los datos recolectados fueron tabulados en hoja de cálculo electrónico y para su representación se utilizaron gráficos estadísticos de barras y líneas.

3.4. METODOLOGÍA DE DESARROLLO DEL SISTEMA

Se utilizó la metodología ágil de programación extrema por la facilidad en darle mayor importancia al desarrollo de la aplicación web que a la documentación de la misma.

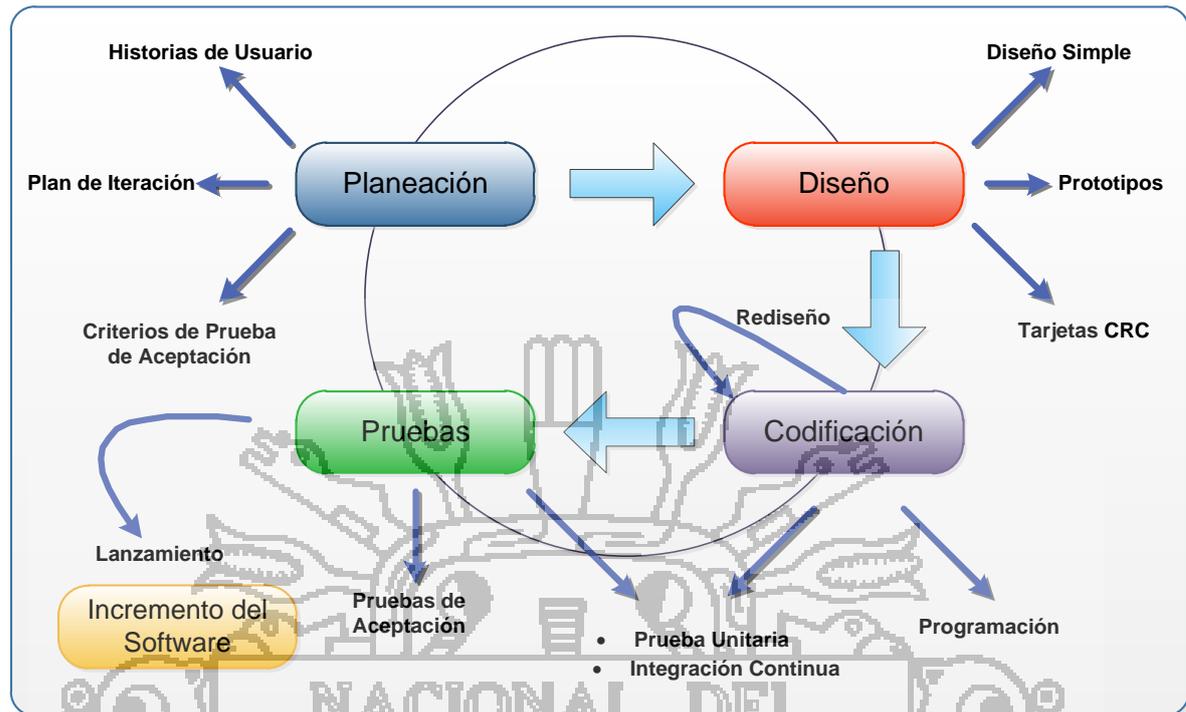


Figura N° 6: Programación Extrema XP

Fuente: (PRESMAN, 2009)

3.4.1. PLANEACIÓN.

A partir de este capítulo se describe la experiencia obtenida en la realización del proyecto. Entre los elementos a discutir se encuentran las historias de usuario, el plan de entregas y lo relacionado con las iteraciones.

3.4.1.1. HISTORIAS DE USUARIO.

En el desarrollo de este proyecto el cliente no fue quien escribió personalmente las historias de usuario, pero fue el quien dirigió la redacción de las mismas. Sin embargo se logró abstraer la información suficiente de ellas para realizar su implementación. Por otro lado es muy importante resaltar el papel fundamental que jugaron las historias de usuario en la estimación de los tiempos requeridos para el desarrollo del proyecto. Una vez recolectadas todas las historias de

usuario, se hizo una reunión del equipo de trabajo donde se plantearon los tiempos necesarios para su implementación, los cuales resultaron en estimaciones inusualmente aproximadas de los tiempos de desarrollo en comparación con los realmente requeridos. Esto es importante resaltarlo debido al poco nivel de detalle que las historias de usuario tenían, significando la poca información sobre las implicaciones técnicas de su implementación.

Finalmente desde el punto de vista del número de historias de usuario, se obtuvo un total de 69. Considerando por un lado la recomendación de que no sean menos de 20 ni más de 80, se deduce que en número es muy adecuado.

3.4.1.2. DESCRIPCIÓN DE HISTORIAS DE USUARIO.

- Historia de Usuario: Módulo Producción
 - Historia de Usuario: Módulo Acopio y Almacén
 - Historia de Usuario: Módulo Procesamiento
 - Historia de Usuario: Módulo Comercialización
 - Historia de Usuario: Módulo Gerencia
 - Historia de Usuario: Módulo Asistencia Personal Administrativo
 - Historia de Usuario: Módulo Control Documentario
 - Historia de Usuario: Módulo Caja
 - Historia de Usuario: Módulo Transporte
 - Historia de Usuario: Módulo Herramientas de Administración del Sistema
- (Anexo A).

3.4.1.3. VELOCIDAD DEL PROYECTO

Si bien esta medida de velocidad del proyecto fue tomada en cuenta para el análisis de tiempos, resultado de mayor utilidad estimar el número de horas que tomaría implementar cada historia de usuario y planificar las entregas acorde con esta medida, De esta forma se pudo estimar cuantas historias de usuario podían ser asignadas en cada iteración.

Tabla N° 3: Velocidad del Proyecto

MODULOS	ID	HISTORIAS DE USUARIO	TIEMPO ESTIMADO(DIAS)	HORAS ESTIMADAS
Producción	PRO	11	23	184
Acopio y Almacén	ACA	9	23	184
Procesamiento	PRC	16	23	184
Comercialización	COM	6	18	144
Gerencia	GER	4	8	64
Personal Administrativo	APA	3	6	48
Control Documentario	CDO	1	5	40
Caja	CAJ	11	19	152
Transporte	TRA	7	15	120
Herramientas de Administración del Sistema	HAS	1	5	40
TOTAL		69	145	1160

Fuente: Elaboración propia

3.4.1.4. DIVISIÓN EN ITERACIONES.

El proyecto fue dividido en 8 iteraciones, por consiguiente se tuvo un total de 8 entregas. La primera iteración se refirió al módulo de PRODUCCION, este orden se eligió debido a la naturaleza del negocio con el cliente y la importancia que se tiene la prestación de servicio a la comunidad, de esta forma se planearon 8

entregas las cuales se realizaron con observaciones del cliente. Una dificultad inesperada que se presentó en la velocidad del proyecto fue el refactoring, ya que en cada iteración surgieron varias recomendaciones por parte del cliente que se convirtieron en refactoring, el cual no se había considerado dentro de ninguna de las medidas de velocidad. Producto de esta omisión el grupo de desarrollo tuvo que reestimar el tiempo dedicado a la iteración afectada para no tener que remover historias de usuario de esta.

Tabla N° 4: División en Iteraciones

MODULOS	ID	TIEMPO ESTIMADO (DIAS)	ITERACIONES
Producción	PRO	23	1
Acopio y Almacén	ACA	33	2
Procesamiento	PRC	24	3
Comercialización	COM	13	4
Control Documentario	CDO	25	5
Caja	CAJ	22	6
Gerencia	GER	16	7
Personal Administrativo	APA	16	8
	TOTAL	172	

Fuente: Elaboración propia

3.4.1.5. PLAN DE ENTREGAS.

XP propone que el cliente sea quien decida cuales historias se implementaran y cuál es el grado de importancia de cada una de la correspondiente iteración, la tarea de escoger las historias fue realizada por el equipo en conjunto, incluyendo al cliente, lo cual no genero problemas en las entregas de los módulos funcionales. La clasificación de las historias no fue realizada estrictamente por su grado de importancia en el proyecto. Solo se optó por desarrollar el módulo de servicio al

cliente en la primera iteración, por tratarse de la actividad más importante en el negocio. En las demás iteraciones se priorizo la dependencia con los módulos ya implementados. Para aproximar el tiempo de desarrollo de cada iteración, se consideró 8 horas de trabajo durante 5 días a la semana.

Aunque las entregas fueron planeadas con fechas para su realización y la mayoría se cumplieron para las fechas indicadas, la reunión para algunas iteraciones no se pudo realizar el día que se tenía planteado, no por razones de retraso en la implementación de la aplicación sino debido a la disponibilidad del cliente, para solucionar el inconveniente se tuvo que buscar un horario acorde del cliente.

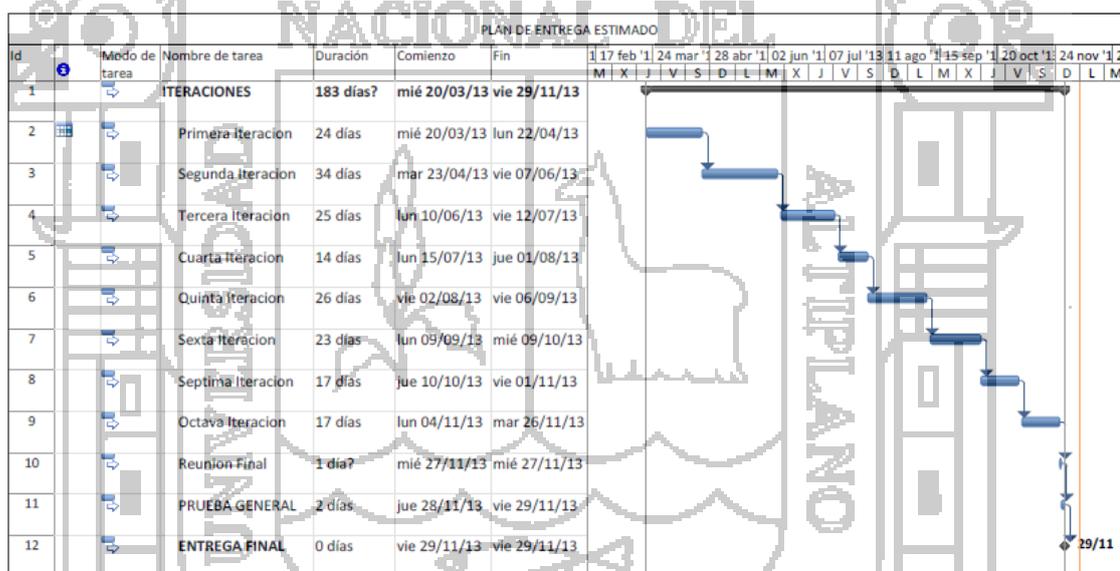


Figura N° 7: Plan de entrega de las iteraciones

Fuente: Elaboración propia

3.4.1.6. REUNIÓN INICIAL DE ITERACIONES.

En la reunión que se realizaba con el equipo de trabajo, se transformaba el contenido de las historias de usuario en responsabilidades que eran plasmadas en las CRC, para luego procede a la asignación de dichas tareas a los programadores, Esta traducción facilito la creación de clases y métodos iniciales de las mismas, ya que fue parte de la etapa de diseño del proyecto. Esta etapa fue realizada en conjunto por ambos miembros del equipo de desarrollo, lo que supone una pequeña modificación a XP, en el sentido que no plantea el diseño en esta reunión.

Las tareas fueron cuidadosamente estimadas en horas, lo cual apporto mayor precisión al momento de calcular las historias a implementar. En dichas estimaciones no se tomó en cuenta el tiempo que se necesita para el refactoring, lo que se considera una omisión, sin embargo cuando se requirió, se les pudo hacer gestión sin afectar al proyecto.

3.4.1.6.1. Primera Iteración: Módulo Producción

Tabla N° 5: Planificación de Tareas: Módulo Producción

N°	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
1	PRO01: Gestionar datos de Asociación y Sector	2	4
2	PRO02: Gestionar datos de Productores, Inscripciones y Generar Reporte de Productores	4	5
3	PRO03: Gestionar datos de Campañas y Parcelas	3	4
4	PRO04: Gestionar datos de Unidad Productiva	2	4
5	PRO05: Gestionar datos de Estimaciones	3	4
6	PRO06: Gestionar datos de Ficha Unidad Productiva	5	4
7	PRO07: Gestionar datos de Inspecciones	4	4
TOTAL		23	29

Fuente: Elaboración Propia.

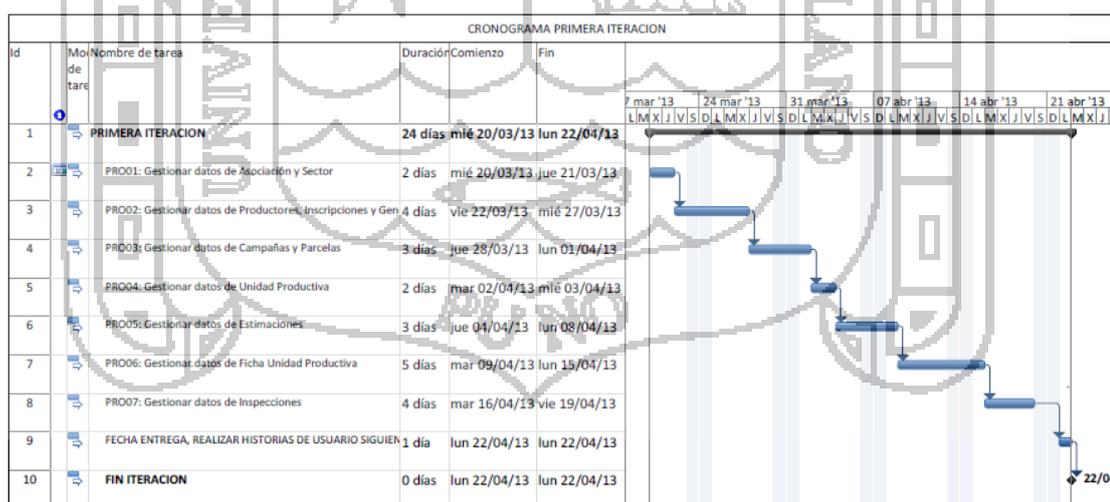


Figura N° 8: Cronograma: Módulo Producción

Fuente: Elaboración Propia.

Historias de Usuario: Producción, (Anexo B)

3.4.1.6.2. Segunda Iteración: Módulo Acopio y Almacén

Tabla N° 6: Planificación de Tareas: Módulo Acopio y Almacén

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
2	ACA01: Gestionar datos de Productor	9	4
2	ACA02: Gestionar datos de Acopio	9	6
2	ACA03: Gestionar datos de Lotes de Compra	9	4
2	ACA04: Generar Reporte de Acopio	3	2
2	ACA05: Generar Reporte de Estimaciones de producción de Productores	3	2
TOTAL		33	18

Fuente: Elaboración propia.

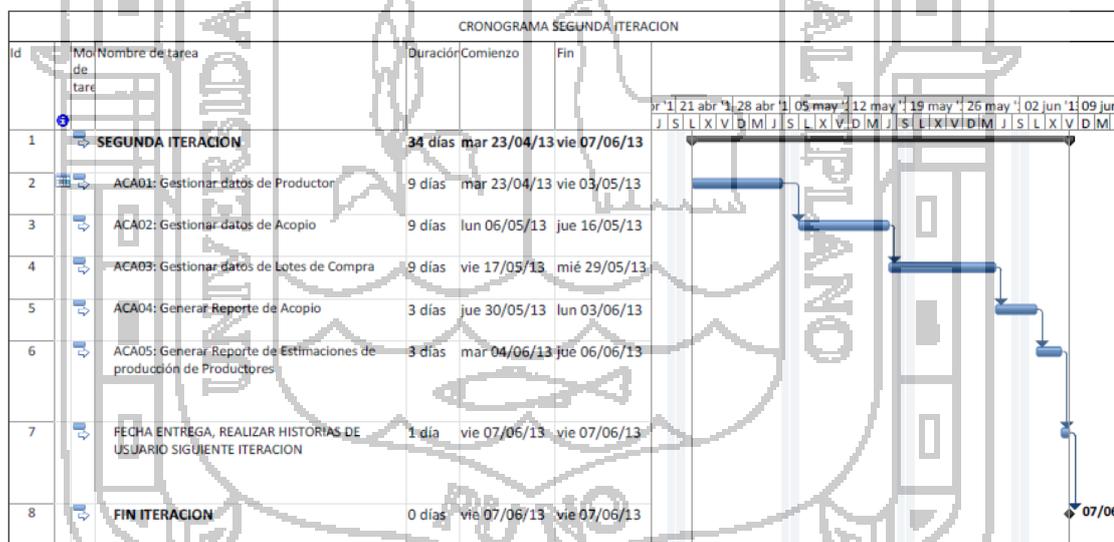


Figura N° 9: Cronograma: Módulo Acopio y Almacén

Fuente: Elaboración propia.

Historias de Usuario: Acopio y Almacén, (Anexo B)

3.4.1.6.3. Tercera Iteración: Módulo Procesamiento

Tabla N° 7: Planificación de Tareas: Módulo Procesamiento

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
3	PRC01: Gestionar datos de Procesamiento	10	14
3	PRC02: Gestionar datos Lotes de Venta	7	8
3	PRC03: Gestionar datos Partes de Salida	3	4
3	PRC04: Gestionar datos de Empresas de Transporte	4	8
TOTAL		24	34

Fuente: Elaboración propia.

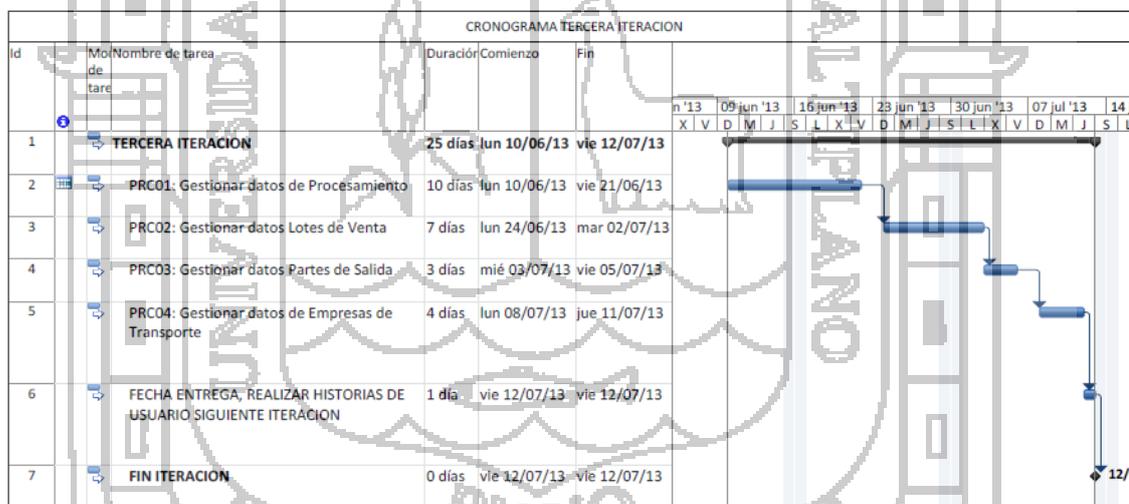


Figura N° 10: Cronograma: Módulo Procesamiento

Fuente: Elaboración propia.

Historias de Usuario: Procesamiento, (Anexo B)

3.4.1.6.4. Cuarta Iteración: Módulo Comercialización

Tabla N° 8: Planificación de Tareas: Módulo Comercialización

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
4	COM01: Gestionar datos de Clientes	3	5
4	COM02: Gestionar datos de Venta	10	13
	TOTAL	13	18

Fuente: Elaboración propia.

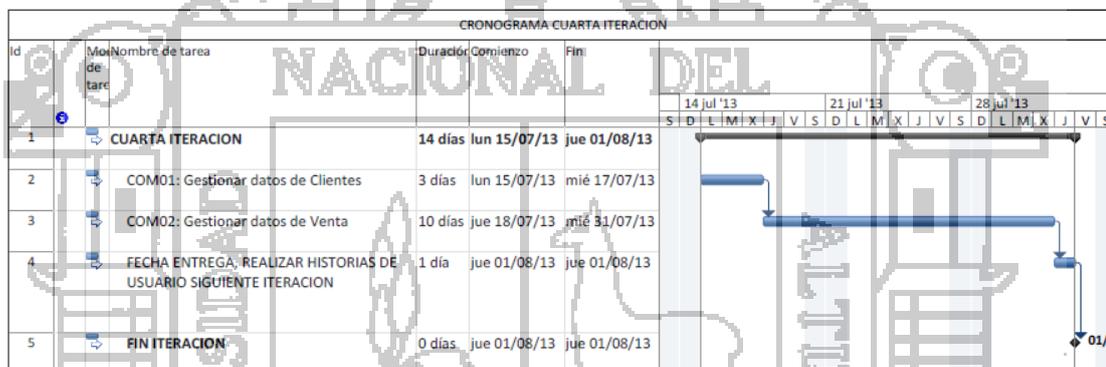


Figura N° 11: Cronograma: Módulo Comercialización

Fuente: Elaboración propia.

Historias de Usuario: Comercialización, (Anexo B)

3.4.1.6.5. Quinta Iteración: Módulo Control Documentario

Tabla N° 9: Planificación de Tareas: Módulo Control Documentario

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
5	CDO01: Gestionar datos del Documento Carta	4	5
5	CDO02: Gestionar datos del Documento Informe	5	5
5	CDO03: Gestionar datos del Documento Memorándums	4	5
5	CDO04: Gestionar datos del Documento Oficio	4	5
5	CDO05: Gestionar datos del Documento Solicitud	4	5
5	CDO06: Gestionar datos del Documento Requerimiento	4	5
TOTAL		25	30

Fuente: Elaboración propia.

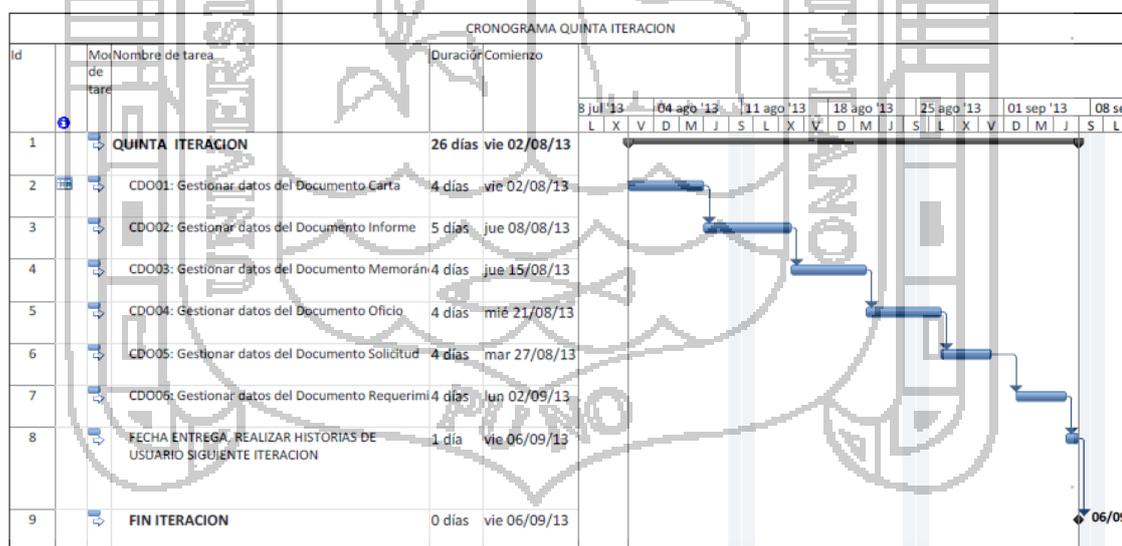


Figura N° 12: Cronograma: Módulo Control Documentario

Fuente: Elaboración propia.

Historias de Usuario: Control Documentario, (Anexo B)

3.4.1.6.6. Sexta Iteración: Módulo Caja

Tabla N° 10: Planificación de Tareas: Módulo Caja

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
6	CJA01: Gestionar datos de Ingreso	4	4
6	CJA02: Gestionar datos de Egreso	4	4
6	CJA03: Gestionar datos de Pago al Personal	4	4
6	CJA04: Gestionar datos de Aporte de Socio.	4	4
6	CJA05: Generar Reporte de Caja	6	2
TOTAL		22	18

Fuente: Elaboración propia.

