

ESCUELA POLITÉCNICA DE EJÉRCITO

DPTO. DE CIENCIAS DE LA COMPUTACIÓN

CARRERA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

DESARROLLO DE UN SISTEMA WEB PARA EL MANEJO DE INVENTARIO DE MULTIBODEGAS, GESTIÓN CONTABLE, Y CREACIÓN DE UN REPOSITORIO DIGITAL DE INFORMES TÉCNICOS, UTILIZANDO DSPACE, TECNOLOGÍA PRIMEFACES Y HERRAMIENTAS OPEN SOURCE, PARA LA EMPRESA CROSSTRONIK CIA. LTDA.

Previa a la obtención del Título de:

INGENIERO EN SISTEMAS E INFORMÁTICA

POR:

**DIANA CATALINA CRUZ CAIZA
CHRISTIAN XAVIER ZHAMUNGUI OVIEDO**

Sangolquí, Junio de 2013

AUTORIZACIÓN

Nosotros Diana Catalina Cruz Caiza, Christian Xavier Zhamungui Oviedo autorizamos a la Escuela Politécnica del Ejército a que publique en el repositorio digital de la biblioteca Alejandro Segovia el presente proyecto de tesis, así como también los materiales y documentos relacionados a la misma.

Sangolquí, Junio de 2013

Cruz Caiza Diana Catalina

Zhamungui Oviedo Christian Xavier

DECLARACIÓN

Nosotros Diana Catalina Cruz Caiza, Christian Xavier Zhamungui Oviedo, declaramos que el presente trabajo es de nuestra autoría; que no ha sido previamente presentado para ningún grado o calificación personal y que he consultado las referencias bibliográficas que se incluyen en el documento.

La Escuela Politécnica del Ejército puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual por su reglamento y por la normativa institucional vigente.

Sangolquí, Junio de 2013

CERTIFICACIÓN

Certifico que el presente trabajo fue realizado en su totalidad por los Srs. CRUZ CAIZA DIANA CATALINA, ZHAMUNGUI OVIEDO CHRISTIAN XAVIER como requerimiento parcial a la obtención del título de INGENIEROS EN SISTEMAS E INFORMÁTICA

Sangolquí, Junio de 2013

ING. ÁLEX JIMÉNEZ.

DEDICATORIA

Dedico este trabajo
a mi familia,
a Crosstronik, la empresa que con tanto
amor mis hermanos han formado y,
¿por qué no?, también
a la vida, un juego fuerte y alucinante,
lleno de riesgos,
que me ha enseñado a caer y
levantarme.

Diana Catalina Cruz Caiza

DEDICATORIA

Dedico el presente trabajo a toda mi familia, por brindarme todo su amor y darme valor en todo momento; también a los hermanos de mi comunidad Neocatecumenal porque con sus oraciones me mantienen fuerte en la consecución de mis metas.

Christian Xavier Zhamungui Oviedo

AGRADECIMIENTOS

Tiempo de nacer, tiempo de morir,
tiempo de plantar, tiempo de arrancar la planta,
tiempo de matar, tiempo de curar,
tiempo de destruir, tiempo de construir,
tiempo de llorar, tiempo de reír,
tiempo de gemir, tiempo de bailar,
tiempo de tirar piedras, tiempo de recoger piedras,
tiempo de abrazar, tiempo de separar,
tiempo de buscar, tiempo de perder,
tiempo de guardar, tiempo de tirar,
tiempo de rasgar, tiempo de coser,
tiempo de callar, tiempo de hablar,
tiempo de amar, tiempo de odiar,
tiempo de guerra, tiempo de paz.

(11 Minutos, Paulo Coelho)

Hoy, estoy convencida de que lo vivido durante este proceso de aprendizaje y formación, ha sido una verdadera experiencia. Vivirlo no habría sido posible sin el apoyo incondicional de Dios, de mis padres y hermanos, de mi ojitos de luna y su padre, de mis amigos sinceros. A ellos mi agradecimiento infinito por haber sabido ayudarme y por haber creído en mí.