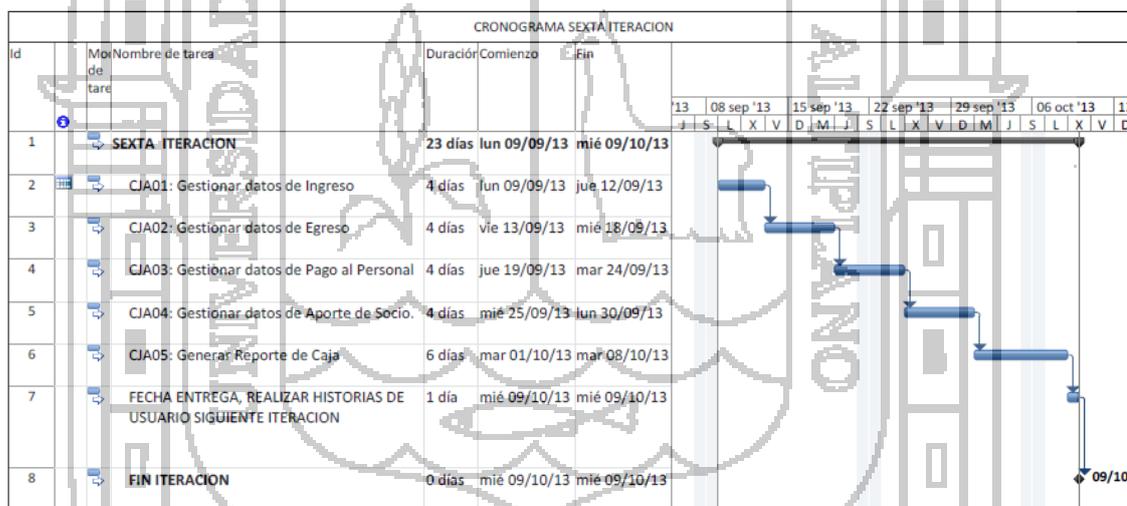


Figura N° 13: Cronograma: Módulo Caja

Fuente: Elaboración propia.

Historias de Usuario: Caja, (Anexo B)

3.4.1.6.7. Séptima Iteración: Módulo Gerencia

Tabla N° 11: Planificación de Tareas: Módulo Gerencia

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
7	GER01: Gestionar datos de Precio de Producción	8	4
7	GER02: Gestionar datos de Precio de Comercialización	8	4
TOTAL		16	8

Fuente: Elaboración propia.

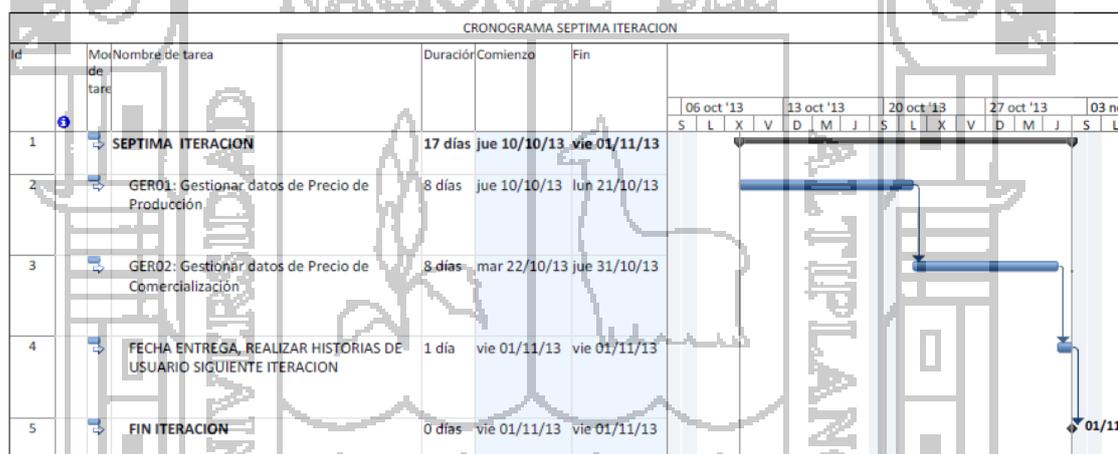


Figura N° 14: Cronograma: Módulo Gerencia

Fuente: Elaboración propia.

Historias de Usuario: Gerencia, (Anexo B)

3.4.1.6.8. Octava Iteración: Módulo Personal Administrativo

Tabla N° 12: Planificación de Tareas: Módulo Personal Administrativo

ITERACION	HISTORIAS REALIZADAS	TIEMPO ESTIMADO (DIAS)	N° TAREAS
8	APA01: Gestionar datos del Personal Administrativo	10	4
8	APA02: Gestionar datos de Precio de Comercialización	3	3
8	APA03: Gestionar acceso al sistema y perfiles de usuario	3	3
TOTAL		16	10

Fuente: Elaboración propia.

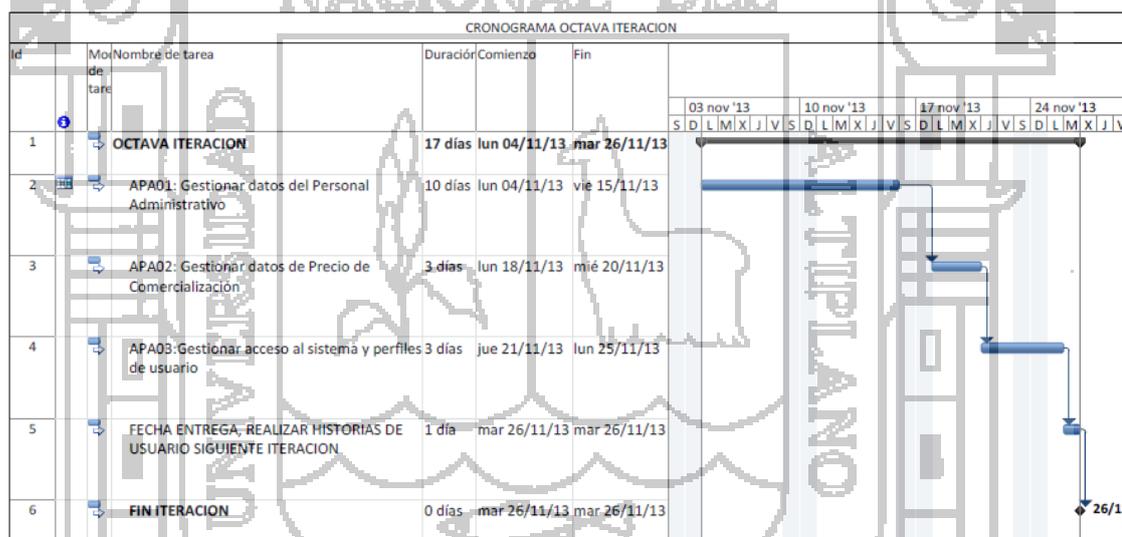


Figura N° 15: Cronograma: Módulo Personal Administrativo

Fuente: Elaboración propia.

Historias de Usuario: Personal Administrativo, (Anexo B)

3.4.1.7. ROLES XP.

Tabla N° 13: Distribución de roles XP

MIEMBRO	GRUPO	ROLES	METODOLOGIA
CHILA CHOQUE, H. FRANK	G-1	MANAGER, PROGRAMADOR.	XP
MAMANI ARENAS, JOEL	G-1	PROGRAMADOR, TESTER	XP
MIRANDA ALEJO, MARIO	CLIENTE	CUSTOMER	XP

Fuente: Elaboración propia.

3.4.1.8. REUNIÓN MATINAL

Este aspecto no fue implementado a cabalidad, la comunicación de problemas y soluciones se realizó a lo largo de la jornada de trabajo. Las discusiones que surgieron no fueron muy largas. Los problemas que se plantearon no demandaron mucho tiempo para encontrar su solución, debido a que eran dudas relacionadas con la codificación, base de datos y diseño.

3.4.1.9. MOVIMIENTO DEL PERSONAL

Debido a que solo se tenían dos desarrolladores, la posibilidad de hacer una rotación de los programadores alrededor del proyecto, en el más estricto sentido de la palabra, era imposible, sin embargo cada decisión de diseño fue tomada por ambos miembros del grupo y los detalles de la implementación fueron compartidos al otro compañero lográndose así el objetivo de evitar las islas de conocimiento. La rotación de programadores no es un fin en XP, es un medio para lograr la propiedad colectiva del código, la que se obtuvo gracias a que solo fueron 2 programadores, que la comunicación fue buena y que se hizo el diseño en conjunto.

3.4.2. DISEÑO.

A diferencia de las metodologías predictivas, el diseño se realizó durante todo el tiempo de vida del proyecto, siendo frecuentemente revisado y algunas veces modificado debido a cambios presentados durante el desarrollo. En este capítulo presentamos una estructura similar a la sección de diseño del marco teórico.

3.4.2.1. SIMPLICIDAD

Desde el punto de vista de las interfaces, no se invirtió mucho tiempo en su diseño, sin embargo se prestó mucha atención a ubicar los elementos tal y como el cliente las habría solicitado y presentándolos en una forma elegante pero sencilla.

A consecuencia de esto se notó una reacción muy positiva del cliente, manifestando conformidad con la apariencia visual de la aplicación. En lo que se refiere a diagramas, se crearon las tarjetas CRC, algunos diagramas de secuencia y el modelo Entidad-Relación, del cual surgieron varias versiones en la medida que se incorporaban funcionalidades a la aplicación. Si bien no fueron muchos diagramas, si fueron muy útiles. Todos estos diagramas fueron elaboradas a mano y sin prestar mucha atención a la estética de los mismos tal y como lo plantea XP.

La única excepción fue el modelo relacional y las tarjetas CRC.

3.4.2.2. DISEÑO DE LA PANTALLA PRINCIPAL

El diseño de la interfaz gráfica de usuario se orientó para que sea atractivo y útil a la mayoría de usuarios. Se determinó un esquema genérico.



Figura N° 16: Esquema de la interfaz del Sistema
Fuente: Elaboración propia.

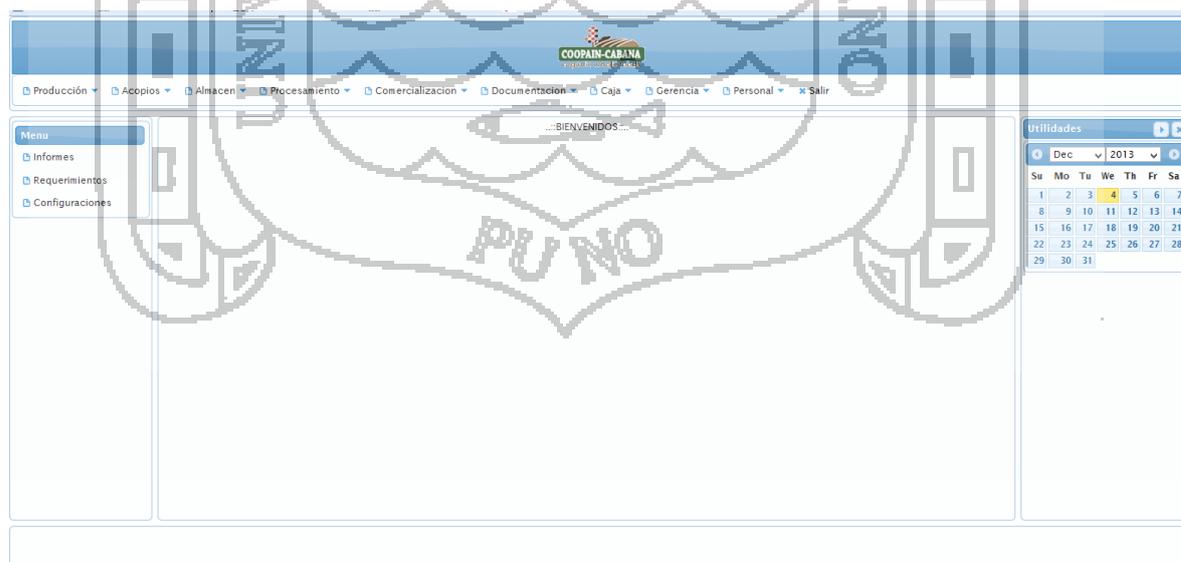


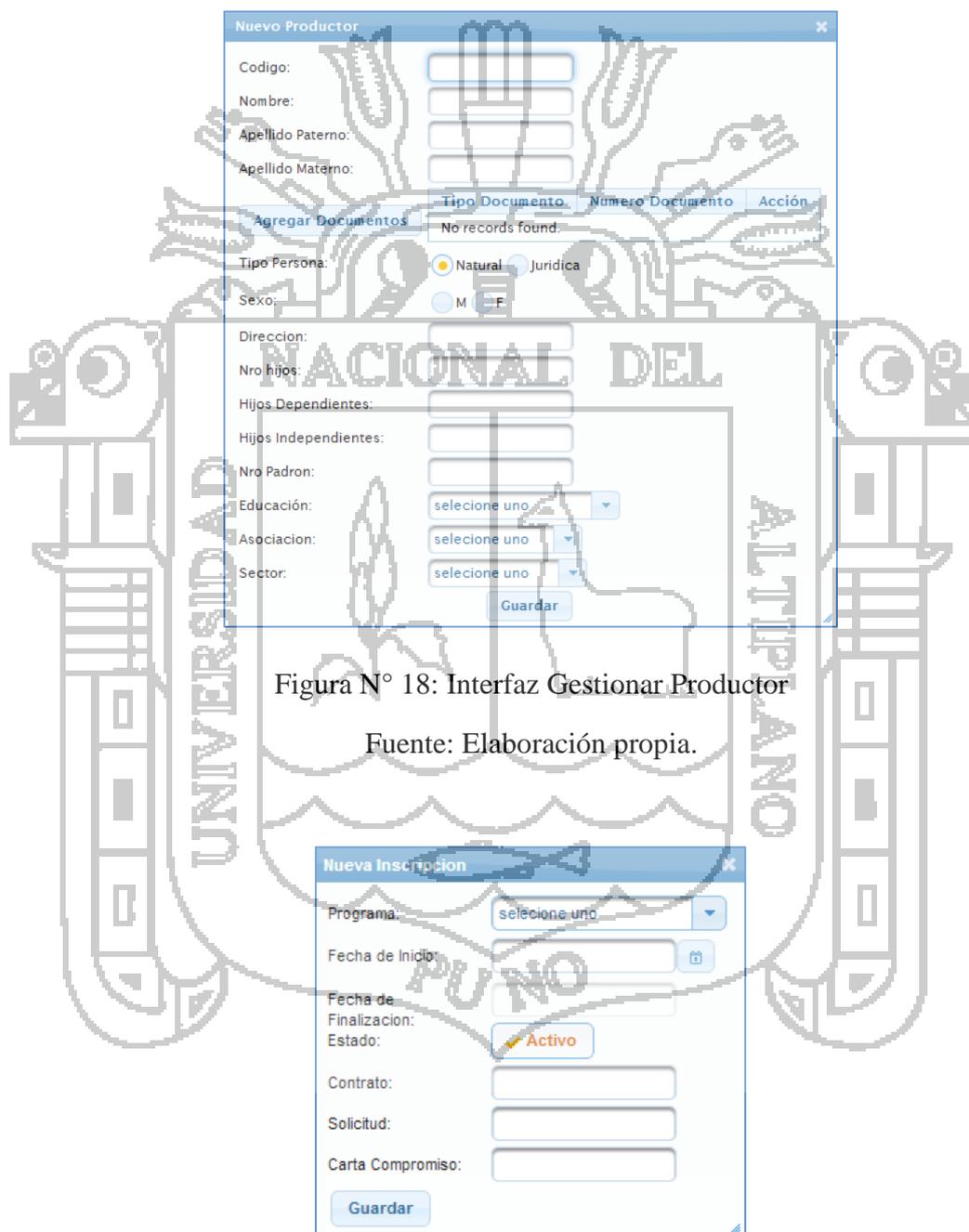
Figura N° 17: Interfaz final del Sistema
Fuente: Elaboración propia.

3.4.2.3. ETAPA DE DISEÑO DE INTERFAZ

3.4.2.3.1. Diseño Interfaz: Módulo Producción

3.4.2.3.1.1. Interfaz Gestionar Datos de Productores, Inscripciones y Generar

Reporte Productores



Nuevo Productor

Codigo:

Nombre:

Apellido Paterno:

Apellido Materno:

Tipo Documento	Numero Documento	Acción
No records found.		

Agregar Documentos

Tipo Persona: Natural Juridica

Sexo: M F

Direccion:

Nro hijos:

Hijos Dependientes:

Hijos Independientes:

Nro Padron:

Educación:

Asociación:

Sector:

Guardar

Nueva Inscripción

Programa:

Fecha de Inicio:

Fecha de Finalización:

Estado:

Contrato:

Solicitud:

Carta Compromiso:

Guardar

Figura N° 18: Interfaz Gestionar Productor

Fuente: Elaboración propia.

Figura N° 19: Interfaz Gestionar Inspecciones

Fuente: Elaboración propia.

Reportes:

Seleccione Tipo de Reporte : Productores Productores por Sector

Sector :

Lista de Productores : Todo los Productores Productores Con Certificacion Organica Productores Sin Certificacion Organica

Seleccione Sexo : Todo Masculino Femenino

Figura N° 20: Interfaz Generar Reporte Productor

Fuente: Elaboración propia.

3.4.2.3.1.2. Interfaz Gestionar Datos de Campañas y Parcelas

Nueva Campaña

Nombre:

Descripción:

Estado:

Parcela

Unidad Productiva:

Código:

Nombre:

Lado:

Altura:

Hectareas:

Ubicacion:

Figura N° 21: Interfaz Gestionar Campañas

Fuente: Elaboración propia.

Figura N° 22: Interfaz Gestionar Campañas

Fuente: Elaboración propia.

3.4.2.3.2. Diseño Interfaz: Módulo Acopio y Almacén

3.4.2.3.2.1. Interfaz Gestionar Datos de Acopio

Figura N° 23: Interfaz Gestionar Acopios

Fuente: Elaboración propia.

3.4.2.3.2.2. Interfaz Gestionar Datos de Lotes de Compra

Figura N° 24: Interfaz Gestionar Lotes de Compra

Fuente: Elaboración propia.

3.4.2.3.2.3. Interfaz Generar Reporte de Acopio

Reportes:

Seleccione Tipo de Reporte : Acopio Acumulado

Seleccione Periodo : Dia Mes Año Intervalo

Seleccione fecha :

Jan 2014

Su	Mo	Tu	We	Th	Fr	Sa
29	30	31	1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30	31	

Fecha Elegida : 2014-01-08

Realizar Reporte

Figura N° 25: Interfaz Generar Reporte Acopio

Fuente: Elaboración propia.

3.4.2.3.3. Diseño Interfaz: Módulo Procesamiento

3.4.2.3.3.1. Interfaz Gestionar Datos de Procesamiento

Registrar Nuevo Procesamiento

Campaña :

Almacen :

Lote de Acopio :

Color de Quinoa :

Lote de Proceso :

Fecha de Proceso :

Numero de Saco	Cantidad	Observaciones	Acción
No records found.			

Cantidad Procesada :

Peso de Lote de Acopio :

Crear

Figura N° 26: Interfaz Gestionar Procesamiento

Fuente: Elaboración propia.

3.4.2.3.3.2. Interfaz Gestionar Datos de Lotes de Venta

Figura N° 27: Interfaz Gestionar Lotes de Venta

Fuente: Elaboración propia.

3.4.2.3.3.3. Interfaz Gestionar Datos de Empresas de Transporte

Figura N° 28: Interfaz Gestionar Empresas de Transporte

Fuente: Elaboración propia.

3.4.2.3.4. Diseño Interfaz: Módulo Comercialización

3.4.2.3.4.1. Interfaz Gestionar Datos de Clientes

Nuevo Cliente

Nombre:

Apellido Paterno:

Apellido Materno:

Tipo Persona: Natural Jurídica

Dirección:

Crear

Figura N° 29: Interfaz Gestionar Clientes

Fuente: Elaboración propia.

3.4.2.3.4.2. Interfaz Gestionar Datos de Ventas

Registrar Nueva Venta

Cliente:

Código de Venta:

Fecha de Envío:

Pesos de Sacos:

	Producto	Cantidad KG	Pre Unitario	Pre Total	Acción
Agregar Productos	Quinoa Blanca Organica Extra	1	15.00	15.00	Quitar

Cantidad Vendida (KG):

Saldo (kg):

Precio Total:

Saldo:

Estado:

Crear

Figura N° 30: Interfaz Gestionar Ventas

Fuente: Elaboración propia.



Agregar Productos

Producto:

Cantidad (KG):

Saldo (KG):

Precio Unitario:

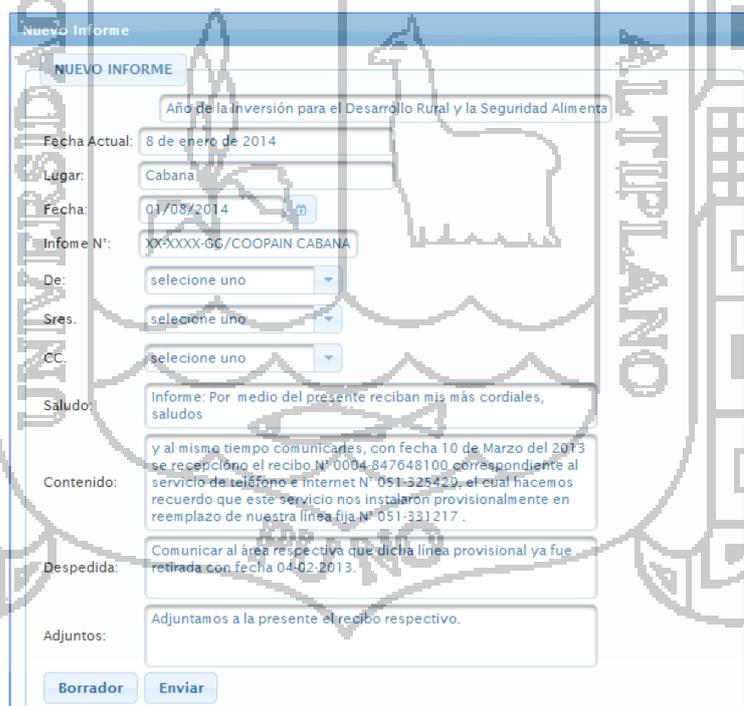
Precio Total:

Figura N° 31: Interfaz Gestionar Productos

Fuente: Elaboración propia.

3.4.2.3.5. Diseño Interfaz: Módulo Control Documentario

3.4.2.3.5.1. Interfaz Gestionar Datos Documento



Nuevo Informe

NUEVO INFORME

Año de la inversión para el Desarrollo Rural y la Seguridad Alimentaria:

Fecha Actual:

Lugar:

Fecha:

Informe N°:

De:

Sres.:

CC.:

Saludo:

Contenido:

Despedida:

Adjuntos:

Figura N° 32: Interfaz Gestionar Documentos

Fuente: Elaboración propia.

3.4.2.3.6. Diseño Interfaz: Módulo Caja

3.4.2.3.6.1. Interfaz Gestionar Datos de Ingresos



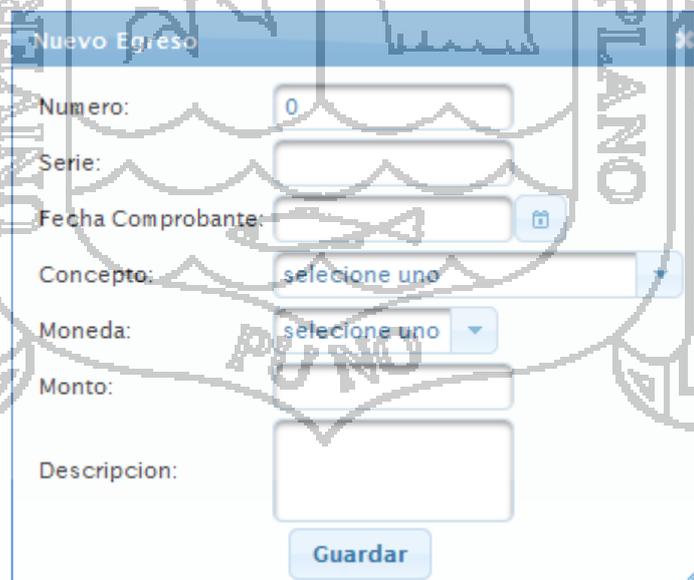
The screenshot shows a web form titled "Nuevo Ingreso" with a close button (X) in the top right corner. The form contains the following fields and controls:

- Numero:
- Serie:
- Fecha Comprobante: with a calendar icon on the right.
- Concepto: with a dropdown arrow on the right.
- Moneda: with a dropdown arrow on the right.
- Monto:
- Descripción:
- A "Guardar" button is located at the bottom center of the form.

Figura N° 33: Interfaz Gestionar Ingresos

Fuente: Elaboración propia.

3.4.2.3.6.2. Interfaz Gestionar Datos de Egresos



The screenshot shows a web form titled "Nuevo Egreso" with a close button (X) in the top right corner. The form contains the following fields and controls:

- Numero:
- Serie:
- Fecha Comprobante: with a calendar icon on the right.
- Concepto: with a dropdown arrow on the right.
- Moneda: with a dropdown arrow on the right.
- Monto:
- Descripción:
- A "Guardar" button is located at the bottom center of the form.

Figura N° 34: Interfaz Gestionar Egresos

Fuente: Elaboración propia.

3.4.2.3.6.3. Interfaz Gestionar Datos de Pagos al Personal

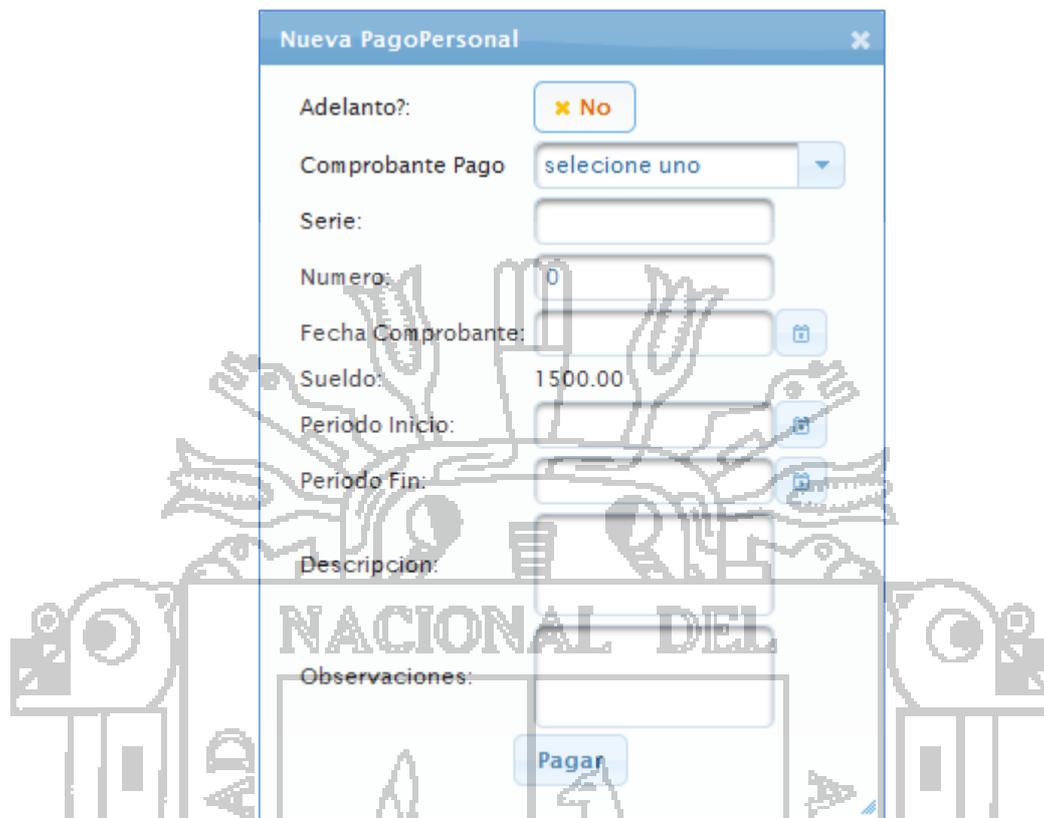


Figura N° 35: Interfaz Gestionar Pagos al Personal

Fuente: Elaboración propia.

3.4.2.3.6.4. Interfaz Generar Reporte de Caja

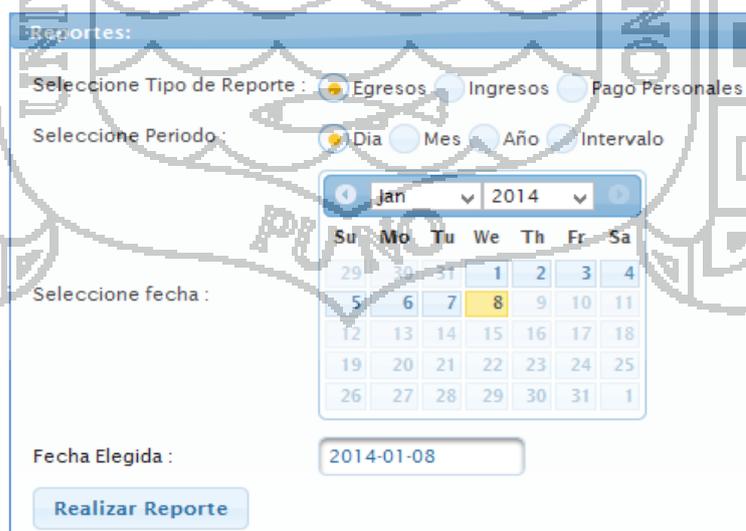


Figura N° 36: Interfaz Generar Reportes de Caja

Fuente: Elaboración propia.

3.4.2.3.7. Diseño Interfaz: Módulo Gerencia

3.4.2.3.7.1. Interfaz Gestionar Datos de Precios de Producción



Figura N° 37: Interfaz Gestionar precios de Producción

Fuente: Elaboración propia.

3.4.2.3.7.2. Interfaz Gestionar Datos de Precios de Comercialización

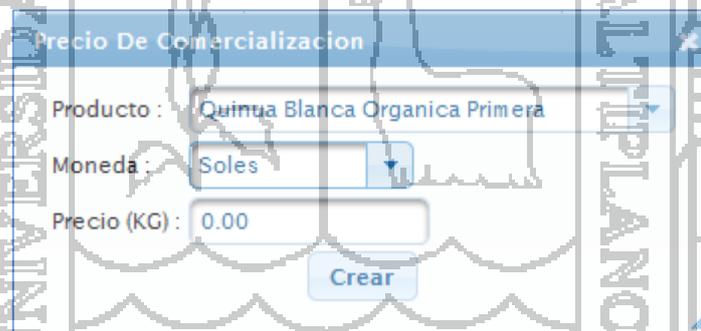
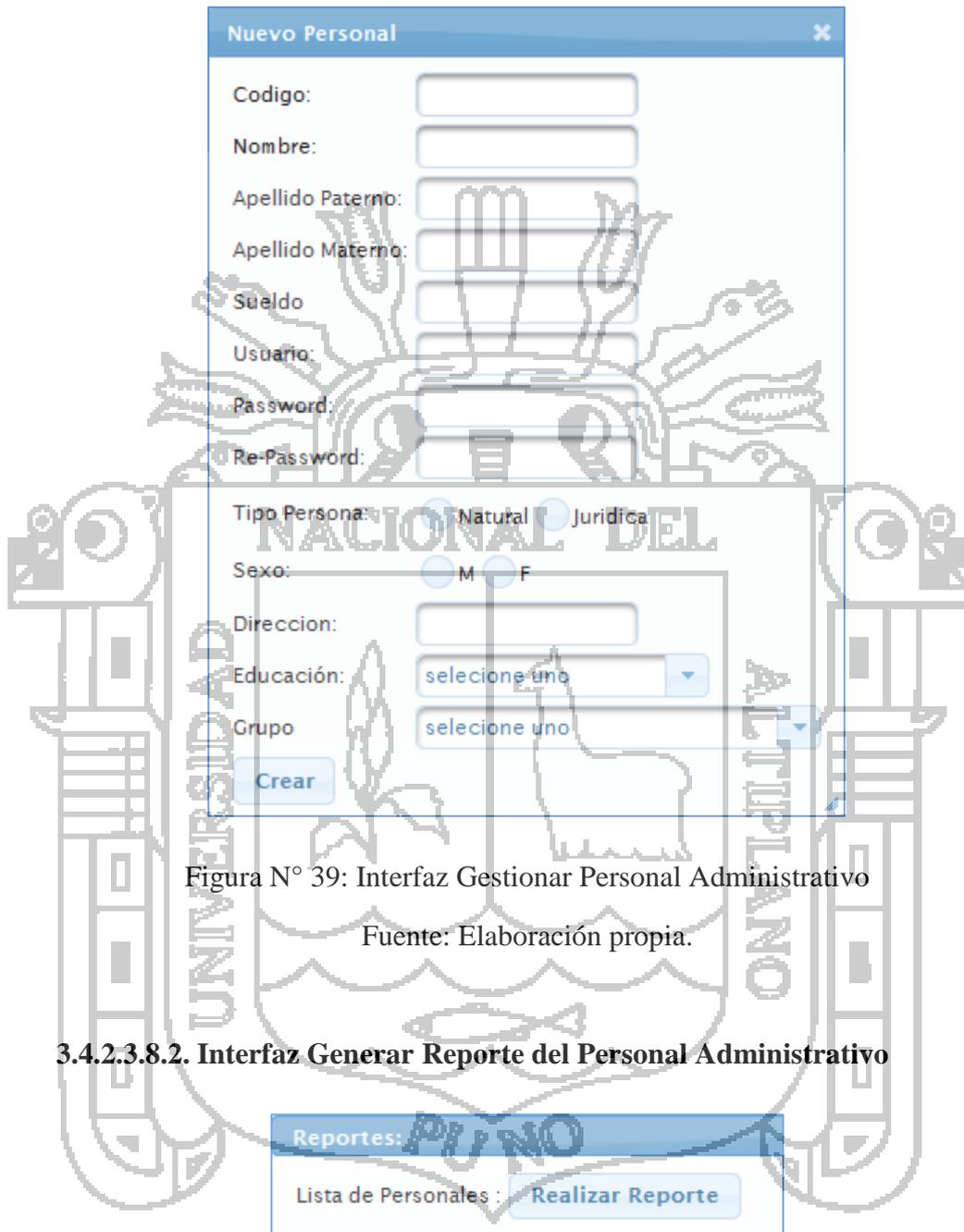


Figura N° 38: Interfaz Gestionar precios de Comercialización

Fuente: Elaboración propia.

3.4.2.3.8. Diseño Interfaz: Módulo Personal Administrativo

3.4.2.3.8.1. Interfaz Gestionar Datos del Personal Administrativo



Nuevo Personal

Codigo:

Nombre:

Apellido Paterno:

Apellido Materno:

Sueldo:

Usuario:

Password:

Re-Password:

Tipo Persona: Natural Juridica

Sexo: M F

Direccion:

Educación:

Grupo:

Reportes:

Lista de Personales:

Figura N° 39: Interfaz Gestionar Personal Administrativo

Fuente: Elaboración propia.

3.4.2.3.8.2. Interfaz Generar Reporte del Personal Administrativo

Figura N° 40: Interfaz Generar Reporte de Personal Administrativo

Fuente: Elaboración propia.

3.4.2.4. DISEÑO DE BASE DE DATOS

3.4.2.4.1. Modelo Relacional de Base de Datos

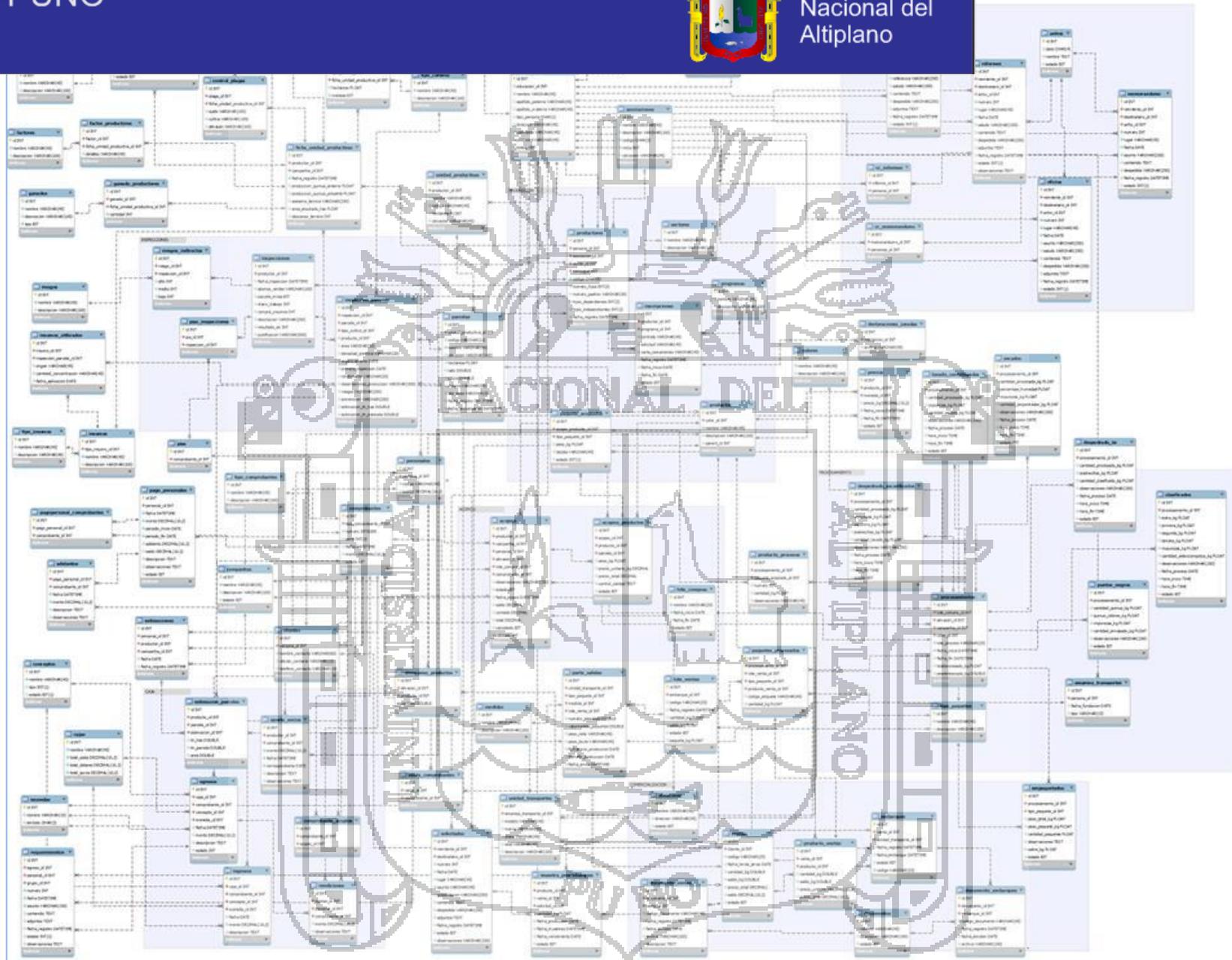


Figura N° 41: Modelo Relacional de la Base de Datos
Fuente: Elaboración propia.

3.4.2.5. TARJETAS CRC

Una de las principales piezas de diseño empleada en el proyecto fueron las tarjetas CRC que no solo sirvieron como columna vertebral de este, sino que también fueron la base del modelo Entidad Relación, elaborada para modelar la base de datos. Cada tarjeta CRC se convirtió en un objeto, sus responsabilidades en métodos y sus colaboradores en llamados a otras clases.

En XP el proceso de diseño es iterativo por lo cual las tarjetas CRC no fueron creadas todas en la primera iteración. Al inicio de cada iteración se les fueron agregando responsabilidades, o fueron creadas CRC nuevas de modo tal que el diseño se convirtió en un proceso dinámico que se adaptaba a las necesidades planteadas para el momento. Sin embargo su utilidad no fue la misma durante todo el proceso de desarrollo. En las primeras iteraciones fueron útiles dando una idea clara de la arquitectura del sistema, distribución de clases, paquetes y la ubicación de las diferentes responsabilidades sobre la lógica de negocio, pero en las últimas iteraciones donde ya se tenía claridad sobre todos estos elementos, las tarjetas CRC fueron menos empleadas. Primero fueron implementadas las clases sencillas, aquellas que no hacían llamados a ninguna otra, para seguir con las que hacían llamados a las ya implementadas y así sucesivamente. Aunque XP no plantea una metodología una metodología debida a que era la forma más cómoda de poder aplicar las pruebas en todo momento.

3.4.2.5.1. Diseño Tarjetas CRC: Módulo Producción



Figura N° 42: Tarjeta CRC Producción

Fuente: Elaboración propia.

3.4.2.5.2. Diseño Tarjetas CRC: Módulo Acopio y Almacén

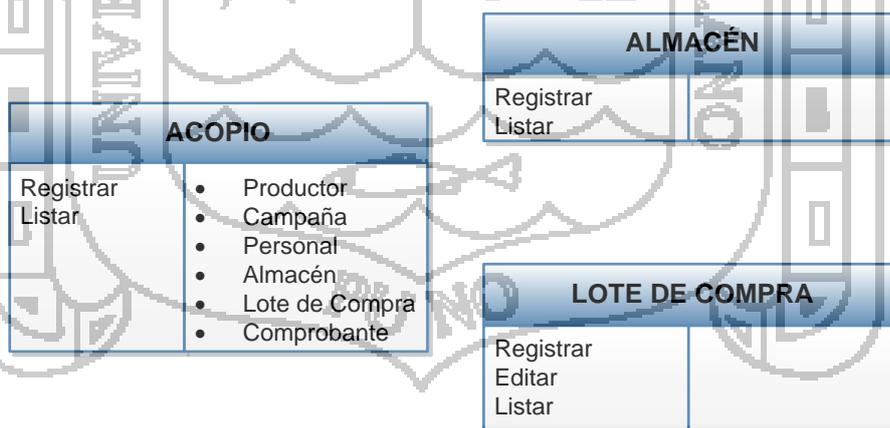


Figura N° 43: Tarjeta CRC Acopio y Almacén

Fuente: Elaboración propia.

3.4.2.5.3. Diseño Tarjetas CRC: Módulo Procesamiento

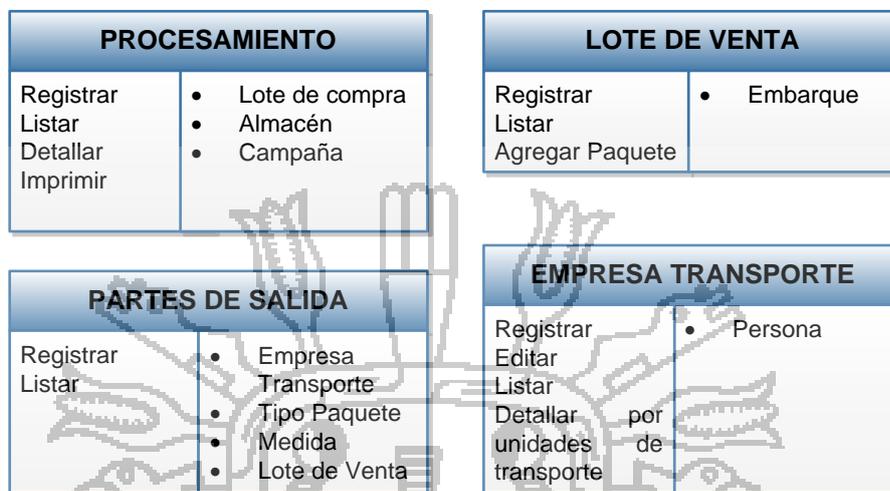


Figura N° 44: Tarjeta CRC Procesamiento

Fuente: Elaboración propia.

3.4.2.5.4. Diseño Tarjetas CRC: Módulo Comercialización y Gerencia



Figura N° 45: Tarjeta CRC Comercialización y Gerencia

Fuente: Elaboración propia.

3.4.2.5.5. Diseño Tarjetas CRC: Módulo Control Documentario

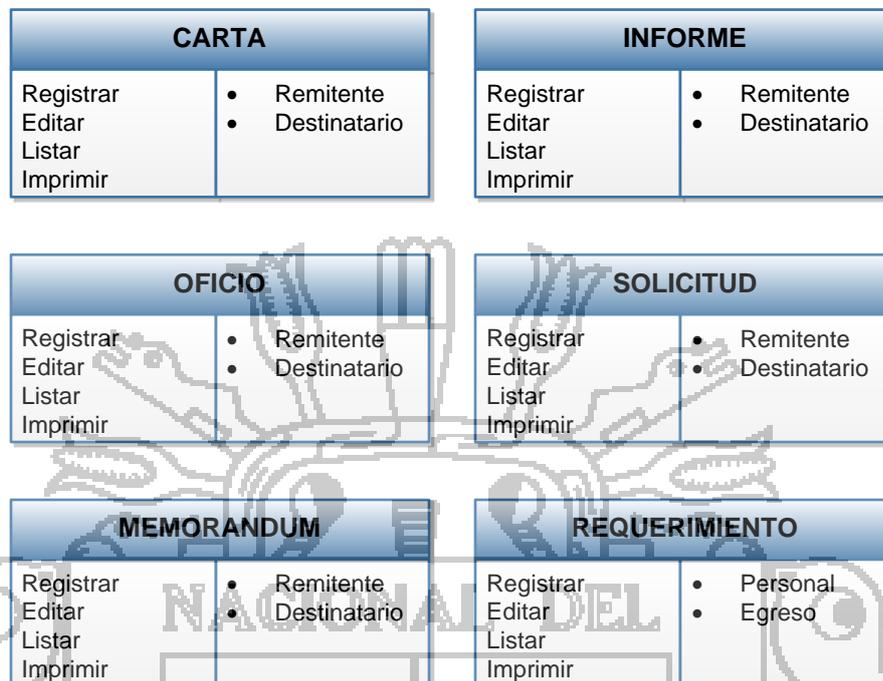


Figura N° 46: Tarjeta CRC Control Documentario

Fuente: Elaboración propia.

3.4.2.5.6. Diseño Tarjetas CRC: Módulo Caja



Figura N° 47: Tarjeta CRC Caja

Fuente: Elaboración propia.

3.4.3. CODIFICACIÓN

La implementación del Sistema, se hizo utilizando toda la potencialidad de plataforma JAVA EE, Tecnologías Open Source y el gestor de Base de datos MySQL.

Se utilizó el patrón MVC para la implementación teniéndose en la vista al framework Java Server Faces por su alta usabilidad en las interfaces para usuario; en el controlador se utilizó el framework Spring por su alta interacción con diferentes frameworks; en el modelo se utilizó el framework Hibernate por que facilita el mapeo de atributos entre una base de datos relacional tradicional y el modelo de objetos de una aplicación, mediante archivos declarativos (XML) o anotaciones en los beans de las entidades que permiten establecer estas relaciones. Para la generación de reportes se utilizó JaspersReports conjuntamente con su herramienta de diseño IReport, dándonos la capacidad de tener un diseño exacto según los requerimientos de COOPAIN Cabana.

3.4.3.1. CLIENTE SIEMPRE ESTÁ PRESENTE

La idea de tener al cliente o un representante no es fácil de asimilar si se consideran los costos que esto representa. Para este proyecto, el cliente tuvo visitas frecuentes dado que debía estar al frente de su negocio. Se implementó la estrategia de comunicación distinta. El equipo de desarrollo podía llamar vía telefónica o usando cualquier aplicación de comunicación de internet en el momento que se tuvo que solucionar cualquier duda en el proceso de implementación, dicha estrategia fue suficiente para lograr una buena comunicación con el cliente. Es importante tener en cuenta que contar con un representante del cliente en las instalaciones del equipo de desarrollo demandara una inversión económica.

3.4.3.2. ESTÁNDARES DE CODIFICACIÓN

Tabla N° 14: Estándares de Codificación

PROPÓSITO	DESCRIPCION
Sentencias de paquete	La primera línea no comentada de un fichero fuente debe ser la sentencia de paquete, que indica el paquete al que pertenece(n) la(s) clase(s) incluida(s) en el fichero fuente. Por ejemplo, package com.coopain.almacen.bo;
Sentencias de importación	Tras la declaración del paquete se incluirán las sentencias de importación de los paquetes necesarios. import com.coopain.generic.dao.AbstractDao; import com.coopain.model.Adelantos;
Declaraciones de clases e interfaces	Métodos - Deben agruparse por funcionalidad en lugar de agruparse por ámbito o accesibilidad. Por ejemplo, un método privado puede estar situado entre dos métodos públicos. El objetivo es desarrollar código fácil de leer y comprender.
Comentarios de implementación	Comentarios de bloque: Permiten la descripción de ficheros, clases, bloques, estructuras de datos y algoritmos./*.....*/ Comentarios de línea: Son comentarios cortos localizados en una sola línea y tabulados al mismo nivel que el código que describen. Si ocupa más de una línea se utilizará un comentario de bloque. Deben estar precedidos por una línea en blanco. //.....
Sentencias	Cada línea debe contener como máximo una sentencia. Ejemplo: int contador++; int variable--;
Paquetes	Se escribirán siempre en letras minúsculas para evitar que entren en conflicto con los nombres de clases o interfaces.
Clases e interfaces	Los nombres de clases deben ser sustantivos y deben tener la primera letra en mayúsculas. Si el nombre es compuesto, cada palabra componente deberá comenzar con mayúsculas. Las interfaces se nombrarán siguiendo los mismos criterios que los indicados para las clases. Ejemplo: public class Anhos; public interface AdelantoDao;
Métodos	Los métodos deben ser verbos escritos en minúsculas. Cuando el método esté compuesto por varias palabras cada una de ellas tendrá la primera letra en mayúsculas (UpperCamelCase).
Valores de retorno	Los valores de retorno tendrán que ser simples y comprensibles, de acuerdo al propósito y comportamiento del objeto en el que se utilicen.
Asignación sobre variables	Se deben evitar las asignaciones de un mismo valor sobre múltiples variables en una misma sentencia, ya que dichas sentencias suelen ser difíciles de leer. int a = b = c = 2; // Evitar
Visibilidad de atributos de instancia y de clase	Los atributos de instancia y de clase serán siempre privados, excepto cuando tengan que ser visibles en subclases heredadas, en tales casos serán declarados como protegidos. El acceso a los atributos de una clase se realizará por medio de los métodos "get" y "set" correspondientes, incluso cuando el acceso a dichos atributos se realice en los métodos miembros de la clase. Ejemplo: public class Unidad { private int id; private String nombre; public void actualizaUnidad(Unidad unidad) { this.setId(unidad.getId()); this.setNombre(unidad.getNombre()); } }

Fuente: Elaboración propia.

3.4.3.3. ILUSTRACIÓN DE PAQUETES UTILIZADOS

3.4.3.3.1. Vistas

En los directorios de vistas se agruparon según la funcionalidad y la modularidad, teniendo en el directorio almacén las interfaces de: almacenes y validar acopios, en el directorio caja las interfaces de aporte socios, egresos, ingresos, pago personal, reportes, etc.; en el directorio template se tiene los diseños de las interfaces las cuales serán aplicadas a cada interfaz del sistema.