Diana Catalina Cruz Caiza

AGRADECIMIENTOS

A Dios por regalarme la vida, el discernimiento y la sabiduría para la consecución de mis metas; a mis padres por ser mi apoyo incondicional en todo momento y guiarme siempre con su ejemplo, a mis hermanos Daniel y Jessica, por darme valor en los momentos más difíciles; a mi compañera Diana Cruz por compartir conmigo tantas noches de desvelo en fin de lograr nuestro objetivo; a nuestro Director y Co-director los ingenieros Alex Jiménez y Freddy Dueñas respectivamente, por apoyarnos en la dirección del presente trabajo; y a todas las personas que directa o indirectamente colaboraron con este trabajo.

Christian Xavier Zhamungui Oviedo

TABLA DE CONTENIDOS

RESUMEN	1
CAPÍTULO 1: INTRODUCCIÓN.....	2
1.1 Antecedentes	2
1.2 Planteamiento del Problema.....	3
1.3 Objetivos.....	5
1.3.1 Objetivo general	5
1.3.2 Objetivos específicos	5
1.4 Justificación	6
1.5 Alcance	8
1.6 Factibilidad	10
1.6.1 Factibilidad Técnica.....	10
1.6.2 Factibilidad Económica	11
1.6.3 Factibilidad Operativa.....	12
1.6.4 Factibilidad Operacional.....	12
1.6.5 Factibilidad Tecnológica	13
1.6.6 Factibilidad Legal	14
1.7 Metodología.....	14
CAPÍTULO 2: MARCO TEÓRICO.....	15
2.1 Sistemas en Ambientes Web	15
2.1.1 Desarrollo Aplicaciones Web	15
2.1.2 Tecnologías para implementar Aplicaciones Web	15
2.1.2.1 Plataformas Tecnológicas	16
2.1.2.1.1 Sistema Operativo del Servidor.....	16
2.1.2.1.2 Motor de Base de Datos Relacional.....	16
2.1.2.1.3 Servidor Web, Servidor de Aplicaciones JEE	17
2.1.2.1.4 Entorno de Desarrollo Integrado (IDE).....	18
2.2 Introducción a las metodologías de Desarrollo Ágil.....	19
Figura 2.2: Principios Ágiles.....	22
2.2.1 Análisis de las Metodologías ágiles de Desarrollo MSF, AUP, X.P.....	23
2.2.1.1 Microsoft Solution Framework (MSF).....	23
2.2.1.1.1 Principales ventajas de MSF:.....	24
2.2.1.1.2 Principales desventajas de MSF:	24
2.2.1.2 Extreme Programming	25
2.2.1.2.1 Principales ventajas de X.P.:.....	26
2.2.1.2.2 Principales desventajas de X.P.:.....	27
2.2.1.3 Ágil Unified Process (AUP)	27
2.2.1.3.1 Principales ventajas de AUP.:.....	28
2.2.1.3.2 Principales desventajas de AUP:	28
2.3 Comparación entre las Metodologías candidatas	29
2.3.1 Diferencias entre Metodología Tradicionales y Ágiles	29
2.4 Descripción de la metodología seleccionada	31

2.4.1	Metodología X.P (eXtreme Programming)	31
2.4.1.1	Introducción a la metodología X.P	31
2.4.1.2	Valores de X.P.....	35
2.4.1.3	Principios de X.P	36
2.4.1.4	Actividades de X.P	36
2.4.1.5	Prácticas de X.P.....	37
2.4.1.6	Ciclo de Vida de La Metodología X.P	38
2.4.1.7	Roles en X.P.....	41
2.4.1.8	¿Cuándo no implantar X.P?	43
2.5	Selección de las herramientas de desarrollo.....	43
2.5.1	Descripción de Las Herramientas de Desarrollo.....	45
2.5.1.1	Eclipse	45
2.5.1.2	DSpace.....	46
2.5.1.2.1	Beneficios de utilizar DSpace.....	49
2.5.1.2.2	Ventajas Técnicas de utilizar DSpace.....	49
2.5.1.3	PostgreSQL.....	50
2.5.1.3.1	Características PostgreSQL.....	51
2.5.1.3.1.1	Características Generales.....	51
CAPÍTULO 3: ANÁLISIS DEL PROYECTO		52
3.1	Línea Base y Estado del Arte	52
3.1.1	Producción de Información vs Necesidades y Uso.....	53
3.1.2	Capacitación y Aspectos Organizativos	54
3.2	Apreciación General	54
3.3	Usuarios del Sistema.....	55
3.4	Requisitos Funcionales.....	56
3.4.1	Historias de Usuario módulo: Administración de Usuarios	57
3.4.2	Historias de Usuario módulo: Repositorio digital de informes técnicos	58
3.4.3	Historias de Usuario módulo: Gestión de Inventario.....	58
3.4.4	Historias de Usuario módulo: Gestión Contable	60
3.5	Requisitos No Funcionales.....	62
3.6	Definición de Prioridades.....	63
3.7	Definición de Iteraciones	65
CAPÍTULO 4: DISEÑO DEL PROYECTO		66
4.1	Arquitectura y Diseño de Aplicaciones Java EE	66
4.1.1	Qué es JEE:	67
4.1.2	Componentes Java EE.....	67
4.1.3	Contenedores Java EE	68
4.1.4	Patrón Arquitectónico MVC	68
4.1.5	Capas en una Aplicación Java EE	69
4.2	Arquitectura del Sistema Web.....	70
4.3	Interfaces	73
4.3.1	Página Principal.....	73

4.3.2	Login.....	74
4.4	Portal Repositorio digital de Informes técnicos	75
4.5	Interfaces Módulo de Inventario.....	76
4.5.1	Movimientos – Ingresos	76
4.5.2	Movimientos – Egresos	77
4.5.3	Administración Productos.....	78
4.5.4	Administración Bodegas.....	79
4.5.5	Administración Órdenes de Trabajo.....	80
4.5.6	Administración Cotizaciones	81
4.6	Interfaces Módulo Contable.....	82
4.6.1	Administración Comprobantes de Ingreso, Egreso y Transacciones	82
4.6.2	Administración Facturas.....	83
4.6.3	Reportes Contables.....	84
4.7	Casos de Uso	85
4.7.1	Diagrama de Casos de Uso módulo I: Gestión de Usuarios	85
4.7.1.1	Descripción de C.U módulo I: Gestión de Usuarios.....	86
4.7.2	Diagrama de C.U módulo II: Gestión de Informes Técnicos.....	89
4.7.2.1	Descripción de C.U módulo II: Gestión de Informes Técnicos	90
4.7.3	Diagrama de C.U módulo III: Gestión de Inventario	101
4.7.3.1	Descripción de C.U módulo III: Gestión de Inventario	102
4.7.4	Diagrama de C.U módulo IV: Gestión Contable	119
4.7.4.1	Descripción de casos de uso módulo IV: Gestión Contable	120
4.8	Diagrama Entidad – Relación CTKSYS (Módulo de Inventario).....	128
4.9	Diagrama Entidad – Relación CTKSYS (Módulo Contable).....	129
4.10	Diagrama Entidad – Relación CTKREPOSITORIO (Parte I)	130
4.11	Diagrama Entidad – Relación CTKREPOSITORIO (Parte II)	131
4.12	Diagrama Entidad – Relación CTKREPOSITORIO (Parte III)	132
4.13	Diagrama de Clases CTKWEB – Módulo de Inventario (Parte I).....	133
4.14	Diagrama de Clases CTKWEB – Módulo de Inventario (Parte II).....	134
4.15	Diagrama de Clases CTKWEB – Módulo Contable	135
4.16	Diagrama de Clases CTKREPOSITORIO	136
4.17	Diagrama de flujo de trabajo – Inventario Administrar Proyectos	137
4.18	Diagrama de Componentes	138
4.19	Diagrama de Despliegue	139
CAPÍTULO 5: DESARROLLO DEL SISTEMA		140
5.1	Estandarización de los elementos de desarrollo	141
5.1.1	Código de Aplicaciones.....	142
5.1.1.1	Paquetes	142
5.1.1.2	Clases.....	142
5.1.1.3	Interfaces.....	145
5.1.1.4	Archivos de presentación	146
5.1.1.5	Métodos.....	148