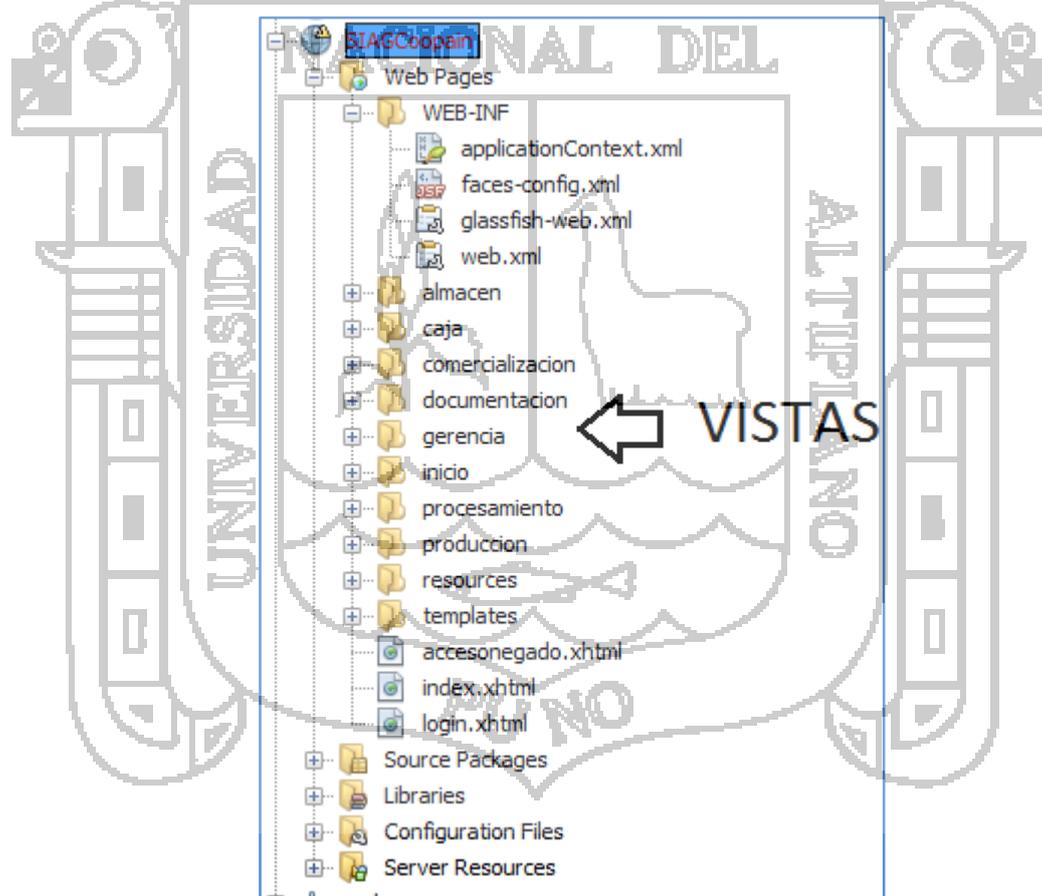


Figura N° 48: Vistas del Sistema

Fuente: Elaboración propia.

3.4.3.3.2. Paquetes

Los paquetes de código se organizaron según la funcionalidad de las clases, los paquetes `com.coopain.almacen.bo`, `com.coopain.caja.bo`, etc. Las clases dentro de los sub-paquetes `bo`, contienen las clases Business Object. Las clases dentro de `bo.impl`, contienen la implementación de las clases Business Object. Las clases dentro de los subpaquetes `dao`, contienen las clases Data Access Object. Las clases dentro de `dao.impl` contienen la implementación de las clases Data Access Object.

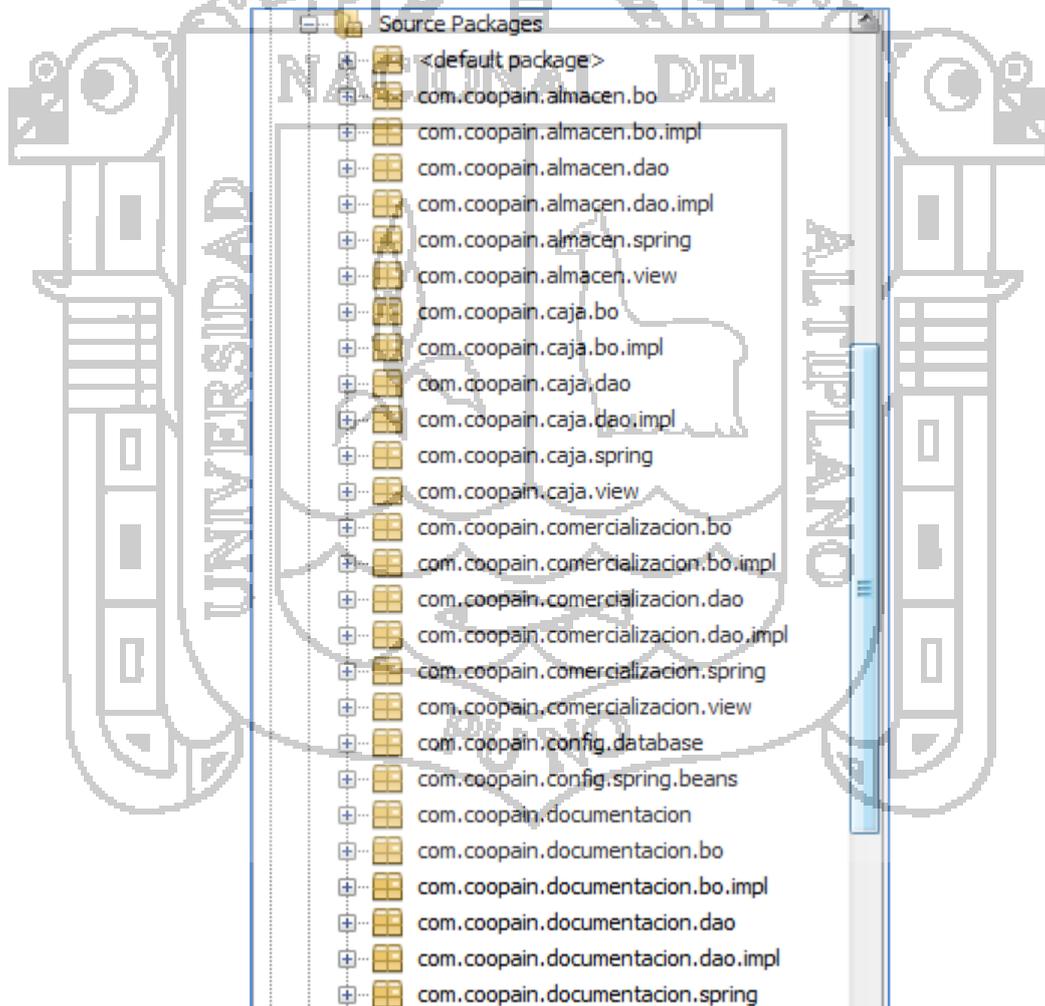


Figura N° 49: Paquetes del Sistema

Fuente: Elaboración propia.

3.4.3.3.3. Archivos de Configuración

- **Archivos faces-config.xml**

Archivo de configuración de JSF, que define las reglas de navegación, incluyendo el ciclo de la interfaz de la aplicación.

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <faces-config version="2.0"
3.     xmlns="http://java.sun.com/xml/ns/javaee"
4.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
5.     xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-facesconfig_2_1.xsd">
6. <application>
7.     <el-resolver>
8.         org.springframework.web.jsf.el.SpringBeanFacesELResolver
9.     </el-resolver>
10. </application>
11. <managed-bean>
12.     <managed-bean-name>procesamiento</managed-bean-name>
13.     <managed-bean-class>com.coopain.procesamiento.view.ProcesamientoBean</managed-bean-class>
14.     <managed-bean-scope>session</managed-bean-scope>
15.     <managed-property>
16.         <property-name>procesamientoBo</property-name>
17.         <value>#{procesamientoBo}</value>
18.     </managed-property>
19. </managed-bean>

```

Figura N° 50: Código de configuración de JSF

- **Archivo applicationContext.xml**

Archivo de configuración que se encarga de la integración de JSF y SPRING.

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans xmlns="http://www.springframework.org/schema/beans"
3.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4.     xsi:schemaLocation="http://www.springframework.org/schema/beans
5.         http://www.springframework.org/schema/beans/spring-beans-3.0.xsd">
6. <!-- Database Configuration -->
7. <import resource="classes/com/coopain/config/spring/beans/DataSource.xml"/>
8. <import resource="classes/com/coopain/config/spring/beans/HibernateSessionFactory.xml"/>
9. <import resource="classes/com/coopain/config/spring/beans/Spring-security.xml"/>
10. <bean
11.     class="com.coopain.generic.dao.AbstractDao" abstract="true" >
12.     <property name="sessionFactory" ref="sessionFactory" />
13. </bean>
14. <import resource="classes/com/coopain/almacen/spring/AlmacenProductoBean.xml"/>
15. <import resource="classes/com/coopain/produccion/spring/AcopioBean.xml"/>
16. <import resource="classes/com/coopain/produccion/spring/AcopioProductoBean.xml"/>
17. <import resource="classes/com/coopain/produccion/spring/AlmacenBean.xml"/>
18. <import resource="classes/com/coopain/produccion/spring/AplicacionInsumoBean.xml"/>
19. <import resource="classes/com/coopain/produccion/spring/AsociacionBean.xml"/>
20. <import resource="classes/com/coopain/produccion/spring/CampanhaBean.xml"/>
21. </beans>

```

Figura N° 51: Código de integración de JSF y Spring

- **Archivo Db.properties**

Archivo de propiedades que de conexión con la base de datos.

```

1. jdbc.driverClassName=com.mysql.jdbc.Driver
2. jdbc.url=jdbc:mysql://localhost:3306/dbcoopain
3. jdbc.username=root
4. jdbc.password=sistemas

```

Figura N° 52: Código de conexión con la Base de Datos

- **Archivo DataSource.xml**

Archivo de configuración, que realiza la conexión con la base de datos utilizando el archivos de propiedades Db.properties.

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans xmlns="http://www.springframework.org/schema/beans">
3. <bean class="org.springframework.beans.factory.config.PropertyPlaceholderConfigurer">
4.     <property name="location">
5.         <value>WEB-INF/classes/com/coopain/config/database/db.properties</value>
6.     </property>
7. </bean>
8. <bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">
9.     <property name="driverClassName" value="{jdbc.driverClassName}" />
10.    <property name="url" value="{jdbc.url}" />
11.    <property name="username" value="{jdbc.username}" />
12.    <property name="password" value="{jdbc.password}" />
13. </bean>
14. </beans>

```

Figura N° 53: Configuración de DataSource

- **Archivo HibernateSessionFactory.xml**

Archivo de configuración que permite la integración de HIBERNATE y SPRING.

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <bean id="sessionFactory" class="org.springframework.orm.hibernate3.annotation.AnnotationSessionFactoryBean">
3.     <property name="dataSource"> <ref bean="dataSource"/> </property>
4.     <property name="hibernateProperties">
5.         <props>
6.             <prop key="hibernate.dialect">org.hibernate.dialect.MySQLDialect</prop>
7.             <prop key="hibernate.show_sql">true</prop>
8.         </props>
9.     </property>
10.    <property name="annotatedClasses">
11.        <list>
12.            <value>com.coopain.model.Acopios</value>
13.            <value>com.coopain.model.Almacenes</value>
14.            <value>com.coopain.model.Asociaciones</value>
15.            <value>com.coopain.model.Cajas</value>
16.        </list>
17.    </property>
18. </bean>

```

Figura N° 54: Configuración integración de Hibernate y Spring

- **Archivo Spring-security.xml**

Archivo que permite la configuración de seguridad de la aplicación.

```

1. <?xml version="1.0" encoding="UTF-8"?>
2. <beans>
3.     <sec:http auto-config="true" access-denied-page="/faces/accesonegado.xhtml" realm="Sistema Integrado de Administracion y
        Gestion COOPAIN CABANA" >
4.
5.         <sec:intercept-url pattern="/login*" access="IS_AUTHENTICATED_ANONYMOUSLY" />
6.         <sec:intercept-url pattern="/faces/produccion/*"
        access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_PRODUCION,ROLE_GERENTEPRODUCCION,ROLE_GERENTEGENERAL" />
7.         <sec:intercept-url pattern="/faces/almacen/*"
        access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_ALMACEN,ROLE_GERENTEPRODUCCION,ROLE_GERENTEGENERAL"
        />
8.         <sec:intercept-url pattern="/faces/procesamiento/*"
        access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_PROCESAMIENTO,ROLE_GERENTEPRODUCCION,ROLE_GERENTEGENERAL" />
9.         <sec:intercept-url pattern="/faces/comercializacion/*"
        access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_ADMINISTRACION,ROLE_GERENTEGENERAL" />
10.        <sec:intercept-url pattern="/faces/documentacion/*"
        access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_ASISTENTEADMINISTRATIVO,ROLE_GERENTEGENERAL" />
11.        <sec:intercept-url pattern="/faces/caja/*" access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_CAJA" />
12.        <sec:intercept-url pattern="/faces/gerencia/*" access="ROLE_SUPERADMIN,ROLE_ADMIN,ROLE_GERENTEGENERAL"
        />
13.        <sec:form-login login-page="/faces/login.xhtml"
14.            login-processing-url="/j_spring_security_check"
15.            default-target-url="/faces/inicio/index.xhtml"
16.            authentication-failure-url="/faces/login.xhtml" />
17.        <sec:logout logout-success-url="/faces/login.xhtml" />
18.    </sec:http>
19.    <sec:authentication-manager>
20.        <sec:authentication-provider>
21.            <sec:jdbc-user-service data-source-ref="dataSource"
22.                users-by-username-query="
23.                    select username,password, estado
24.                    from personas where username=?"
25.                authorities-by-username-query="
26.                    select p.username, g.nombre authority from personas p
27.                    INNER JOIN grupos_personas gp ON gp.persona_id = p.id
28.                    INNER JOIN grupos g ON gp.grupo_id = g.id
29.                    WHERE p.username = ?"
30.            />
31.        </sec:authentication-provider>
32.    </sec:authentication-manager>
33. </beans>

```

Figura N° 55: Configuración de seguridad Spring

3.4.4. PRUEBAS

El equipo de desarrollo realizó pruebas para verificar el correcto funcionamiento del sistema antes de asistir a las reuniones con el cliente en las cuales se realizaron pruebas funcionales. Pero al realizar las reuniones con el cliente, se realizaron una serie de ajustes que permitieron la entrega de los módulos funcionales. Durante el proceso de investigación se realizó la búsqueda de una herramienta capaz de generar las pruebas unitarias y se encontró el framework Junit. Sin embargo la creación de pruebas fue una experiencia nueva para el equipo de trabajo, el aprendizaje de esta herramienta fue traumático y demandó más tiempo de lo planeado, lo cual no fue conveniente ya que la metodología intenta disminuir los cuellos de botella. Sin embargo, se verificó la correcta funcionalidad de cada uno de los módulos a través de pruebas consecutivas con el cliente para confirmar que la historia de usuario ha sido implementada correctamente (ANEXO C). Una vez terminada la implementación del Sistema, se realizó la prueba con el Personal Administrativo de COOPAIN Cabana - Puno. Después de esto, se realizó una encuesta (ANEXO D) general para ver cuál es la opinión que tienen los usuarios del sistema.

3.5. MATERIAL APLICATIVO

Para el desarrollo de la investigación se utilizaron instrumentos de hardware, software y servicios. A continuación se detalla cada uno de ellos:

3.5.1. HARDWARE

- 2 Computadoras Personales.
- Impresora Láser.

- Impresora de Ticket Bixelon Srp-270.
- Memoria USB.
- Discos Compactos.

3.5.2. SOFTWARE

- Microsoft Windows XP © SP3.
- Microsoft Office 2010 ©.
- Microsoft Visio 2010.
- Microsoft Project 2010.
- Adobe Reader X.
- Mozilla Firefox 25.1.
- Adobe PhotoshopCS5.
- Macromedia Flash CS5.
- MySQLWorkbench 5.2 CE.
- NetBeans IDE 7.4
- Glassfish v3
- JAVA
- Framework JSF(Java Server Faces-PrimeFaces)
- Framework Spring
- Framework Hibernate
- MySQL 5.

3.5.3. SERVICIOS

- Conexión a Internet.

CAPÍTULO IV.

RESULTADOS Y DISCUSIÓN

4.1. RESULTADOS DE LAS ENCUESTAS

Los resultados de la mencionada encuesta se basaron en 5 aspectos: Confiabilidad, Facilidad de Uso (Usabilidad), Tiempo, Costo y Control. La evaluación de mejora se ponderó bajo la escala de 5 a 25.

Tabla N° 15: Resultado general de la Encuesta

USUARIO	CONFIABILIDAD	USABILIDAD	TIEMPO	COSTO	CONTROL	APRECIACION DE MEJORA
1	4	5	4	4	5	22
2	5	4	5	5	5	24
3	4	4	5	5	5	23
4	4	5	3	5	5	22
5	5	5	5	5	5	25
6	4	3	5	5	5	22
7	4	4	5	5	5	23
8	5	4	5	5	5	24
9	4	5	4	3	5	21
10	4	5	3	5	4	21
11	4	5	5	5	5	24
12	5	4	4	5	5	23
13	3	5	5	5	5	23
14	4	4	4	5	5	22
15	5	5	5	5	5	25
16	4	5	5	5	5	24
17	4	4	4	5	5	22
18	4	5	5	5	5	24
19	4	5	5	5	5	24
20	4	5	4	5	5	23
					PROMEDIO	23.05
					Desv. Est	1.190974833

Fuente: Elaboración propia.

Leyenda: Excelente: 5, Bueno: 4; Regular: 3; Malo: 2; Deficiente: 1

1. En cuanto a la confiabilidad del Sistema ¿Cuál es su apreciación con el manejo de información en sus procesos?

Tabla N° 16: Opinión respecto a la Confiabilidad

APRECIACION	FRECUENCIA ABSOLUTA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA RELATIVA	FRECUENCIA RELATIVA PORCENTUAL (%)
Excelente	5	5	0.25	25%
Bueno	14	19	0.7	70%
Regular	1	20	0.05	5%
Malo	0	20	0	0%
Deficiente	0	20	0	0%
Total	20		1	100%

Fuente: Anexo D: Encuesta.

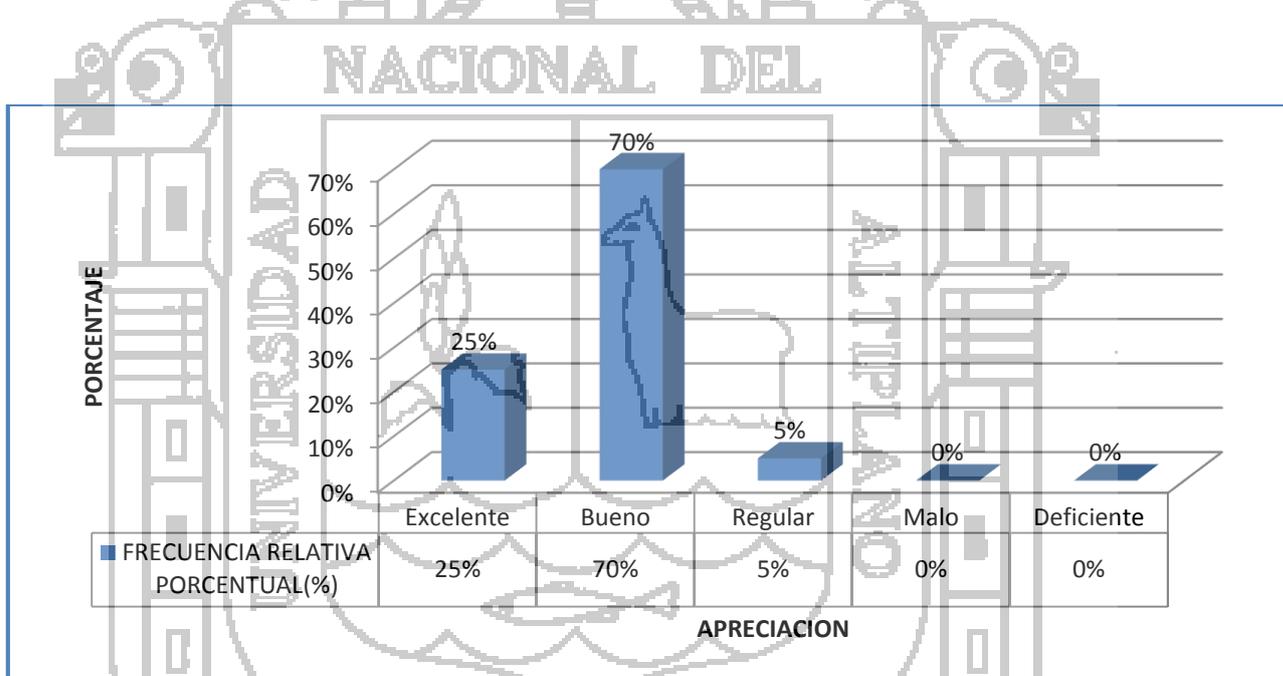


Figura N° 56: Opinión con respecto a la Confiabilidad del Sistema

Fuente: Tabla N° 16.

Interpretación: De los resultados se observa que el 25% de los encuestados les pareció excelente y el 70% les pareció bueno. Con esto se concluye que la Confiabilidad es buena para el Usuario.

2. Respecto a la Facilidad del Uso del Sistema ¿Cuál es su apreciación?

Tabla N° 17: Opinión respecto a la Usabilidad del Sistema

APRECIACION	FRECUENCIA ABSOLUTA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA RELATIVA	FRECUENCIA RELATIVA PORCENTUAL (%)
Excelente	12	12	0.6	60%
Bueno	7	19	0.35	35%
Regular	1	20	0.05	5%
Malo	0	20	0	0%
Deficiente	0	20	0	0%
Total	20		1	100%

Fuente: Anexo D: Encuesta.

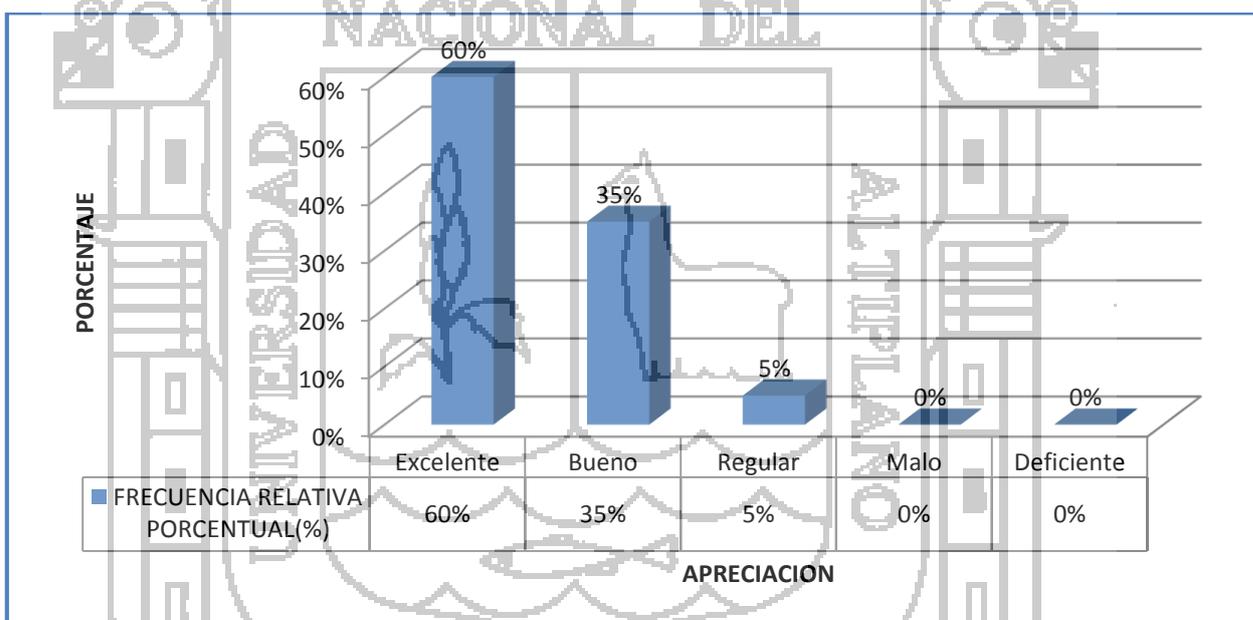


Figura N° 57: Opinión con respecto a la Usabilidad del Sistema

Fuente: Tabla N° 17

Interpretación: De los resultados se observa que el 60% de los encuestados les pareció excelente en cuanto a la usabilidad del Sistema, el 35% les pareció Bueno y el 5% les pareció Regular. Con esto se concluye que el Sistema es fácil de usar.