5.1.1.6	Variables.....	148
5.1.2	Modelado de Datos	149
5.2	Módulos del Sistema	149
5.2.1	Gestión de perfiles de usuario.....	149
5.2.2	Gestión de Inventario	150
5.2.3	Gestión Contable.....	151
CAPÍTULO 6:	PRUEBAS	152
6.1	Pruebas de Unidad	153
6.2	Pruebas de Integración	154
6.3	Pruebas de Aceptación	155
CAPÍTULO 7:	CONCLUSIONES Y RECOMENDACIONES.....	161
7.1	Conclusiones	161
7.2	Recomendaciones	162
BIBLIOGRAFÍA	163

LISTADO DE TABLAS

Tabla 1.2: Descripción del Software.....	11
Tabla 1.3: Descripción recursos humanos.....	11
Tabla 1.4: Descripción suministros de oficina.....	11
Tabla 1.5: Descripción servicio Hosting y Dominio	11
Tabla 1.6: Descripción Total	12
Tabla 2.1: Ejemplo Sistema Operativo del Servidor – Software Libre.....	16
Tabla 2.2: Ejemplo Sistema Operativo del Servidor – Software Propietario.....	16
Tabla 2.3: Ejemplo Motor BDD Relacional MySql – SW Libre	16
Tabla 2.4: Ejemplo Motor BDD Relacional PostreSQL – SW Libre	16
Tabla 2.5: Ejemplo Motor BDD Relacional Oracle– SW Propietario	17
Tabla 2.6: Ejemplo Motor BDD Relacional SQL Server – SW Propietario	17
Tabla 2.7: Ejemplo Servidor Web – SW Libre.....	17
Tabla 2.8: Ejemplo Servidor de Aplicaciones – SW Libre.....	18
Tabla 2.9: Ejemplo Servidor de Aplicaciones – SW Propietario	18
Tabla 2.10: Ejemplo IDE – SW Libre	18
Tabla 2.11: Ejemplo IDE – SW Propietario.....	18
Tabla 2.12: Comparación entre las Metodologías Ágiles y las Tradicionales	29
Tabla 2.13: Matriz de comparación entre las Metodologías candidatas	30
Tabla 2.14: Lo “extremo” de X.P.....	32
Tabla 2.15: Valores de X.P.....	35
Tabla 2.16: Principios de X.P	36
Tabla 2.17: Actividades de X.P	36
Tabla 2.18: Prácticas de X.P.....	37
Tabla 2.19: Ciclo de Vida de X.P	38
Tabla 2.20: Herramientas de Software Libre utilizadas.....	44
Tabla 3.1: Historia de Usuario I: Administración de Usuarios	57
Tabla 3.2: Historia de Usuario II: Ingreso y utilización del sistema.....	57
Tabla 3.3: Historia de Usuario III: Publicación de informes técnicos en la web.....	58
Tabla 3.4: Historia de Usuario IV: Control de la entrada de productos en bodegas.....	58
Tabla 3.5: Historia de Usuario V: Control de salidas de producto en bodegas.....	59
Tabla 3.6: Historia de Usuario VI: Administración Órdenes de Trabajo.....	59
Tabla 3.7: Historia de Usuario VII: Administración Cotizaciones	60
Tabla 3.8: Historia de Usuario VIII: Elaboración comprobantes de Ingreso	60
Tabla 3.9: Historia de Usuario IX: Elaboración comprobantes de Egreso.....	60
Tabla 3.10: Historia de Usuario X: Elaboración de facturas	61
Tabla 3.11: Historia de Usuario VIII: Elaboración del Plan de Cuentas	61
Tabla 3.12: Historia de Usuario XII: Elaboración del reporte contable Libro Diario	61
Tabla 3.13: Historia de Usuario XIII: Elaboración del reporte Mayor General	62
Tabla 3.14: Historia de Usuario XIV: Elaboración Balance de Comprobación	62

Tabla 3.15: Historia de Usuario XV: Elaboración del Estado de Resultados.....	62
Tabla 3.16: Análisis Historias de Usuario – Prioridad Alta.....	64
Tabla 3.17: Análisis Historias de Usuario – Prioridad Media.....	64
Tabla 3.18: Análisis Historias de Usuario – Prioridad Baja.....	64
Tabla 3.19: Estimación de Iteraciones.....	65
Tabla 6.1: Resultado de Pruebas.....	158

LISTADO DE FIGURAS

Figura 2.1: La realidad del desarrollo de Software.....	19
Figura 2.2: Principios Ágiles.....	22
Figura 2.3: Prácticas X.P.....	31
Figura 2.4: Partes fundamentales en las que se basa X.P.....	34
Figura 2.5: Ciclo de vida de un proyecto X.P.....	39
Figura 2.6: El flujo del Juego de la Planificación.....	40
Figura 2.7: Roles en X.P.....	41
Figura 2.8: Cuando no implantar X.P.....	43
Figura 2.9: Forma de organizar la información en DSpace.....	47
Figura 2.10: Funcionamiento de DSpace.....	48
Figura 2.12: Componentes PostgreSQL.....	50
Figura 4.1: Arquitectura aplicación JEE.....	70
Figura 4.2: Arquitectura del Sistema Web CTKSYS.....	71
Figura 4.3: Pantalla página de inicio CTKSYS.....	73
Figura 4.4: Pantalla de Login.....	74
Figura 4.5: Pantalla de Inicio Repositorio CTK.....	75
Figura 4.6: Pantalla Movimientos de Inventario – Ingresos.....	76
Figura 4.7: Pantalla Movimientos de Inventario – Egresos/Requisiciones.....	77
Figura 4.8: Pantalla Administración de Productos.....	78
Figura 4.9: Pantalla Administración de Bodegas.....	79
Figura 4.10: Pantalla Administración de Órdenes de Trabajo.....	80
Figura 4.11: Pantalla Administración de Cotizaciones.....	81
Figura 4.12: Pantalla Administración Comprobantes.....	82
Figura 4.13: Pantalla Administración de Facturas.....	83
Figura 4.14: Pantalla Reportes Contables.....	84
Figura 4.15: Diagrama C.U módulo Gestión de Usuarios.....	85
Figura 4.16: Diagrama C.U módulo Gestión de Informes técnicos.....	89
Figura 4.17: Diagrama C.U módulo Gestión de Inventario.....	101
Figura 4.18: Diagrama C.U módulo Gestión Contable.....	119
Figura 4.19: Diagrama Entidad Relación CTKWEB (Módulo de Inventario).....	128
Figura 4.20: Diagrama Entidad Relación CTKWEB (Módulo Contable).....	129
Figura 4.21: Diagrama Entidad Relación CTKREPOSITORIO (Parte I).....	130
Figura 4.22: Diagrama Entidad Relación CTKREPOSITORIO (Parte II).....	131
Figura 4.23: Diagrama Entidad Relación CTKREPOSITORIO (Parte III).....	132
Figura 4.27: Diagrama de Clases CTKREPOSITORIO.....	136
Figura 4.28: Diagrama de Flujo de Trabajo – Inventario Administrar Proyectos.....	137
Figura 4.29: Diagrama de Componentes CTKSYS.....	138
Figura 4.30: Diagrama de Despliegue CTKSYS.....	139
Figura 5.1: Clase Cliente, captura desde IDE Eclipse.....	143

Figura 5.2: Clase AdminCrudImpl implementa la interface genérica IAdminCrud	144
Figura 5.3: Clase ClienteBean, captura desde IDE Eclipse	145
Figura 5.4: Interface genérica IAdminCrud, captura desde IDE Eclipse	146
Figura 5.5: plantillaInventario.xhtml, captura desde IDE Eclipse	147
Figura 5.6: admincliente.xhtml, captura desde IDE Eclipse	147
Figura 6.1: Ciclo de Pruebas	152
Figura 6.2: Formato Test Case.....	156
Figura 6.3: Porcentaje de adaptación al sistema CTKSYS	158
Figura 6.4: Gasto de papel en hojas.....	159
Figura 6.5: Tiempo ejecución de actividades en minutos	159

RESUMEN

El presente trabajo muestra el desarrollo de un sistema web que permite automatizar los procesos de control de Inventario de multibodegas, gestión de los principales procesos contables, y la publicación de documentos (informes técnicos) en la web, a través de la utilización/acoplamiento del Repositorio Digital DSpace a las necesidades de la empresa Crosstronik Cía. Ltda. El sistema web también toma en consideración los procesos de generación de Órdenes de trabajo, y Cotizaciones.