3. Según Usted, ¿Cómo califica el tiempo para realizar un proceso?

Tabla N° 18: Opinión respecto al tiempo de ejecución de procesos

APRECIACION	DESCRIPCION	FRECUENCIA ABSOLUTA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA RELATIVA	FRECUENCIA RELATIVA PORCENTUAL (%)
Excelente	Menos de 1 minuto	12	12	0.6	60%
Bueno	De 2 a 3 minutos	6	18	0.3	30%
Regular	De 4 a 5 minutos	2	20	0.1	10%
Malo	De 5 a 10 minutos	0	20	0	0%
Deficiente	Más de 10	0	20	0	0%
Total		20		1	100%

Fuente: Anexo D: Encuesta.

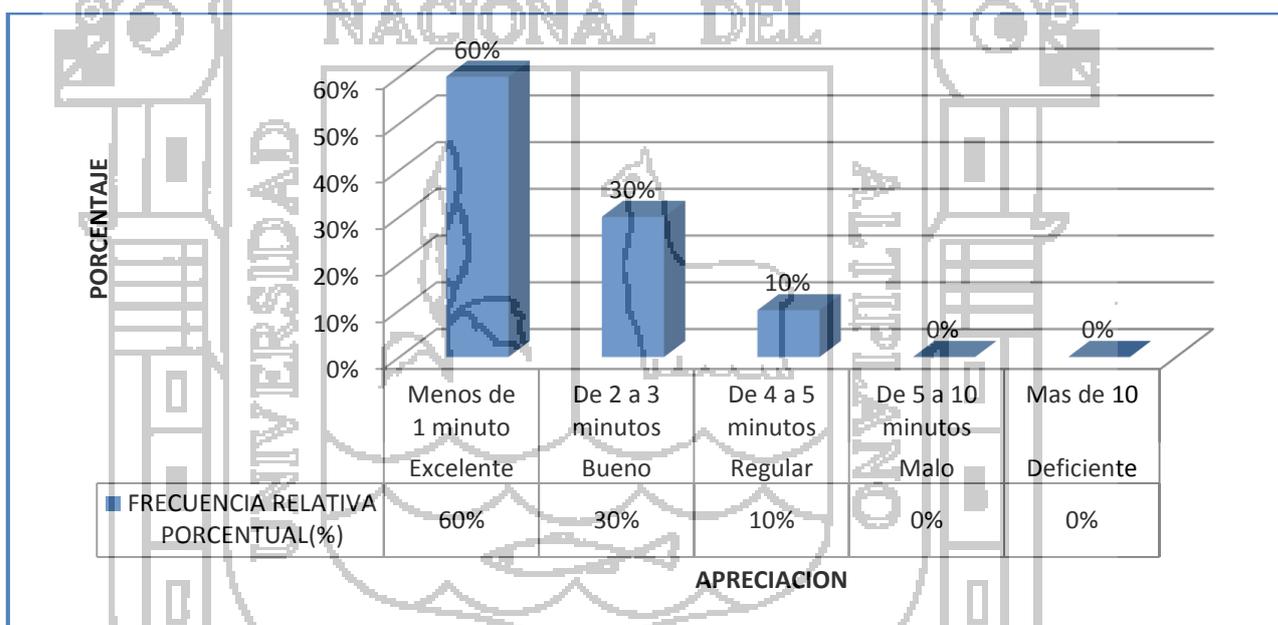


Figura N° 58: Opinión con respecto al tiempo de ejecución de procesos

Fuente: Tabla N° 18.

Interpretación: De los resultados se observa que el 60% de los encuestados concluyeron sus operaciones con el Sistema en menos de 1 minuto, el 30% de 2 a 3 minutos y 10% de 4 a 5 minutos. Con esto se concluye que el Sistema realiza sus procesos en menos tiempo de lo normal.

4. Según usted, como califica, acerca de que los costos de operación son menores con el uso del Sistema.

Tabla N° 19: Opinión respecto a los costos de operación con el Sistema

APRECIACION	FRECUENCIA ABSOLUTA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA RELATIVA	FRECUENCIA RELATIVA PORCENTUAL (%)
Excelente	18	18	0.9	90%
Bueno	1	19	0.05	5%
Regular	1	20	0.05	5%
Malo	0	20	0	0%
Deficiente	0	20	0	0%
Total	20		1	100%

Fuente: Anexo D: Encuesta.

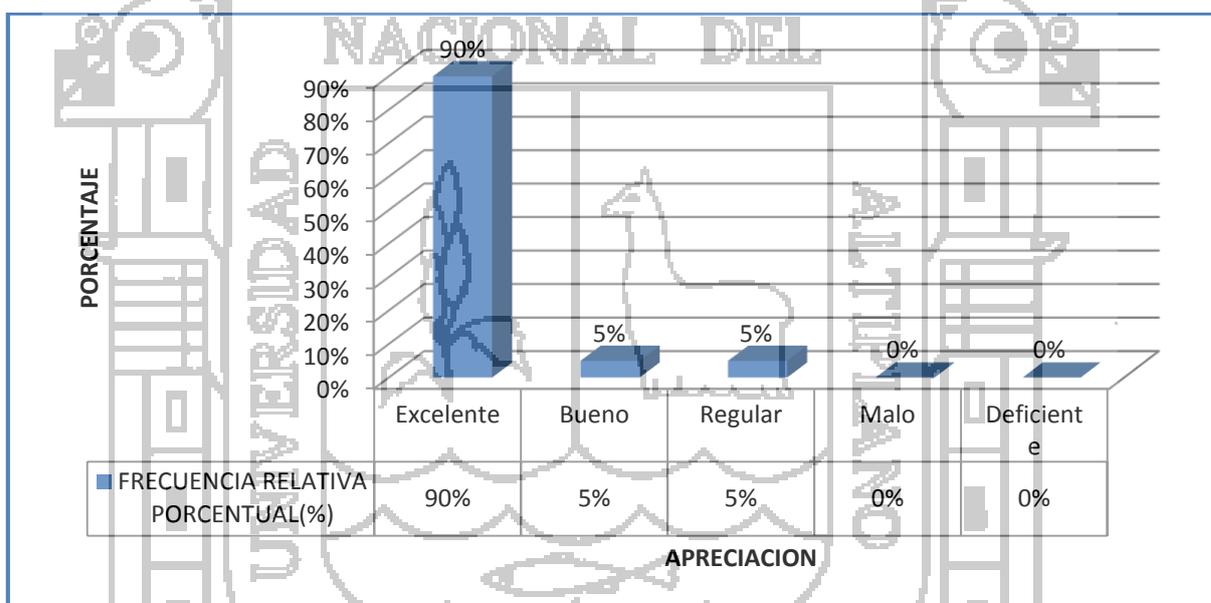


Figura N° 59: Opinión con respecto a los costos de operación con el Sistema

Fuente: Tabla N° 19.

Interpretación: De los resultados se observa que el 90% de los encuestados señalaron que se redujo los costos en las operaciones con el Sistema. Con esto se concluye que el Sistema redujo Costos.

5. Con la Implementación del Sistema, ¿Cuál es la calificación, respecto a un mejor control?

Tabla N° 20: Opinión respecto al control de procesos con el Sistema

APRECIACION	FRECUENCIA ABSOLUTA	FRECUENCIA ABS. ACUMULADA	FRECUENCIA RELATIVA	FRECUENCIA RELATIVA PORCENTUAL (%)
Excelente	19	19	0.95	95%
Bueno	1	20	0.05	5%
Regular	0	20	0	0%
Malo	0	20	0	0%
Deficiente	0	20	0	0%
Total	20		1	100%

Fuente: Anexo D: Encuesta.

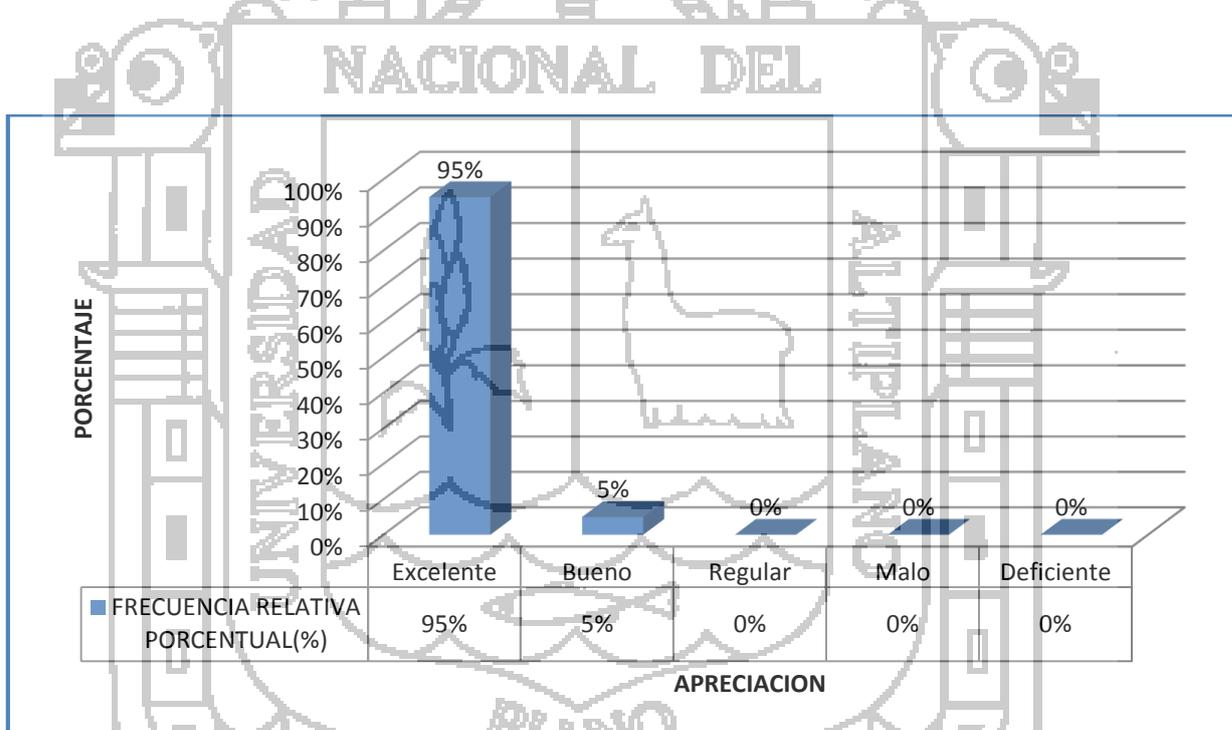


Figura N° 60: Opinión con respecto a control de procesos con el Sistema

Fuente: Tabla N° 20.

Interpretación: De los resultados se observa que el 95% de los encuestados señalaron que se tiene mejor control con el Sistema. Con esto se concluye que el Sistema realiza un mejor control de procesos.

4.2. PRUEBA DE HIPÓTESIS

H0: El Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object no mejora la Gestión de la Información de COOPAIN Cabana – Puno.

H1: El Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object mejora la Gestión de la Información de COOPAIN Cabana – Puno.

Debido a que para la evaluación de mejora se ponderó bajo la escala de 5 a 25, las dos hipótesis a probar fueron:

$$H0: \mu \leq 22$$

$$H1: \mu > 22$$

El estadístico de prueba utilizado para probar la hipótesis fue la distribución t ya que la muestra es menor que 30. El nivel de significancia (α) escogido para la prueba de la hipótesis será del 5%. Siendo $\alpha = 0.05$ (Nivel de Significancia) y $n-1 = 19$ grados de libertad (n : número de muestra).

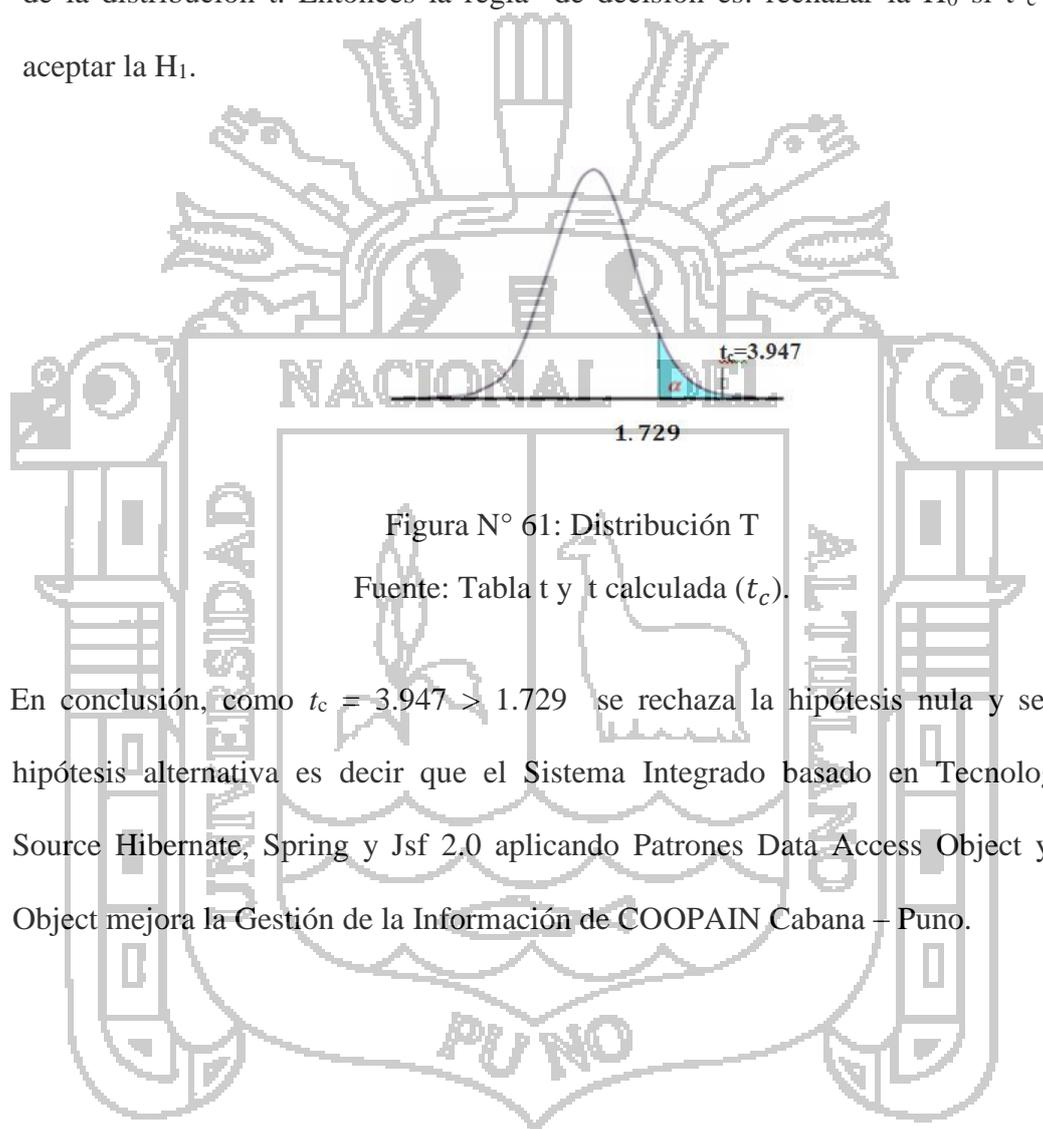
$$t_c = \frac{\bar{X} - \mu}{\frac{S}{\sqrt{n}}}$$

$$t_c = \frac{23.05 - 22}{\frac{1.191}{\sqrt{20}}}$$

$$t_c = \frac{1.05}{0.266}$$

$$t_c = 3.947$$

Para la regla de decisión se trabajó con la distribución t donde en la tabla t muestra que para un nivel de significancia de $\alpha=0.05$ y 19 grados de libertad, el valor crítico t es: 1.729. Tiene un valor positivo debido a que la hipótesis alternativa especifica que la media es mayor que 22, es decir, la región crítica se encuentra a la derecha de la media 0 de la distribución t. Entonces la regla de decisión es: rechazar la H_0 si $t_c \geq 1.729$ y aceptar la H_1 .



En conclusión, como $t_c = 3.947 > 1.729$ se rechaza la hipótesis nula y se acepta la hipótesis alternativa es decir que el Sistema Integrado basado en Tecnologías Open Source Hibernate, Spring y Jsf 2.0 aplicando Patrones Data Access Object y Business Object mejora la Gestión de la Información de COOPAIN Cabana – Puno.

CONCLUSIONES

PRIMERA: Desarrollando el Sistema Integrado basado en Tecnologías Open Source aplicando Patrones Data Access Object y Business Object, se mejoró la gestión de la información de COOPAIN Cabana, permitiendo realizar los diferentes procesos con mayor control, reducción de costos y tiempo con un alto grado de productividad

SEGUNDA: La obtención de requerimientos para el desarrollo del Sistema fue primordial ya que ayudó a reunir todas las necesidades y problemas que tiene el usuario, teniendo como resultado la satisfacción del Usuario.

TERCERA: Con el análisis y diseño del Sistema permitió especificar qué es lo que el Sistema debe hacer. Se logró conocer a detalle todas las necesidades de los usuarios y establecer las funcionalidades del nuevo Sistema para el logro de objetivos de COOPAIN Cabana.

CUARTA: Con la implementación del Sistema aplicando Patrones Data Access Object y Business Object, se logró construir un producto robusto que está sujeto a cambios futuros, reutilización de código y proveen un marco de trabajo amigable además de la interacción con diferentes lenguajes, que permitirá a COOPAIN Cabana realizar sus labores para beneficio de la organización.

QUINTA: Con la encuesta elaborada en su mayoría los usuarios califican como Excelente el Sistema Integrado, con una ponderación de 23 en base a 25; lo que comprueba que el Sistema Integrado basado en Tecnologías Open Source mejora en un 92% en los procesos

de gestión de información de COOPAIN Cabana, permitiendo realizar los diferentes procesos con mayor control, reducción de costos y tiempo con un alto grado de productividad.



RECOMENDACIONES Y SUGERENCIAS

PRIMERA: Se recomienda investigar sobre otras tecnologías Open Source, metodologías de desarrollo, normas de calidad de software y buenas practicas que pudieran mejorar la gestión de la información, tales como: MPLS, ITIL, COBIT, normas ISO, etc.

SEGUNDA: Se recomienda que para una buena obtención de requerimientos de usuario se realice la recopilación de datos, el uso del método de comunicación directa y entrevistas para este tipo de investigaciones por que ayudan a extraer información real, y de esta forma realizar un mejor análisis y diseño del sistema.

TERCERA: Se recomienda investigar los diferentes patrones de diseño y patrones de arquitectura.

CUARTA: Se recomienda que al desarrollar un Sistema es necesario seguir un proceso de plan de trabajo para desarrollar actividades ordenadas y cumplir con las metas. Verificar la portabilidad a otro gestor de base de datos aplicando la potencialidad del Framework Hibernate.

QUINTA: Ampliar la funcionalidad la obtención de reportes con información estadística para la toma de decisiones.

BIBLIOGRAFÍA

- AGÜERO, M. (2007). *Introducción a Spring Framework*. Recuperado el 10 de 08 de 2013, de http://www.palermo.edu/ingenieria/downloads/introduccion_spring_framework_v1.0.pdf
- AVILA, R. (2001). *Metodología de la investigación*. Lima: Estudios y ediciones R.A.
- BAUER, C. &. (2004). *Hibernate in Action* (2da ed.). United States of America: Manning Publications Co.
- BEYNON, P. (2004). *Database System*. Madrid: Palgrave Macmillan.
- BISQUERRA, R. (2004). *Metodología de la Investigación* (3ra ed.). Madrid: La muralla.
- CARRILLO, I. P. (2008). *Metodología de desarrollo de software*. Buenos Aires: Ciencia que ladra.
- CASILLAS, R. (2004). *Desarrollo de aplicaciones web*. Barcelona: Eureka.
- CASILLAS, R. (2005). *Base de Datos*. Barcelona: Eureka.
- CAZAU, P. (2006). *Introducción a la investigación*. Buenos Aires: Ciencia que ladra.
- CHOO, C. W. (2002). *Information Management for the Intelligent Organization: The Art of Scanning the Environment* (3ra ed.). American Society for Information Science.
- DBSHARDS. (2009). *Data Access Object*. Recuperado el 10 de 08 de 2013, de <http://www.codefutures.com/data-access-object/>
- FIELDS, D. &. (2000). *Web Development with JavaServer Pages* (2da ed.). Estados Unidos: Manning Publications Co.
- GONCALVES, A. (2009). *Beginning Java EE 6 with GlassFish 3* (2da ed.). Estados Unidos: Apress.
- HERNANDEZ, S. F. (2006). *Metodología de la Investigación* (4ta ed.). México: McGrawHill.
- JACOBSON, I. B. (2000). *El Proceso Unificado de Desarrollo de Software*. Madrid: Pearson Education.

- KAPUR, R. B. (2010). *Open Source Development* (1ra ed.). Canada: IBM Canada.
- KENDALL, E. &. (2005). *Análisis y Diseño de Sistemas*. México: Pearson Education.
- KING, G. B. (2010). *Hibernate Reference Documentation*.
- LA GRECA, P. (27 de 12 de 2007). *Tutorial de Spring*. Recuperado el 10 de 08 de 2013, de http://pablolg.wikispaces.com/file/view/spring_tutorial_v0.271.pdf
- LARMAN, C. (2003). *UML y Patrones*. Madrid: Prentice Hall.
- LAUDON, K. &. (2002). *Sistemas de Información Gerencial*. México: Alhambra.
- LINWOOD, J. &. (2010). *Beginning Hibernate (Expert's Voice in Java)* (2da ed.). Estados Unidos: Apress.
- MANN, K. D. (2005). *JavaServer Faces in Action*. Estados Unidos: Manning Publications Co.
- MARTIN, A. (2008). *Programador Certificado JAVA 2* (2da ed.). Madrid, España: Alfaomega, RA-MA.
- MORALES, P. (13 de Diciembre de 2012). *Estadística Aplicada a las Ciencias Sociales: Tamaño necesario de la muestra*, Recuperado el 10 de Agosto de 2013, de <http://www.upcomillas.es/personal/peter/investigacion/Tama%F1omuestra.pdf>
- PRESMAN, R. (2009). *Ingeniería del software: un enfoque práctico* (6th ed.). Madrid: McGraw-Hill.
- PRYTHERCH, R. (2000). *Harrod's Librarian's Glossary and Reference Book*. Gower.
- QUATRANI, T. (2000). *Visual modeling with Rational Rose and UML*. Estados Unidos: Technology Series.
- ROLDÁN, D. V. (2010). *Aplicaciones Web: Un Enfoque Práctico*. México: ALFAOMEGA RA-MA.
- SILBERSCHATZ, A. (2002). *Fundamentos de Base de Datos* (4ta ed.). Madrid: Concepción Fernández.
- SOMERVILLE, I. (2005). *Ingeniería del Software*. Madrid: Pearson Education.

TAPIA, S. (2011). Integración JSF - Spring - Hibernate. Quito Ecuador.

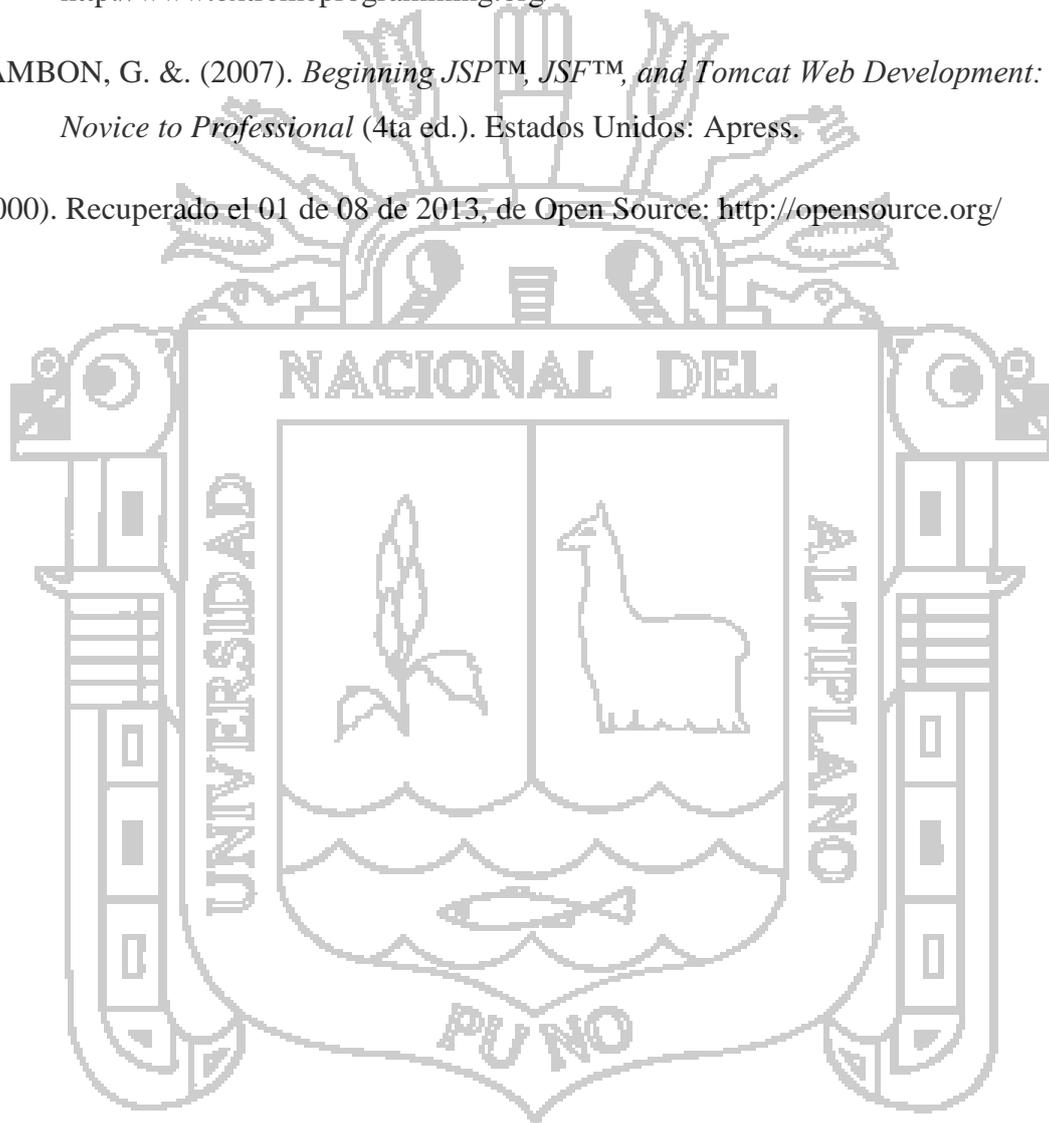
VÉLEZ, C. (2001). *Apuntes de metodología de la investigación*. Colombia: EAFIT.

WALLS, C. (2011). *Spring* (3ra ed.). Estados Unidos: Anaya Multimedia.

WELLS, D. (2009). *Extreme Programming XP*. Recuperado el 01 de 08 de 2013, de <http://www.extremeprogramming.org/>

ZAMBON, G. &. (2007). *Beginning JSP™, JSF™, and Tomcat Web Development: From Novice to Professional* (4ta ed.). Estados Unidos: Apress.

(2000). Recuperado el 01 de 08 de 2013, de Open Source: <http://opensource.org/>







1. Descripción de Historias de Usuario

1.1. Historia de Usuario: Módulo Producción

Historia de Usuario			
Número: PRO01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Registro de productores		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar productores, según a los datos solicitados.			
Observaciones: Se hace el registro de la mayoría de datos del productor.			

Historia de Usuario			
Número: PRO02	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Registro de área de cultivos orgánicos del productor(Unidad Productiva)		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar las áreas de cultivos orgánicos del productor según productor.			

Historia de Usuario			
Número: PRO03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Registro de descripción de la unidad productiva		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar la descripción de la unidad productiva según a cultivos orgánicos.			



Historia de Usuario			
Número: PRO04	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Registro de informes de inspección		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar la los informes de inspección según a cada productor.			

1.2. Historia de Usuario: Módulo Acopio y Almacén

Historia de Usuario			
Número: ACA01	Usuario: Jefe de Almacén, Jefe de Acopio		
Nombre Historia:	Registro de acopio de quinua		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar el acopio de quinua de un determinado productor.			

Historia de Usuario			
Número: ACA02	Usuario: Jefe de Almacén, Jefe de Acopio		
Nombre Historia:	Generar parte de ingreso a almacén (PIA)		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 4 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite generar o imprimir PIA, en el cual incluye detalles del acopio de quinua.			

Historia de Usuario			
Número: ACA03	Usuario: Jefe de Acopio		
Nombre Historia:	Generar un lote de compra(semanal)		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite generar un lote de compra según fecha.			

Historia de Usuario			
Número: ACA04	Usuario: Jefe de Almacén		
Nombre Historia:	Registro del Ingreso de quinua a almacén		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar el ingreso de quinua a almacén			

1.3. Historia de Usuario: Módulo Procesamiento

Historia de Usuario			
Número: PRC01	Usuario: Jefe de Planta		
Nombre Historia:	Registro de Muestra de Pre embarque		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite Enviar la muestra de pre embarque a la oficina COOPAIN Juliaca.			



Historia de Usuario			
Número: PRC02	Usuario: Jefe de Planta		
Nombre Historia:	Registro de Orden de Proceso		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar una orden de proceso.			

Historia de Usuario			
Número: PRC03	Usuario: Jefe de Planta		
Nombre Historia:	Registro de formato de procesamiento en el área de despedrado y escarificado		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite el formato de procesamiento en el área de despedrado y escarificado.			

Historia de Usuario			
Número: PRC04	Usuario: Jefe de Planta		
Nombre Historia:	Registro de formato de procesamiento en el área de lavado y centrifugado		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite realizar el formato de procesamiento del área de secado.			



1.4. Historia de Usuario: Módulo Comercialización

Historia de Usuario			
Número: COM01	Usuario: Asistente Administrativo, Gerente General		
Nombre Historia:	Realizar solicitud de envió de muestra de pre embarque		
Prioridad:	Baja	Riesgo en desarrollo:	Alta
Puntos Estimados: 4 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite realizar una solicitud de muestra de pre embarque a planta de procesamiento Juliaca.			

Historia de Usuario			
Número: COM02	Usuario: Asistente Administrativo, Gerente General		
Nombre Historia:	Administrar lotes de venta		
Prioridad:	Baja	Riesgo en desarrollo:	Alta
Puntos Estimados: 4 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar todo un proceso de exportación de un determinado contrato y realizar el seguimiento respectivo.			

1.5. Historia de Usuario: Módulo Gerencia

Historia de Usuario			
Número: GER01	Usuario: Gerente General		
Nombre Historia:	Registrar convocatorias a reuniones		
Prioridad:	Baja	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar y enviar las convocatorias y/o citaciones a reuniones.			

Historia de Usuario			
Número: GER02	Usuario: Gerente General		
Nombre Historia:	Registrar solicitud de informes		
Prioridad:	Baja	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite enviar una solicitud de informes al personal que labora en la entidad.			

1.6. Historia de Usuario: Módulo Asistencia Personal Administrativo

Historia de Usuario			
Número: APA01	Usuario: Asistente Administrativo		
Nombre Historia:	Ver reporte de asistencia del personal		
Prioridad:	Baja	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite ver el reporte de asistencia de personal en planta y en oficina de administración de COOPAIN CABANA.			

Historia de Usuario			
Número: APA02	Usuario: Asistente Administrativo		
Nombre Historia:	Registrar contratos de personal		
Prioridad:	Baja	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar contratos con personal.			

1.7. Historia de Usuario: Módulo Control Documentario

Historia de Usuario			
Número: CDO01	Usuario: Asistente Administrativo		
Nombre Historia:	Registrar formatos de documentos de redacción		
Prioridad:	Baja	Riesgo en desarrollo:	Alta
Puntos Estimados: 5 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar y controlar la numeración de los documentos oficios, memorándums, solicitudes, requerimientos generados en la institución.			

1.8. Historia de Usuario: Módulo Caja

Historia de Usuario			
Número: CAJ01	Usuario: Cajero		
Nombre Historia:	Registrar retiros de dinero en banco		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar todo los retiros de dinero realizados.			

Historia de Usuario			
Número: CAJ02	Usuario: Cajero		
Nombre Historia:	Registrar pago de PIA's (Parte de Ingreso a Almacén)		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 4 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar los pagos al productor de los acopios realizados.			

Historia de Usuario			
Número: CAJ03	Usuario: Cajero		
Nombre Historia:	Registrar pago a personal		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Puntos Estimados: 1 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar los pagos que realizan al personal que labora en la institución.			

1.9. Historia de Usuario: Módulo Transporte

Historia de Usuario			
Número: TRA01	Usuario: Jefe de Transporte		
Nombre Historia:	Registro de Transportista		
Prioridad:	Alta	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite el registro de transportistas.			

Historia de Usuario			
Número: TRA02	Usuario: Jefe de Transporte		
Nombre Historia:	Registro de Unidad de Transporte		
Prioridad:	Alta	Riesgo en desarrollo:	Baja
Puntos Estimados: 1 día	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite el registro de la unidad de transporte.			

Historia de Usuario			
Número: TRA03	Usuario: Jefe de Transporte		
Nombre Historia:	Registro de Contrato con transportista		
Prioridad:	Alta	Riesgo en desarrollo:	Baja
Puntos Estimados: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite el registro de la unidad de transporte.			

Historia de Usuario			
Número: TRA04	Usuario: Jefe de Transporte		
Nombre Historia:	Registro de programación de transportistas		
Prioridad:	Alta	Riesgo en desarrollo:	Baja
Puntos Estimados: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite el registro de programación de transportistas para los diferentes destinos (Juliaca – Lima, Lima – Puerto callao).			

1.10. Historia de Usuario: Módulo Herramientas de Administración del Sistema

Historia de Usuario			
Número: HAS01	Usuario: Todos		
Nombre Historia:	Gestionar el uso de perfiles usuarios y roles en el sistema.		
Prioridad:	Alta	Riesgo en desarrollo:	Baja
Puntos Estimados: 5 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Inicio de sesión según rol asignado.			





1. Historias de Usuario

1.1. Primera Iteración: Módulo Producción

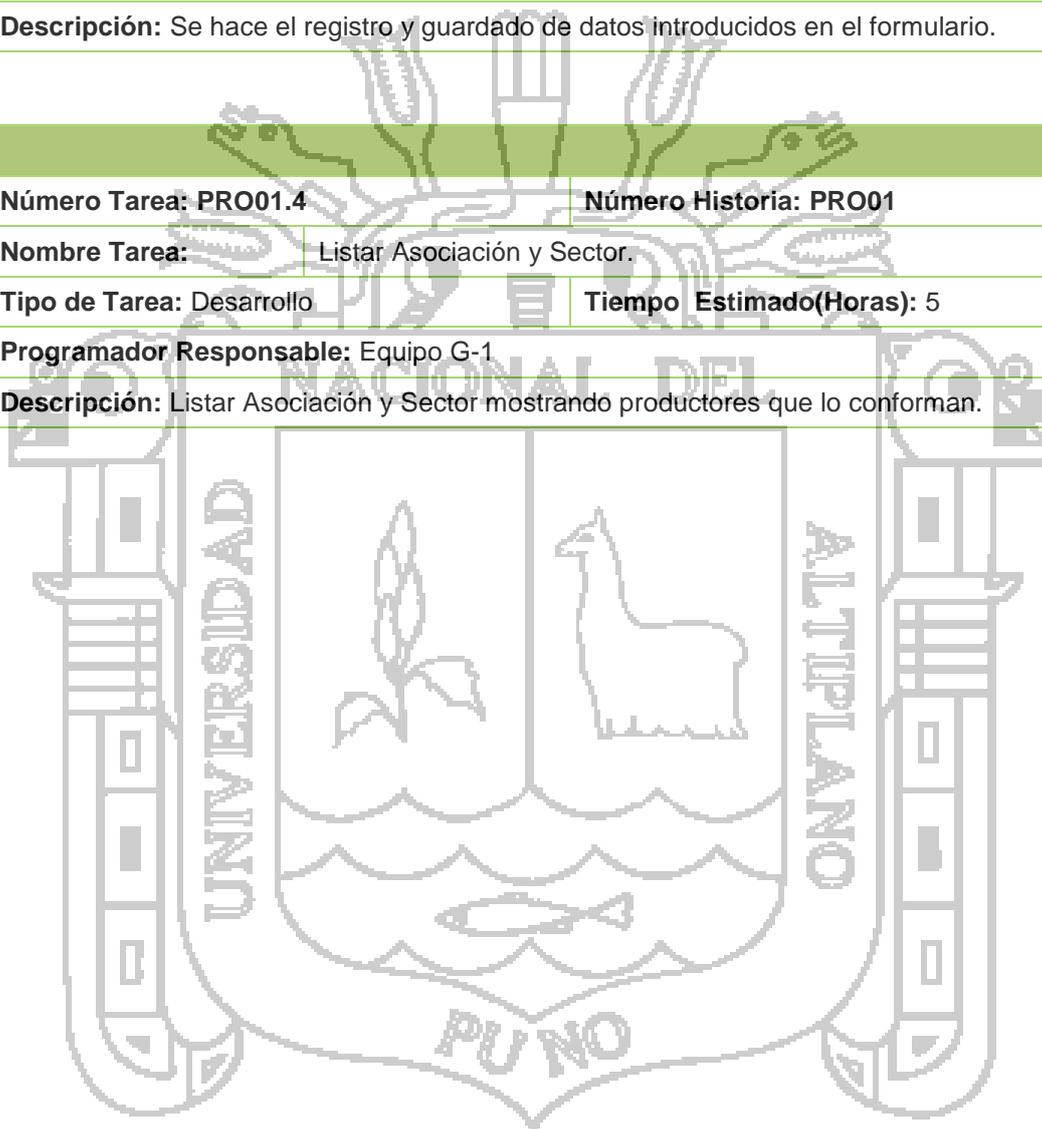
Historia de Usuario			
Módulo: Producción			
Número: PRO01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Asociación y Sector		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 2 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Asociación y Sector, según a los siguientes datos solicitados: Registrar y Editar Asociación: <ul style="list-style-type: none"> ➤ Código. ➤ Nombres. ➤ Descripción. ➤ Inicio. ➤ Ubicación. Registrar y Editar Sector: <ul style="list-style-type: none"> ➤ Nombre ➤ Descripción Listar Asociación y Sector mostrando productores que lo conforman.			
Observaciones:			

Tarea	
Número Tarea: PRO01.1	Número Historia: PRO01
Nombre Tarea:	Diseñar Base de datos para registro y edición de Asociación y Sector
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Asociaciones y Sectores.	

Tarea	
Número Tarea: PRO01.2	Número Historia: PRO01
Nombre Tarea:	Crear interfaz gráficas para registrar Asociación y Sector.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño del formulario en el que se introducirán nueva Asociación y Sector.	

Tarea	
Número Tarea: PRO01.3	Número Historia: PRO01
Nombre Tarea:	Registro de Datos solicitados de Asociación y Sector.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Se hace el registro y guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO01.4	Número Historia: PRO01
Nombre Tarea:	Listar Asociación y Sector.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Listar Asociación y Sector mostrando productores que lo conforman.	





Historia de Usuario			
Módulo: Producción			
Número: PRO02	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Productores, Inscripciones y Generar Reporte de Productores		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días		Iteración Asignada: 1	
Programador Responsable: Equipo G-1			
<p>Descripción: Permite registrar y editar Productores e Inscripciones, según a los siguientes datos solicitados:</p> <p>Registro y Edición de Productores:</p> <ul style="list-style-type: none"> ➤ Código ➤ Nombres ➤ Apellidos ➤ Grado de Instrucción(Educación) ➤ Asociación ➤ Sector ➤ Documento de Identidad ➤ Nombre del conyugue ➤ Nro. de hijos ➤ Nro. de padrón ➤ Dirección ➤ Nro. de Hijos dependientes ➤ Nro. de Hijos independientes ➤ Fecha de registro ➤ Estado <p>Registro y Edición de Inscripciones:</p> <ul style="list-style-type: none"> ➤ Programa(Producción Orgánica o Convencional) ➤ Fecha de Inicio ➤ Fecha Finalización ➤ Contrato ➤ Solicitud ➤ Carta Compromiso <p>Listar Productores según asociación. Listar e imprimir Inscripciones según productor y asociación. Generar Reporte según Sector, Tipo de Productor y Género.</p>			
Observaciones:			

Tarea	
Número Tarea: PRO02.1	Número Historia: PRO02
Nombre Tarea:	Diseñar Base de datos para registro de Productor e Inscripción.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Productor e Inscripción.	

Tarea	
Número Tarea: PRO02.2	Número Historia: PRO02
Nombre Tarea:	Crear interfaz gráficas para registrar Productor e Inscripción.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán nuevo Productor e Inscripción.	

Tarea	
Número Tarea: PRO02.3	Número Historia: PRO02
Nombre Tarea:	Registro y Edición de Datos solicitados de Productor e Inscripción.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace el registro, edición y guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO02.4	Número Historia: PRO02
Nombre Tarea:	Listar Productores e Inscripciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Listar Productores e Inscripciones según asociación. Imprimir Inscripciones según al productor.	

Tarea	
Número Tarea: PRO02.5	Número Historia: PRO02
Nombre Tarea:	Generar Reporte de Productor.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Generación de Reporte de Productores con datos como código, nombre, dirección, sector y programa.	

Historia de Usuario			
Módulo: Producción			
Número: PRO03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Campañas y Parcelas		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
Descripción: Permite registrar, editar y listar Campañas y Parcelas, según a los siguientes datos solicitados: Registrar y Editar Campañas: <ul style="list-style-type: none"> ➤ Nombres ➤ descripción ➤ Estado Registrar y Editar Parcelas: <ul style="list-style-type: none"> ➤ Unidad Productiva ➤ Código ➤ Nombre ➤ Lado ➤ Altura ➤ Hectáreas ➤ Ubicación Listar Campañas y Parcelas			
Observaciones:			

Tarea	
Número Tarea: PRO03.1	Número Historia: PRO03
Nombre Tarea:	Diseñar Base de datos para registro de Campañas y Parcelas.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Campañas y Parcelas.	

Tarea	
Número Tarea: PRO03.2	Número Historia: PRO03
Nombre Tarea:	Crear interfaz gráficas para registrar y editar Campañas y Parcelas.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán nuevo Campañas y Parcelas.	



Tarea	
Número Tarea: PRO03.3	Número Historia: PRO03
Nombre Tarea:	Registro de Datos solicitados de Campañas y Parcelas.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace el registro y guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO03.4	Número Historia: PRO03
Nombre Tarea:	Listar Campañas y Parcelas.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Listar Campañas y Parcelas (según Productor).	

Historia de Usuario	
Módulo: Producción	
Número: PRO04	Usuario: Equipo técnico, Gerente de Producción
Nombre Historia:	Gestionar datos de Unidad Productiva
Prioridad: Alta	Riesgo en desarrollo: Alta
Tiempo Estimado: 2 días	Iteración Asignada: 1
Programador Responsable: Equipo G-1	
Descripción: Permite registrar y editar Unidad Productiva de un Productor, según a los siguientes datos solicitados: <ul style="list-style-type: none"> ➤ Código ➤ Nombre ➤ Hectárea ➤ Ubicación Listar Unidad Productiva según Productor	
Observaciones:	

Tarea	
Número Tarea: PRO04.1	Número Historia: PRO04
Nombre Tarea:	Diseñar Base de datos para registro de Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Campañas y Parcelas.	

Tarea	
Número Tarea: PRO04.2	Número Historia: PRO04
Nombre Tarea:	Crear interfaz gráficas para registrar Productor e Inscripción.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de Unidad Productiva.	

Tarea	
Número Tarea: PRO04.3	Número Historia: PRO04
Nombre Tarea:	Registro y edición de Datos solicitados de Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO04.4	Número Historia: PRO04
Nombre Tarea:	Listar Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Listar Unidad Productiva según a un determinado Productor.	

Historia de Usuario			
Módulo: Producción			
Número: PRO05	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Estimaciones		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 1		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite gestionar datos de Estimaciones según un determinado Productor según a los siguientes datos solicitados: Registro y Edición de Estimaciones:</p> <ul style="list-style-type: none"> ➤ Fecha ➤ Campaña ➤ Estimación de parcela <ul style="list-style-type: none"> ○ Parcela ○ Producto ○ Área ○ Estimación en Tn/Hectáreas ○ Estimación Tn/Parcela <p>Listar Estimaciones según Productor Detallar productos estimados para una campaña de un productor.</p>			
Observaciones:			

Tarea	
Número Tarea: PRO05.1	Número Historia: PRO05
Nombre Tarea:	Diseñar Base de datos para registro de Estimaciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Estimaciones.	

Tarea	
Número Tarea: PRO05.2	Número Historia: PRO05
Nombre Tarea:	Crear interfaz gráficas para registrar y editar Estimaciones
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de Estimaciones por parcela.	



Tarea	
Número Tarea: PRO05.3	Número Historia: PRO05
Nombre Tarea:	Registro y edición de Datos solicitados de Estimaciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO05.4	Número Historia: PRO05
Nombre Tarea:	Listar Estimaciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Listar Estimaciones según a un determinado Productor. Detallar los productos según a una estimación.	



Historia de Usuario			
Módulo: Producción			
Número: PRO06	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Ficha Unidad Productiva		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 5 días		Iteración Asignada: 1	
Programador Responsable: Equipo G-1			
<p>Descripción: Permite gestionar datos de Ficha Unidad Productiva según a los siguientes datos solicitados:</p> <p>Registro y Edición de Ficha Unidad Productiva:</p> <ul style="list-style-type: none"> ➤ Fecha ➤ Campaña ➤ Producción anterior ➤ Producción presente ➤ Asesoría Técnica ➤ área alquilada (has) ➤ Descanso terreno ➤ Cultivos orgánicos <ul style="list-style-type: none"> ○ Tipo cultivo(papa, quinua) ○ Hectáreas ○ Malezas ➤ Ganados <ul style="list-style-type: none"> ○ Tipo ganado ○ Cantidad ➤ Factores <ul style="list-style-type: none"> ○ Factor(Plagas, Heladas) ○ Detalles ➤ Conocimiento <ul style="list-style-type: none"> ○ Noma Orgánica(fertilizantes) ➤ Insumos <ul style="list-style-type: none"> ○ Insumo(Guano fermentados) ○ Aplicación(guano) ○ Preparación ➤ Control del plagas <ul style="list-style-type: none"> ○ Plaga ○ Suelo ○ Cultivo ○ Almacén <p>Listar Ficha Unidad Productiva según Productor Imprimir Ficha de Ficha Unidad Productiva de un Productor</p>			

Tarea	
Número Tarea: PRO06.1	Número Historia: PRO06
Nombre Tarea:	Diseñar Base de datos para registro de Ficha Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 12
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Ficha Unidad Productiva.	

Tarea	
Número Tarea: PRO06.2	Número Historia: PRO06
Nombre Tarea:	Crear interfaz gráficas para registrar y editar Ficha Unidad Productiva
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de Ficha Unidad Productiva.	

Tarea	
Número Tarea: PRO06.3	Número Historia: PRO06
Nombre Tarea:	Registro y edición de Datos de Ficha Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 16
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO06.4	Número Historia: PRO06
Nombre Tarea:	Listar Ficha Unidad Productiva.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar Ficha Unidad Productiva según a un determinado Productor e imprimir.	

Historia de Usuario			
Módulo: Producción			
Número: PRO07	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Inspecciones		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días		Iteración Asignada: 1	
Programador Responsable: Equipo G-1			
<p>Descripción: Permite gestionar datos de Inspecciones de un Productor según a los siguientes datos solicitados:</p> <p>Registro de Estimaciones:</p> <ul style="list-style-type: none"> ➤ Fecha ➤ Ficha Unidad Productiva <ul style="list-style-type: none"> ○ Parcela ○ Tipo Cultivo ○ Producto ○ Área ○ Estimación Tn/Has ○ Densidad Siembra ○ Orgánicamente ○ 1ra Inspección ○ Rendimiento ○ Insumos Utilizados <ul style="list-style-type: none"> ➤ Insumo(guano) ➤ Origen ➤ Cantidad ➤ Concentración ○ Observaciones ○ Riesgos ○ Prevención ➤ Inspección parcela ➤ Abonos verdes ➤ Parcela mixta ➤ Diario trabajo ➤ Compra insumos ➤ Riesgos directos <ul style="list-style-type: none"> ○ Riesgo ○ Bajo ○ Medio ○ Alto ➤ Descripciones ➤ Resultado SIC ➤ Justificación <p>Listar Estimaciones según Productor Imprimir Inspecciones de un Productor</p>			
Observaciones:			

Tarea	
Número Tarea: PRO07.1	Número Historia: PRO07
Nombre Tarea:	Diseñar Base de datos para registro de Inspecciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Inspecciones.	

Tarea	
Número Tarea: PRO07.2	Número Historia: PRO07
Nombre Tarea:	Crear interfaz gráficas para registrar y editar Inspecciones
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de Inspecciones.	

Tarea	
Número Tarea: PRO07.3	Número Historia: PRO07
Nombre Tarea:	Registro y edición de Datos de Inspecciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRO07.4	Número Historia: PRO07
Nombre Tarea:	Listar Ficha Inspecciones.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar Inspecciones según a un determinado Productor e imprimir.	





1. Historia de Usuario

1.1. Segunda Iteración: Módulo Acopio y Almacén

Historias de Usuario: Acopio y Almacén

Historia de Usuario			
Módulo: Acopio y Almacén			
Módulo: Acopio y Almacén			
Número: ACA01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Productor		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 9 días	Iteración Asignada: 2		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Listar Acopio, según a los siguientes datos solicitados: Registrar Productor Acopio; <ul style="list-style-type: none"> ➤ Código. ➤ Nombre. ➤ Apellido Paterno. ➤ Apellido Materno. ➤ Agregar Documento. <ul style="list-style-type: none"> ○ Tipo Documento. ○ Número de Documento. ➤ Tipo de Persona (Natural y Jurídica). ➤ Sexo. ➤ Dirección. ➤ Número. Hijos. ➤ Hijos Dependientes. ➤ Hijos Independientes. ➤ Número de Padrón. ➤ Educación. ➤ Asociación ➤ Sector. Listar Productores Acopios mostrando productores que lo conforman.			
Observaciones:			

		Tarea
Número Tarea: ACA01.1	Número Historia: ACA01	
Nombre Tarea:	Diseñar Base de Datos para registro de Productor.	
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2	
Programador Responsable: Equipo G-1		
Descripción: Realizar el diseño de base de datos para el Registro de Productor.		

Tarea	
Número Tarea: ACA01.2	Número Historia: ACA01
Nombre Tarea:	Crear interfaz gráficas para registrar Productor.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Productor.	

Tarea	
Número Tarea: ACA01.3	Número Historia: ACA01
Nombre Tarea:	Registro y Edición de datos de Productor.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: ACA01.4	Número Historia: ACA01
Nombre Tarea:	Listar registro de Productor.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Productor, con las acciones respectivas.	

Historia de Usuario			
Módulo: Acopio y Almacén			
Número: ACA02	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Acopio		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 9 días	Iteración Asignada: 2		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite Registrar, Editar, Listar Acopio, según a los siguientes datos solicitados: Registrar y Editar Acopio de Productor:</p> <ul style="list-style-type: none"> ➤ Serie. ➤ Número. ➤ Fecha Comprobante. ➤ Acopios Productos: <ul style="list-style-type: none"> ○ Producto. ○ Parcela. ○ Precio Unitario (Kg). ○ Peso (Kg). ○ Precio Total. ○ Control de Calidad. ○ Paquete Acopiado <ul style="list-style-type: none"> ▪ Tipo de Paquete. ▪ Peso (Kg). ▪ Tarjeta. ➤ Total. ➤ Contado. ➤ Saldo. ➤ Almacén. ➤ Campaña. ➤ Lote de Compra. ➤ Observaciones. ➤ Fecha de Acopio. <p>Listar Acopio de Productor mostrando los datos que lo conforman. Generar PIA de Productor. Registrar Pago de Productor:</p> <ul style="list-style-type: none"> ➤ Fecha de Acopio. ➤ PIA. ➤ Total. ➤ Pagado. ➤ Pagar. ➤ Saldo. 			
Observaciones:			

Tarea	
Número Tarea: ACA02.1	Número Historia: ACA02
Nombre Tarea:	Diseñar Base de Datos para registro de Acopio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Acopio.	

Tarea	
Número Tarea: ACA02.2	Número Historia: ACA02
Nombre Tarea:	Crear interfaz gráficas para registrar Acopio.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Acopio.	

Tarea	
Número Tarea: ACA02.3	Número Historia: ACA02
Nombre Tarea:	Registro y Edición de datos de Acopio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: ACA02.4	Número Historia: ACA02
Nombre Tarea:	Listar registro de Acopio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Acopio, con las acciones respectivas.	

Tarea	
Número Tarea: ACA02.5	Número Historia: ACA02
Nombre Tarea:	Generar Reporte de PIA.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Generar reporte de PIA.	

Tarea	
Número Tarea: ACA02.6	Número Historia: ACA02
Nombre Tarea:	Registro y Edición de Pagos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario de Pago.	

Historia de Usuario			
Módulo: Acopio y Almacén			
Número: ACA03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Lotes de Compra		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 9 días	Iteración Asignada: 2		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Lotes de Compra, según a los siguientes datos solicitados: Registrar y Editar Lotes de Compra: <ul style="list-style-type: none"> ➤ Nombre. ➤ Fecha de Inicio. ➤ Fecha Fin. ➤ Estado. Listar Lotes de Compra mostrando nombre del Lote de Compra, fecha inicio, fecha fin y estado.			
Observaciones:			

Tarea	
Número Tarea: ACA03.1	Número Historia: ACA03
Nombre Tarea:	Diseñar Base de Datos para registro de Lote de Compra.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Lote de Compra.	

Tarea	
Número Tarea: ACA03.2	Número Historia: ACA03
Nombre Tarea:	Crear interfaz gráficas para registrar Lote de Compra.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Lote de Compra.	



Tarea	
Número Tarea: ACA03.3	Número Historia: ACA03
Nombre Tarea:	Registro y Edición de datos de Lote de Compra.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: ACA03.4	Número Historia: ACA03
Nombre Tarea:	Listar registro de Lote de Compra.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Lote de Compra, con las acciones respectivas.	

Historia de Usuario			
Módulo: Acopio y Almacén			
Número: ACA04	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Generar Reporte de Acopio		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 2		
Programador Responsable: Equipo G-1			
Descripción: Permite Generar Reporte, según a los siguientes datos solicitados: Generar Reporte de Acopio:			
<ul style="list-style-type: none"> ➤ Tipo de Reporte. ➤ Selección de Periodo (Día, Mes, Año, Intervalo de Fechas). ➤ Fecha Seleccionada. 			
Observaciones:			

Tarea	
Número Tarea: ACA04.1	Número Historia: ACA04
Nombre Tarea:	Realizar el Query-SQL para la consulta a la Base de Datos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realzar el script para la consulta a la base de datos y generar el reporte solicitado.	



Tarea	
Número Tarea: ACA04.2	Número Historia: ACA04
Nombre Tarea:	Crear interfaz gráficas del reporte de Acopio.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del modelo de reporte.	

Historia de Usuario			
Módulo: Acopio y Almacén			
Número: ACA05	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Generar Reporte de Estimaciones de producción de Productores		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 2		
Programador Responsable: Equipo G-1			
Descripción: Permite Generar Reporte, según a los siguientes datos solicitados: Generar Reporte de Acopio: <ul style="list-style-type: none"> ➤ Tipo de Reporte. ➤ Selección de Periodo (Día, Mes, Año, Intervalo de Fechas). ➤ Fecha Seleccionada. 			
Observaciones:			

Tarea	
Número Tarea: ACA05.1	Número Historia: ACA05
Nombre Tarea:	Realizar el Query-SQL para la Consulta a la Base de Datos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realzar el script para la consulta a la base de datos y generar el reporte solicitado.	

Tarea	
Número Tarea: ACA05.2	Número Historia: ACA05
Nombre Tarea:	Crear interfaz gráficas del reporte de estimación de producción de productores.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del modelo de reporte.	



1. Historia de Usuario

1.1. Tercera Iteración: Módulo Procesamiento

Historias de Usuario: Procesamiento

Historia de Usuario			
Módulo: Procesamiento			
Número: PRC01	Usuario: Jefe de Planta		
Nombre Historia:	Gestionar datos de Procesamiento.		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 10 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite gestionar datos de Procesamiento según a los siguientes datos solicitados:</p> <p>Registro de Procesamiento:</p> <ul style="list-style-type: none"> ➤ Campaña ➤ Almacén ➤ Lote de Acopio (lote1, lote2, etc.) ➤ Color de Quinua (Blanco Orgánico, Rojo Orgánico, etc.) ➤ Fecha de Proceso ➤ Paquetes Acopiados <ul style="list-style-type: none"> ○ Tarjeta ○ Número ○ Cantidad ○ Observaciones ➤ Cantidad Procesada ➤ Peso de lote de acopio <p>Listar Procesamientos</p> <p>Detallar Procesamiento</p> <ul style="list-style-type: none"> ➤ Registrar, Listar e Imprimir Despedrado y Escarificado ➤ Registrar, Listar e Imprimir Lavado y centrifugado ➤ Registrar, Listar e Imprimir Secado ➤ Registrar, Listar e Imprimir Despedrado II ➤ Registrar, Listar e Imprimir Clasificado ➤ Registrar, Listar e Imprimir Puntos Negros <p>Registrar, Listar e Imprimir Empaquetados</p>			
Observaciones:			

Tarea	
Número Tarea: PRC01.1	Número Historia: PRC01
Nombre Tarea:	Diseñar Base de datos para registro de Procesamiento.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Procesamiento.	

Tarea	
Número Tarea: PRC01.2	Número Historia: PRC01
Nombre Tarea:	Diseñar Base de datos para registro de paquetes procesados.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro paquetes procesados.	

Tarea	
Número Tarea: PRC01.3	Número Historia: PRC01
Nombre Tarea:	Crear interfaz gráficas para registrar Procesamientos
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de procesamiento	

Tarea	
Número Tarea: PRC01.4	Número Historia: PRC01
Nombre Tarea:	Crear interfaz gráfica para registrar paquetes acopiados.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de paquetes acopiados.	

Tarea	
Número Tarea: PRC01.5	Número Historia: PRC01
Nombre Tarea:	Registro de Procesamiento.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC01.6	Número Historia: PRC01
Nombre Tarea:	Registro de paquetes acopiados.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC01.7	Número Historia: PRC01
Nombre Tarea:	Listar Procesamientos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Listar Procesamientos según fecha de inicio, total procesado y detalles.	

Tarea	
Número Tarea: PRC01.8	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir Detalles Despedrado y Escarificado.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Registrar Despedrado y Escarificado según a datos solicitados	
<ul style="list-style-type: none"> ➤ Cantidad Procesada ➤ Impurezas ➤ Saponina ➤ Piedrecillas ➤ Cantidad para lavado ➤ Observaciones ➤ Fecha de proceso ➤ Hora inicio ➤ Hora fin ➤ Estado 	
Listar e imprimir	



Tarea	
Número Tarea: PRC01.9	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir lavado y centrifugado.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Registrar lavado y centrifugado según a datos solicitados <ul style="list-style-type: none"> ➤ Cantidad Procesada ➤ Impurezas ➤ Cantidad para secado ➤ Observaciones ➤ Fecha de proceso ➤ Hora inicio y fin ➤ estado Listar e imprimir	

Tarea	
Número Tarea: PRC01.10	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir secado.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Registrar secado según a datos solicitados <ul style="list-style-type: none"> ➤ Cantidad Procesada ➤ Porcentaje de humedad ➤ Impurezas ➤ Cantidad para despedrado II ➤ Observaciones ➤ Fecha proceso ➤ Hora inicio y fin ➤ estado Listar e imprimir	

Tarea	
Número Tarea: PRC01.11	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir Despedrado II
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Registrar Despedrado II según a datos solicitados <ul style="list-style-type: none"> ➤ Cantidad Procesada ➤ Piedrecillas ➤ Cantidad para clasificado ➤ Observaciones ➤ Fecha proceso ➤ Hora inicio y fin ➤ Estado Listar e imprimir	

Tarea	
Número Tarea: PRC01.12	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir Clasificado
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Registrar Clasificado según a datos solicitados <ul style="list-style-type: none"> ➤ Cantidad Procesada ➤ Extra ➤ Primera ➤ Segunda ➤ Tercera ➤ Impurezas ➤ Cantidad para selección óptica ➤ Observaciones ➤ Fecha proceso ➤ Hora inicio y fin ➤ Estado Listar e imprimir	

Tarea	
Número Tarea: PRC01.13	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir Selección de puntos negros.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Registrar Selección de puntos negros según a datos solicitados <ul style="list-style-type: none"> ➤ Cantidad quinua ➤ Quinua colores ➤ Impurezas ➤ Cantidad para envasado ➤ Observaciones ➤ Estado Listar e imprimir	

Tarea	
Número Tarea: PRC01.14	Número Historia: PRC01
Nombre Tarea:	Registrar, listar e imprimir Empaquetado.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Registrar Empaquetado según a datos solicitados <ul style="list-style-type: none"> ➤ Peso total ➤ Tipo paquete (sacos) ➤ Peso de paquetes ➤ Cantidad de paquetes ➤ Observaciones ➤ Estado Listar e imprimir	

Historia de Usuario			
Módulo: Procesamiento			
Número: PRC02	Usuario: Jefe de Planta		
Nombre Historia:	Gestionar datos Lotes de Venta.		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 7 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite gestionar datos de Lotes de Venta según a los siguientes datos solicitados:</p> <p>Registrar Lotes de Venta:</p> <ul style="list-style-type: none"> ➤ Venta ➤ Embarque ➤ Código de Lote de Venta ➤ Cantidad en Kg ➤ Peso de paquetes en Kg ➤ Saldo en Kg ➤ Estado <p>Listar Lotes de ventas</p> <p>Agregar Paquetes a un Lote de venta</p> <ul style="list-style-type: none"> ➤ Cantidad total en Kg ➤ Agregar Paquetes Procesados <ul style="list-style-type: none"> ○ Producto ○ Procesamiento ○ Tipo de paquete ○ Código de Etiqueta ○ Peso Paquete en kg ○ Cantidad empaquetada en Kg ○ Restantes en Kg. 			
Observaciones:			

Tarea	
Número Tarea: PRC02.1	Número Historia: PRC02
Nombre Tarea:	Diseñar Base de datos para registro Lotes de Venta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Inspecciones.	

Tarea	
Número Tarea: PRC02.2	Número Historia: PRC02
Nombre Tarea:	Diseñar Base de datos para registro paquetes procesados.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de paquetes procesados.	

Tarea	
Número Tarea: PRC02.3	Número Historia: PRC02
Nombre Tarea:	Crear interfaz gráficas para registrar Lotes de Venta.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Crear interfaz gráficas para registrar Lotes de Venta	

Tarea	
Número Tarea: PRC02.4	Número Historia: PRC02
Nombre Tarea:	Crear interfaz gráficas para registrar paquetes procesados.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de paquetes procesados.	

Tarea	
Número Tarea: PRC02.5	Número Historia: PRC02
Nombre Tarea:	Registrar Datos de Lotes de Venta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC02.6	Número Historia: PRC02
Nombre Tarea:	Registrar Datos de paquetes procesados.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	



Tarea	
Número Tarea: PRC02.7	Número Historia: PRC02
Nombre Tarea:	Listar Lotes de Venta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Listar Inspecciones según detalle de lote de venta, cantidad en Kg.	

Tarea	
Número Tarea: PRC02.8	Número Historia: PRC02
Nombre Tarea:	Listar Lotes de paquetes procesados.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Listar paquetes procesados según producto, peso y paquete.	

Historia de Usuario			
Módulo: Procesamiento			
Número: PRC03	Usuario: Jefe de Planta		
Nombre Historia:	Gestionar datos Lotes de Venta.		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
Descripción: Permite gestionar datos de Partes de Salida según a los siguientes datos solicitados: Registrar Partes de Salida: <ul style="list-style-type: none"> ➤ Lote venta ➤ Unidad de Transporte ➤ Unidad de medida ➤ Tipo de medida ➤ Tipo de Paquete ➤ Numero de paquete ➤ Peso Paquete ➤ Peso Neto ➤ Peso Bruto ➤ Fecha inicio producción ➤ Fecha envió Listar Partes de Salida			
Observaciones:			

Tarea	
Número Tarea: PRC03.1	Número Historia: PRC03
Nombre Tarea:	Diseñar Base de datos para registro de partes de salida.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de partes de salida.	

Tarea	
Número Tarea: PRC03.2	Número Historia: PRC03
Nombre Tarea:	Crear interfaz gráficas para registrar partes de salida.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de partes de salida.	

Tarea	
Número Tarea: PRC03.3	Número Historia: PRC03
Nombre Tarea:	Registro de Datos de partes de salida.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC03.4	Número Historia: PRC03
Nombre Tarea:	Listar partes de salida.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Listar partes de salida según a fecha de envío, número de paquetes.	



Historia de Usuario			
Módulo: Procesamiento			
Número: PRC04	Usuario: Jefe de Planta		
Nombre Historia:	Gestionar datos de Empresas de Transporte		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
Descripción: Permite gestionar datos Empresas de Transporte según a los siguientes datos solicitados: Registrar y Editar Empresas de Transporte: <ul style="list-style-type: none"> ➤ Nombre ➤ Tipo de persona (natural, jurídica) ➤ Dirección ➤ Fecha de Fundación ➤ Tipo de transportes (terrestre, marítimo) Listar Empresa de Transporte Detallar vehículos por empresa de transporte <ul style="list-style-type: none"> ➤ Registrar vehículos <ul style="list-style-type: none"> ○ Modelo ○ Marca ○ Placa ○ Soat ○ Descripción 			
Observaciones:			

Tarea	
Número Tarea: PRC04.1	Número Historia: PRC04
Nombre Tarea:	Diseñar Base de datos para registro de Empresas de Transporte.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Empresas de Transporte.	

Tarea	
Número Tarea: PRC04.2	Número Historia: PRC04
Nombre Tarea:	Diseñar Base de datos para registro de Unidad de transporte.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Unidad de Transporte según a una determinada empresa.	

Tarea	
Número Tarea: PRC04.3	Número Historia: PRC04
Nombre Tarea:	Crear interfaz gráficas para registrar Empresas de Transporte.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos Empresas de Transporte	

Tarea	
Número Tarea: PRC04.4	Número Historia: PRC04
Nombre Tarea:	Crear interfaz gráficas para registrar Unidad de Transporte
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos Unidad de Transporte	

Tarea	
Número Tarea: PRC04.5	Número Historia: PRC04
Nombre Tarea:	Registro de Datos de Empresas de Transporte.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC04.6	Número Historia: PRC04
Nombre Tarea:	Registro de Datos de Unidad de Transporte.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: PRC04.7	Número Historia: PRC04
Nombre Tarea:	Listar Empresas de Transporte.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Listar Empresas de Transporte según tipo de transporte, nombre de la empresa y detallar unidades.	

Tarea	
Número Tarea: PRC04.8	Número Historia: PRC04
Nombre Tarea:	Listar Unidades de Transportes.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Listar Unidades de Transporte según a determinada empresa (marca, modelo, tipo empresa)	





1. Historia de Usuario

1.1. Segunda Iteración: Módulo Comercialización

Historias de Usuario: Comercialización

Historia de Usuario			
Módulo: Comercialización			
Número: COM01	Usuario: Asistente Administrativo, Gerencia		
Nombre Historia:	Gestionar datos de Clientes		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
Descripción: Permite gestionar datos de Clientes según a los siguientes datos solicitados: Registrar y Editar Clientes: <ul style="list-style-type: none"> ➤ Nombre ➤ Apellido paterno ➤ Apellido materno ➤ Tipo persona(natural, jurídica) ➤ Dirección Listar Clientes			
Observaciones:			

		Tarea
Número Tarea: COM01.1	Número Historia: COM01	
Nombre Tarea:	Diseñar Base de datos para Clientes.	
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5	
Programador Responsable: Equipo G-1		
Descripción: Diseño de base de datos para el registro de Clientes		

		Tarea
Número Tarea: COM01.2	Número Historia: COM01	
Nombre Tarea:	Crear interfaz gráficas para registrar Clientes	
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5	
Programador Responsable: Equipo G-1		
Descripción: Se hace el diseño de formularios en el que se introducirán datos de un Cliente		



Tarea	
Número Tarea: COM01.3	Número Historia: COM01
Nombre Tarea:	Registrar Datos de Cliente.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: COM01.4	Número Historia: COM01
Nombre Tarea:	Editar Datos de Cliente.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos modificados en el formulario recuperado con datos anteriores.	

Tarea	
Número Tarea: COM01.5	Número Historia: COM01
Nombre Tarea:	Listar Clientes.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Listar Clientes según a nombre, dirección.	



Historia de Usuario			
Módulo: Comercialización			
Número: COM02	Usuario: Asistente Administrativo, Gerencia		
Nombre Historia:	Gestionar datos de Venta		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 10 días	Iteración Asignada: 3		
Programador Responsable: Equipo G-1			
Descripción: Permite gestionar datos de Venta según a los siguientes datos solicitados: Registrar Ventas: <ul style="list-style-type: none"> ➤ Cliente ➤ Código de venta ➤ Pesos de sacos ➤ Productos <ul style="list-style-type: none"> ○ Producto ○ Cantidad ○ Saldo ○ Precio Unitario ○ Precio Total ➤ Cantidad vendida ➤ Saldo ➤ Precio total ➤ Saldo ➤ Estado Listar Ventas Detallar una venta (productos vendidos, embarques, desembolsos, documentos de venta) <ul style="list-style-type: none"> ➤ Agrega embarque ➤ Generar Comprobante Agregar Documentos de venta			
Observaciones:			

Tarea	
Número Tarea: COM02.1	Número Historia: COM02
Nombre Tarea:	Diseñar Base de datos para Ventas.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de ventas	

Tarea	
Número Tarea: COM02.2	Número Historia: COM02
Nombre Tarea:	Diseñar Base de datos para Embarque.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de Embarque	



Tarea	
Número Tarea: COM02.3	Número Historia: COM02
Nombre Tarea:	Diseñar Base de datos para documentos de venta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Diseño de base de datos para el registro de documentos de venta	

Tarea	
Número Tarea: COM02.4	Número Historia: COM02
Nombre Tarea:	Crear interfaz gráficas para registrar Ventas.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de una Venta.	

Tarea	
Número Tarea: COM02.5	Número Historia: COM02
Nombre Tarea:	Crear interfaz gráficas para registrar documentos de ventas
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de documentos de ventas <ul style="list-style-type: none"> • Tipo de documento • Código de documento • Fecha de emisión • Descripción Archivo(selección archivo)	

Tarea	
Número Tarea: COM02.6	Número Historia: COM02
Nombre Tarea:	Crear interfaz gráficas para registrar Embarque.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Se hace el diseño de formularios en el que se introducirán datos de un Embarque. <ul style="list-style-type: none"> • Unidad de transporte • Código • Fecha embarque • Estado • Detalle de lote de venta • Código de venta • Cantidad en Kg • Peso de paquetes • Saldo 	

Tarea	
Número Tarea: COM02.7	Número Historia: COM02
Nombre Tarea:	Registrar Datos de Ventas.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: COM02.8	Número Historia: COM02
Nombre Tarea:	Registrar Datos de Embarque.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Se hace guardado de datos introducidos en el formulario.	

Tarea	
Número Tarea: COM02.9	Número Historia: COM02
Nombre Tarea:	Listar ventas
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Se hace el listado de ventas según cliente, detalle venta, cantidad en Kg, fecha de envío, detalles de una venta	

Tarea	
Número Tarea: COM02.10	Número Historia: COM02
Nombre Tarea:	Generar Comprobantes
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 7
Programador Responsable: Equipo G-1	
Descripción: Se genera la factura según: <ul style="list-style-type: none"> • Tipo de documento • Serie comprobante • Numero de comprobante • Fecha de comprobante • Monto • Saldo e venta 	

Tarea	
Número Tarea: COM02.11	Número Historia: COM02
Nombre Tarea:	Listar comprobantes de venta
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 5
Programador Responsable: Equipo G-1	
Descripción: Listar comprobantes según a tipo de documento, código, fecha de emisión y descripción.	

Tarea	
Número Tarea: COM02.12	Número Historia: COM02
Nombre Tarea:	registrar documentos de venta
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Se hace el guardado de datos introducidos en el formulario de documentos de venta.	

Tarea	
Número Tarea: COM02.13	Número Historia: COM02
Nombre Tarea:	Detallar resumen de venta
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Mostrar en un cuadro el resumen de venta según: <ul style="list-style-type: none"> • Numero de venta, Fecha de envió. • Cantidad en Kilos, Saldo en Kg. • Precio total de venta, Desembolsado, Saldo en soles. 	



1. Historia de Usuario

1.1. Segunda Iteración: Módulo Control Documentario

Historias de Usuario: Control Documentario

Historia de Usuario			
Módulo: Control Documentario			
Número: CDO01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Carta		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Documento Carta, según a los siguientes datos solicitados: Registrar y Editar Documento Carta: <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Lugar. ➤ Fecha del Documento. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Destinatario. ➤ Asunto. ➤ Referencia. ➤ Saludo. ➤ Contenido. ➤ Despedida. ➤ Adjuntos Listar Documentos Carta, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CDO01.1	Número Historia: CDO01
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Carta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Carta.	

Tarea	
Número Tarea: CDO01.2	Número Historia: CDO01
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Carta.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Carta.	

Tarea	
Número Tarea: CDO01.3	Número Historia: CDO01
Nombre Tarea:	Registro y Edición de datos de Documento Carta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO01.4	Número Historia: CDO01
Nombre Tarea:	Listar registro de Documento Carta.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Carta, con las acciones respectivas.	

Tarea	
Número Tarea: CDO01.5	Número Historia: CDO01
Nombre Tarea:	Generar Documento Carta en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Diseñar el formato del documento, para que genera un documento en formato*.PDF.	



Historia de Usuario			
Módulo: Control Documentario			
Número: CDO02	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Informe		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Documento Carta, según a los siguientes datos solicitados: Registrar y Editar Documento Carta: <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Lugar. ➤ Fecha del Documento. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Destinatario. ➤ Asunto. ➤ Referencia. ➤ Saludo. ➤ Contenido. ➤ Despedida. ➤ Adjuntos Listar Documentos Carta, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CDO02.1	Número Historia: CDO02
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Informe.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Informe.	

Tarea	
Número Tarea: CDO02.2	Número Historia: CDO02
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Informe.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Informe.	

Tarea	
Número Tarea: CDO02.3	Número Historia: CDO02
Nombre Tarea:	Registro y Edición de datos de Documento Informe.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO02.4	Número Historia: CDO02
Nombre Tarea:	Listar registro de Documento Informe.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Informe, con las acciones respectivas.	

Tarea	
Número Tarea: CDO02.5	Número Historia: CDO02
Nombre Tarea:	Generar Documento Informe en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Diseñar el formato del documento, para que genera un documento en formato*.PDF.	



Historia de Usuario			
Módulo: Control Documentario			
Número: CDO03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Memorándums		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite Registrar, Editar, Listar Documento Memorándums, según a los siguientes datos solicitados: Registrar y Editar Documento Memorándums:</p> <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Lugar. ➤ Fecha del Documento. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Destinatario. ➤ Asunto. ➤ Referencia. ➤ Saludo. ➤ Contenido. ➤ Despedida. ➤ Adjuntos <p>Listar Documentos Memorándums, mostrando los datos relevantes.</p>			
Observaciones:			

Tarea	
Número Tarea: CDO03.1	Número Historia: CDO03
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Memorándums.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Memorándums.	

Tarea	
Número Tarea: CDO03.2	Número Historia: CDO03
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Memorándums.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Memorándums.	

Tarea	
Número Tarea: CDO03.3	Número Historia: CDO03
Nombre Tarea:	Registro y Edición de datos de Documento Memorándums.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO03.4	Número Historia: CDO03
Nombre Tarea:	Listar registro de Documento Memorándums.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Memorándums, con las acciones respectivas.	

Tarea	
Número Tarea: CDO03.5	Número Historia: CDO03
Nombre Tarea:	Generar Documento Memorándums en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Se Diseña el formato del documento, para que genera un documento en formato *.PDF.	



Historia de Usuario			
Módulo: Control Documentario			
Número: CDO01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Oficio		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Documento Oficio, según a los siguientes datos solicitados: Registrar y Editar Documento Oficio: <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Destinatario. ➤ Lugar ➤ Fecha del Documento. ➤ Asunto. ➤ Saludo. ➤ Contenido. ➤ Despedida. ➤ Adjuntos Listar Documentos Oficio, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CDO04.1	Número Historia: CDO04
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Oficio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Oficio.	

Tarea	
Número Tarea: CDO04.2	Número Historia: CDO04
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Oficio.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Oficio.	

Tarea	
Número Tarea: CDO04.3	Número Historia: CDO04
Nombre Tarea:	Registro y Edición de datos de Documento Oficio.
Tipo de Tarea: Desarrollo	Tiempo Estimado (Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO04.4	Número Historia: CDO04
Nombre Tarea:	Listar registro de Documento Oficio.
Tipo de Tarea: Desarrollo	Tiempo Estimado (Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Oficio, con las acciones respectivas.	

Tarea	
Número Tarea: CDO04.5	Número Historia: CDO04
Nombre Tarea:	Generar Documento Oficio en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado (Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se Diseña el formato del documento, para que genera un documento en formato*.PDF.	



Historia de Usuario			
Módulo: Control Documentario			
Número: CDO01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Solicitud		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Documento Solicitud, según a los siguientes datos solicitados: Registrar y Editar Documento Solicitud: <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Destinatario. ➤ Lugar. ➤ Fecha del Documento. ➤ Asunto. ➤ Presentación. ➤ Contenido. ➤ Despedida. ➤ Adjuntos Listar Documentos Solicitud, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CDO05.1	Número Historia: CDO05
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Solicitud.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Solicitud.	

Tarea	
Número Tarea: CDO05.2	Número Historia: CDO05
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Solicitud.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Solicitud.	

Tarea	
Número Tarea: CDO05.3	Número Historia: CDO05
Nombre Tarea:	Registro y Edición de datos de Documento Solicitud.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO05.4	Número Historia: CDO05
Nombre Tarea:	Listar registro de Documento Solicitud.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Solicitud, con las acciones respectivas.	

Tarea	
Número Tarea: CDO05.5	Número Historia: CDO05
Nombre Tarea:	Generar Documento Solicitud en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se Diseña el formato del documento, para que genera un documento en formato*.PDF.	



Historia de Usuario			
Módulo: Control Documentario			
Número: CDO06	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos del Documento Requerimiento.		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 5		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Documento Requerimiento, según a los siguientes datos solicitados: Registrar y Editar Documento Requerimiento: <ul style="list-style-type: none"> ➤ Nombre del Año. ➤ Fecha Actual. ➤ Número del Documento. ➤ Nombres del Remitente. ➤ Nombres del Grupo. ➤ Fecha del Documento. ➤ Asunto. ➤ Contenido. ➤ Adjuntos ➤ Observaciones. Listar Documentos Requerimiento, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CDO06.1	Número Historia: CDO06
Nombre Tarea:	Diseñar Base de Datos para registro de Documento Requerimiento.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Documento Requerimiento.	

Tarea	
Número Tarea: CDO06.2	Número Historia: CDO06
Nombre Tarea:	Crear interfaz gráficas para registrar Documento Requerimiento.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Documento Requerimiento.	

Tarea	
Número Tarea: CDO06.3	Número Historia: CDO06
Nombre Tarea:	Registro y Edición de datos de Documento Requerimiento.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CDO06.4	Número Historia: CDO06
Nombre Tarea:	Listar registro de Documento Requerimiento.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Documento Requerimiento, con las acciones respectivas.	

Tarea	
Número Tarea: CDO06.5	Número Historia: CDO06
Nombre Tarea:	Generar Documento Requerimiento en formato *.PDF.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Se Diseña el formato del documento, para que genera un documento en formato *.PDF.	



1. Historia de Usuario

1.1. Segunda Iteración: Módulo Caja

Historias de Usuario: Caja

Historia de Usuario			
Módulo: Caja			
Número: CJA01	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Ingreso		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 6		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Ingresos, según a los siguientes datos solicitados: Registrar y Editar Ingresos: <ul style="list-style-type: none"> ➤ Número de Ingreso. ➤ Serie. ➤ Fecha del Comprobante. ➤ Concepto. ➤ Moneda. ➤ Monto. ➤ Descripción. Listar registro de Ingresos, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CJA01.1	Número Historia: CJA01
Nombre Tarea:	Diseñar Base de Datos para registro de Ingreso.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Ingreso.	

Tarea	
Número Tarea: CJA01.2	Número Historia: CJA01
Nombre Tarea:	Crear interfaz gráficas para registrar Ingresos.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Ingresos.	



Tarea	
Número Tarea: CJA01.3	Número Historia: CJA01
Nombre Tarea:	Registro y Edición de datos de Ingreso.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CJA01.4	Número Historia: CJA01
Nombre Tarea:	Listar registro de Ingresos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro Ingresos.	

Historia de Usuario			
Módulo: Caja			
Número: CJA02	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Egreso		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 6		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Egresos, según a los siguientes datos solicitados: Registrar y Editar Egresos: <ul style="list-style-type: none"> ➤ Número de Egreso. ➤ Serie. ➤ Fecha del Comprobante. ➤ Concepto. ➤ Moneda. ➤ Monto. ➤ Descripción. Listar registro de Egresos, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CJA02.1	Número Historia: CJA02
Nombre Tarea:	Diseñar Base de Datos para registro de Egreso.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Egreso.	

Tarea	
Número Tarea: CJA02.2	Número Historia: CJA02
Nombre Tarea:	Crear interfaz gráficas para registrar Egresos.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Egresos.	

Tarea	
Número Tarea: CJA02.3	Número Historia: CJA02
Nombre Tarea:	Registro y Edición de datos de Egreso.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CJA02.4	Número Historia: CJA02
Nombre Tarea:	Listar registro de Egreso.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro Egresos.	



Historia de Usuario			
Módulo: Caja			
Número: CJA03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Pago al Personal		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 6		
Programador Responsable: Equipo G-1			
<p>Descripción: Permite Registrar, Listar Pagos al Personal, según a los siguientes datos solicitados:</p> <p>Registrar Pago al Personal:</p> <ul style="list-style-type: none"> ➤ Opción de Adelanto. ➤ Comprobante de Pago. ➤ Serie. ➤ Número. ➤ Fecha Comprobante. ➤ Sueldo. ➤ Saldo. ➤ Período Inicio. ➤ Período Fin. ➤ Descripción. ➤ Observaciones. <p>Listar registro de Ingresos, mostrando los datos relevantes.</p>			
Observaciones:			

Tarea	
Número Tarea: CJA03.1	Número Historia: CJA03
Nombre Tarea:	Diseñar Base de Datos para registro de Pago al Personal.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Pago al Personal.	

Tarea	
Número Tarea: CJA03.2	Número Historia: CJA03
Nombre Tarea:	Crear interfaz gráficas para registrar Pago al Personal.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Pago al Personal.	



Tarea	
Número Tarea: CJA03.3	Número Historia: CJA03
Nombre Tarea:	Registro de datos de Pago al Personal.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CJA03.4	Número Historia: CJA03
Nombre Tarea:	Listar registro de Pago al Personal.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro Pago al Personal.	

Historia de Usuario			
Módulo: Caja			
Número: CJA04	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar datos de Aporte de Socio.		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 4 días	Iteración Asignada: 6		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Listar Aporte de Socios, según a los siguientes datos solicitados: Registrar Aporte de Socio: <ul style="list-style-type: none"> ➤ Serie. ➤ Número. ➤ Fecha Comprobante. ➤ Nombre del Socio Productor ➤ Fecha correspondiente al aporte. ➤ Monto. ➤ Descripción. ➤ Observaciones. Listar registro de Aporte de Socios, mostrando los datos relevantes.			
Observaciones:			

Tarea	
Número Tarea: CJA04.1	Número Historia: CJA04
Nombre Tarea:	Diseñar Base de Datos para registro de Aporte de Socio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Aporte de Socio.	

Tarea	
Número Tarea: CJA04.2	Número Historia: CJA04
Nombre Tarea:	Crear interfaz gráficas para registrar Aporte de Socio.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Aporte de Socio.	

Tarea	
Número Tarea: CJA04.3	Número Historia: CJA04
Nombre Tarea:	Registro de datos de Aporte de Socio.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: CJA04.4	Número Historia: CJA04
Nombre Tarea:	Listar registro de Aporte de Socios.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Aporte de Socios.	



Historia de Usuario			
Módulo: Caja			
Número: CJA05	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Generar Reporte de Caja		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 6 días	Iteración Asignada: 6		
Programador Responsable: Equipo G-1			
Descripción: Permite Generar Reporte, según a los siguientes datos solicitados: Generar Reporte de Caja: <ul style="list-style-type: none"> ➤ Tipo de Reporte por: Egresos, Ingresos y Pagos Personales ➤ Selección de Período (Día, Mes, Año, Intervalo de Fechas). Fecha Seleccionada.			
Observaciones:			

Tarea	
Número Tarea: CJA05.1	Número Historia: CJA05
Nombre Tarea:	Realizar el Query-SQL para la consulta a la Base de Datos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Realizar el script para la consulta a la base de datos y generar el reporte solicitado.	

Tarea	
Número Tarea: CJA05.2	Número Historia: CJA05
Nombre Tarea:	Crear interfaz gráficas del reporte de Acopio.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del modelo de reporte.	



1. Historia de Usuario

1.1. Segunda Iteración: Módulo Gerencia

Historias de Usuario: Gerencia

Historia de Usuario			
Módulo: Gerencia			
Número: GER01	Usuario: Gerente de Producción		
Nombre Historia:	Gestionar datos de Precio de Producción		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 8 días	Iteración Asignada: 7		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Precio de Producción, según a los siguientes datos solicitados: Registrar Precio de Producción: <ul style="list-style-type: none"> ➢ Producto. ➢ Moneda. ➢ Precio (Kg). Listar Precios de Producción mostrando los datos que lo conforman.			
Observaciones:			

Tarea	
Número Tarea: GER01.1	Número Historia: GER01
Nombre Tarea:	Diseñar Base de Datos para registro de Precio de Producción.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño de base de datos para el Registro de Precio de Producción.	

Tarea	
Número Tarea: GER01.2	Número Historia: GER01
Nombre Tarea:	Crear interfaz gráficas para registrar Precio de Producción.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos del precio de Producción.	

Tarea	
Número Tarea: GER01.3	Número Historia: GER01
Nombre Tarea:	Registro y Edición de datos de Precio de Producción.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: GER01.4	Número Historia: GER01
Nombre Tarea:	Listar registro de Precio de Producción.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Precio de Producción.	

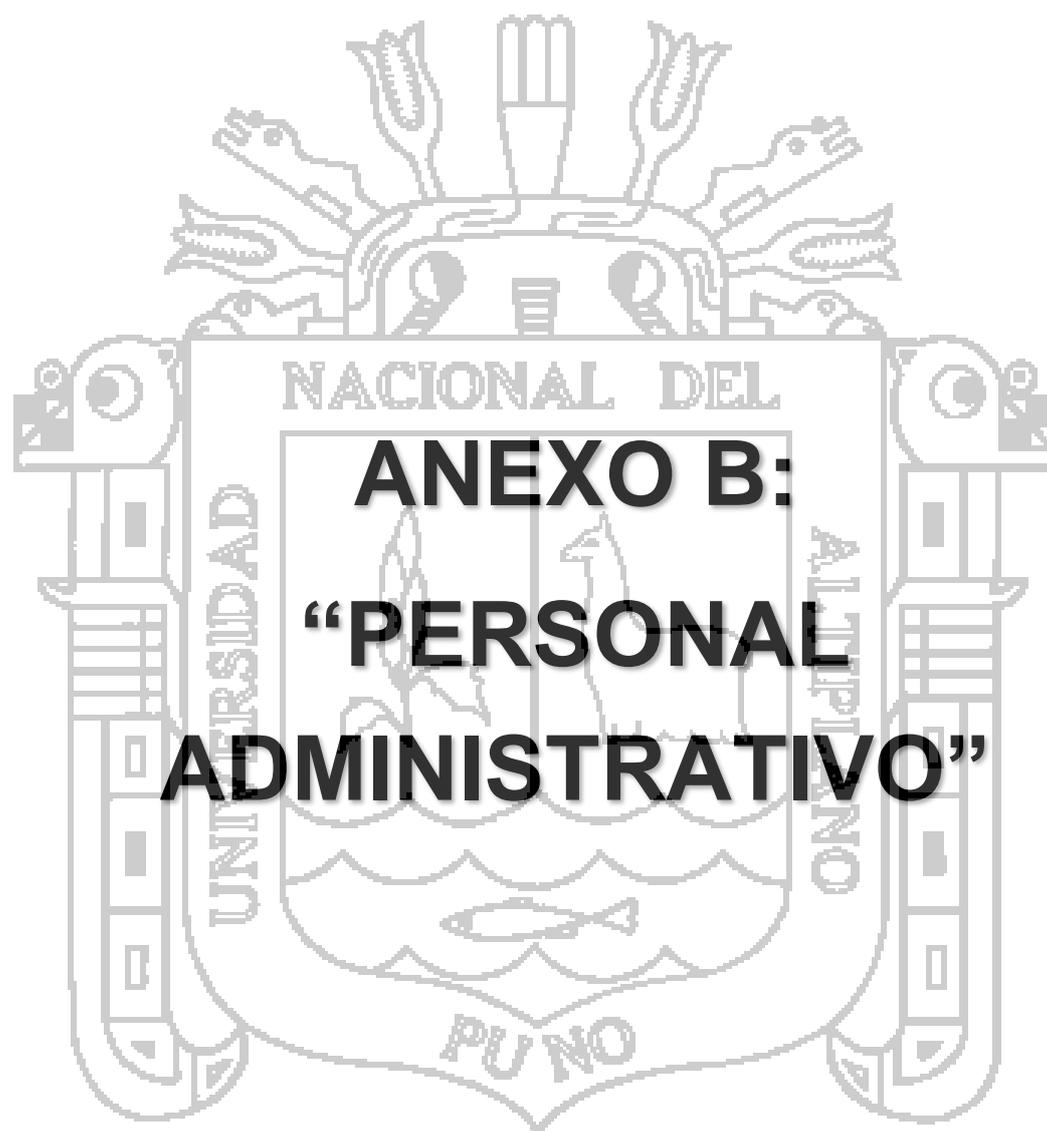
Historia de Usuario			
Módulo: Gerencia			
Número: GER02	Usuario: Gerente de Producción		
Nombre Historia:	Gestionar datos de Precio de Comercialización		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 8 días	Iteración Asignada: 7		
Programador Responsable: Equipo G-1			
Descripción: Permite Registrar, Editar, Listar Precios de Comercialización, según a los siguientes datos solicitados: Registrar Precio de Comercialización: <ul style="list-style-type: none"> ➤ Producto. ➤ Moneda. ➤ Precio (Kg). Listar Precios de Comercialización mostrando los datos que lo conforman.			
Observaciones:			

Tarea	
Número Tarea: GER02.1	Número Historia: GER02
Nombre Tarea:	Diseñar Base de Datos para registro de Precio de Comercialización.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño de base de datos para el Registro de Precio de Comercialización.	

Tarea	
Número Tarea: GER02.2	Número Historia: GER02
Nombre Tarea:	Crear interfaz gráficas para registrar Precio de Comercialización.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos del Precio de Comercialización.	

Tarea	
Número Tarea: GER02.3	Número Historia: GER02
Nombre Tarea:	Registro y Edición de datos de Precio de Comercialización.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: GER02.4	Número Historia: GER02
Nombre Tarea:	Listar registro de Precio de Comercialización.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Precio de Comercialización.	



1. Historia de Usuario

1.1. Segunda Iteración: Módulo Personal Administrativo

Historias de Usuario: Personal Administrativo

Historia de Usuario			
Módulo: Personal Administrativo			
Número: APA01	Usuario: Asistente de gerencia		
Nombre Historia:	Gestionar datos del Personal Administrativo		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 10 días	Iteración Asignada: 8		
Programador Responsable: Equipo G-1			
<p>Descripción: Descripción: Permite Registrar, Editar, Listar Personal Administrativo, según a los siguientes datos solicitados:</p> <p>Registrar y Editar Personal Administrativo:</p> <ul style="list-style-type: none"> ➤ Código. ➤ Nombre. ➤ Apellido Paterno. ➤ Apellido Materno. ➤ Sueldo. ➤ Usuario. ➤ Contraseña. ➤ Confirmación de Contraseña. ➤ Tipo de Persona (Natural, Jurídica). ➤ Sexo. ➤ Dirección. ➤ Educación. ➤ Grupo. <p>Listar Personal Administrativo mostrando código, nombre, apellido paterno, apellido materno.</p>			
Observaciones:			

Tarea	
Número Tarea: APA01.1	Número Historia: APA01
Nombre Tarea:	Diseñar Base de Datos para registro de Personal Administrativo.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realizar el Diseño de base de datos para el Registro de Personal Administrativo.	

Tarea	
Número Tarea: APA01.2	Número Historia: APA01
Nombre Tarea:	Crear interfaz gráficas para registrar Personal Administrativo.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 3
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del formulario en el que se ingresaran datos de Personal Administrativo.	

Tarea	
Número Tarea: APA01.3	Número Historia: APA01
Nombre Tarea:	Registro y Edición de datos de Personal Administrativo.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 10
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el guardado de datos ingresados en el formulario.	

Tarea	
Número Tarea: APA01.4	Número Historia: APA01
Nombre Tarea:	Listar registro de Personal Administrativo.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Listar registro de Personal Administrativo, con las acciones respectivas.	

Historia de Usuario			
Módulo: Personal Administrativo			
Número: APA02	Usuario: Todos		
Nombre Historia:	Generar Reporte del Personal Administrativo		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días		Iteración Asignada: 8	
Programador Responsable: Equipo G-1			
Descripción: Permite Reporte de Personal Administrativo en documento *.PDF.			
Observaciones:			



Tarea	
Número Tarea: APA02.1	Número Historia: APA02
Nombre Tarea:	Realizar el Query-SQL para la Consulta a la Base de Datos.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 2
Programador Responsable: Equipo G-1	
Descripción: Realzar el script para la consulta a la base de datos y generar el reporte solicitado.	

Tarea	
Número Tarea: APA02.2	Número Historia: APA02
Nombre Tarea:	Crear interfaz gráficas del reporte de Personal Administrativo.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 4
Programador Responsable: Equipo G-1	
Descripción: Realizar el diseño del modelo de reporte.	

Tarea	
Número Tarea: APA02.3	Número Historia: APA02
Nombre Tarea:	Generar formato de reporte de Personal Administrativo.
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 6
Programador Responsable: Equipo G-1	
Descripción: Se Realiza el formato del reporte solicitado.	

Historia de Usuario			
Módulo: Personal Administrativo			
Número: APA03	Usuario: Equipo técnico, Gerente de Producción		
Nombre Historia:	Gestionar acceso al sistema y perfiles de usuario		
Prioridad:	Alta	Riesgo en desarrollo:	Alta
Tiempo Estimado: 3 días		Iteración Asignada: 8	
Programador Responsable: Equipo G-1			
Descripción: Acceso al sistema según rol asignado			
Observaciones:			

Tarea	
Número Tarea: APA03.1	Número Historia: APA03
Nombre Tarea:	Crear interfaz gráfica del diseño de login.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Crear interfaz gráfica del diseño de login.	

Tarea	
Número Tarea: APA03.2	Número Historia: APA03
Nombre Tarea:	Desarrollar login sencillo.
Tipo de Tarea: Diseño	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: desarrollar el login.	

Tarea	
Número Tarea: APA03.3	Número Historia: APA03
Nombre Tarea:	Restringir acceso al sistema según roles
Tipo de Tarea: Desarrollo	Tiempo Estimado(Horas): 8
Programador Responsable: Equipo G-1	
Descripción: Restringir acceso al sistema según roles.	



1. Pruebas de Aceptación

1.1. Pruebas: Módulo Producción

Caso de Prueba: MÓDULO DE PRODUCCIÓN	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: PRO01
Nombre de caso de prueba: Gestionar datos de Asociación y Sector	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Asociación o Sector", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Asociación y Sector". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.2. Pruebas: Módulo Acopio y Almacén

Caso de Prueba: MÓDULO DE ACOPIO Y ALMACÉN	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: ACA01
Nombre de caso de prueba: Gestionar datos de Acopio	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Acopio", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Acopio". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.3. Pruebas: Módulo Procesamiento

Caso de Prueba: MÓDULO DE PROCESAMIENTO	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: PRC01
Nombre de caso de prueba: Gestionar datos de Procesamiento.	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Procesamiento", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Procesamiento". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.4. Pruebas: Módulo Comercialización

Caso de Prueba: MÓDULO DE COMERCIALIZACIÓN	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: COM01
Nombre de caso de prueba: Gestionar datos de Clientes.	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Clientes", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Cliente". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.5. Pruebas: Módulo Control Documentario

Caso de Prueba: MÓDULO CONTROL DOCUMENTARIO	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: COM01
Nombre de caso de prueba: Gestionar datos del Documento Carta.	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Carta", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Cartas". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.6. Pruebas: Módulo Caja

Caso de Prueba: MÓDULO CAJA	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: CJA01
Nombre de caso de prueba: Gestionar datos de Ingreso.	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Ingresos", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Ingresos". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.7. Pruebas: Módulo Gerencia

Caso de Prueba: MÓDULO GERENCIA	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: GER01
Nombre de caso de prueba: Gestionar datos de Precio de Producción.	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Precios de Producción", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Precios de Producción". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	

1.8. Pruebas: Módulo Asistencia Personal Administrativo

Caso de Prueba: MÓDULO PERSONAL ADMINISTRATIVO	
Numero de Caso de Prueba: 01	Numero Historia de Usuario: APA01
Nombre de caso de prueba: Gestionar datos del Personal Administrativo	
Descripción: El usuario una vez haya ingresado en el sistema seleccionará la opción del menú "Personal", internamente se procesarán los datos y si no hay ningún error de procesado. Se introducirán los datos en la base de datos.	
Condiciones de ejecución: El usuario deberá estar dado de alta en el sistema.	
Entradas: <ul style="list-style-type: none"> - El usuario introducirá su login y password. - Del menú principal seleccionará "Personal". - Introducir datos en el formulario. - Procesar datos internamente para ser registrado en la base de datos. - El proceso de introducción se considera como finalizado. 	
Resultado Esperado: Tras la introducción de datos, si el procesado ha sido correcto, en la base de datos aparecerán los datos de los nuevos registros.	
Evaluación: Prueba satisfactoria	



ANEXO D: ENCUESTA

La presente encuesta, que se le presenta estimado usuario es con la finalidad de verificar la satisfacción que usted ha alcanzado tras la puesta en funcionamiento del nuevo sistema, además esto corroborará a que el ejecutor de este proyecto realice el sustento adecuado de la aceptación del nuevo sistema.

Nota: Lea cuidadosamente y marque con una "X" la alternativa que usted considere.

1. **En cuanto a la confiabilidad del Sistema ¿Cuál es su apreciación con el manejo de información en sus procesos?**
 - a. Excelente
 - b. Bueno
 - c. Regular
 - d. Malo
 - e. Deficiente

2. **Respecto a la Facilidad del Uso del Sistema ¿Cuál es su apreciación?**
 - a. Excelente
 - b. Bueno
 - c. Regular
 - d. Malo
 - e. Deficiente

3. **Según usted, ¿Cómo califica el tiempo para realizar un proceso?**
 - a. Excelente
 - b. Bueno
 - c. Regular
 - d. Malo
 - e. Deficiente

4. **Según usted, como califica, acerca de que los costos de operación son menores con el uso del Sistema.**
 - a. Excelente
 - b. Bueno
 - c. Regular
 - d. Malo
 - e. Deficiente

5. **Con la Implementación del Sistema, ¿Cuál es la calificación, respecto a un mejor control?**
 - a. Excelente
 - b. Bueno
 - c. Regular
 - d. Malo
 - e. Deficiente