Para estructurar, planificar y controlar el proceso de desarrollo del sistema, se utilizó la metodología de desarrollo ágil, Programación Extrema (X.P); porque es una metodología bastante flexible que promueve un desarrollo iterativo e incremental, y la retroalimentación frecuente del código.

La herramienta desarrollada lleva el nombre CTKSYS. Se trata de un sistema web distribuido desarrollado en Java, haciendo uso del entorno de desarrollo Eclipse Índigo. Utiliza el motor de base de datos PostgreSQL y como servidor de aplicaciones Jboss AS 6.1. Todas las herramientas utilizadas son software libre, con el fin de utilizar las últimas tecnologías de vanguardia y disminuir costos por licencias.

Con la implementación y utilización adecuada de la herramienta desarrollada, se optimiza el manejo de los procesos internos de la empresa, tiempo y organización; todo en su conjunto genera satisfacción al usuario interno y externo, brindando una imagen corporativa robusta.

CAPÍTULO 1: INTRODUCCIÓN

1.1 Antecedentes

Crosstronik Cía. Ltda. es una empresa privada, orientada hacia el área de la Ingeniería Eléctrica y Electrónica. Actualmente, el enfoque de negocio de ésta empresa es prestar servicios en el ámbito de Telecomunicaciones, Control y Automatización.

Dentro de las actividades que son realizadas por el personal técnico y administrativo que trabaja en la empresa, existen una serie de necesidades y problemáticas; sin embargo, a las que se enfoca este proyecto son específicamente a la gestión de inventario de productos, que se encuentran en distintas bodegas, situadas en Quito, y Santo Domingo de los Tsáchilas, y que debido a la ausencia de un sistema informático apropiado, se ha dificultado la administración de éstos, por lo que es necesario recurrir a la gestión sistemática a través del uso de las tecnologías de información para optimizar este proceso.

De forma paralela existe otra necesidad similar referente a la publicación de informes técnicos que se emiten de manera semanal y mensual. Esta información debe estar publicada, de tal manera que sea accesible desde cualquier sitio, ofreciendo alta disponibilidad y seguridad. Por otro lado, también es importante solventar los requerimientos provenientes del Proceso de Apoyo Contable de la empresa. Estas necesidades hacen que se requiera desarrollar tecnología de software aplicado a la medida, basada en las tendencias de desarrollo de software actuales, como: metodologías, arquitecturas,

frameworks, estándares, tecnologías y estilos de uso, entre otras.

Teniendo como base estas tendencias, se propone una solución óptima de desarrollo de software enfocada hacia las aplicaciones web, distintas a las de escritorio, donde ya no es necesario tener todo el software instalado en determinadas computadoras; sino que hacen posible la utilización del software desde la web, ahorrando costos de instalación, actualización, hardware, etc.

Para el desarrollo de éste proyecto, un punto importante a tomar en cuenta, es la realidad económica de esta empresa; ya que no cuenta con los recursos económicos suficientes como para costear pagos por la adquisición de software privativo, por lo tanto, se hace uso de software libre, para no generar ningún gasto por licencias.

Todas estas características son las que impulsan el desarrollo de éste proyecto de tesis para llevarlo a un nivel competitivo en base a las tecnologías actuales.

1.2 Planteamiento del Problema

La primera problemática y la más importante que plantea el personal técnico de Crosstronik, es la de disponer de un Sistema web que además de ser la carta de presentación de la empresa; sea capaz de gestionar, difundir, y facilitar el acceso a través de Internet, a los archivos digitales (informes técnicos) generados de forma semanal y mensual conforme avanza un determinado proyecto en el área de redes. Con el objetivo de organizar la información y evitar la pérdida de tiempo en la emisión, recepción y control de

archivos por correo electrónico, de modo que tanto la gerencia de la empresa contratista encargada de revisar el desempeño de Crosstronik -actualmente dedicada a proveerle sus servicios-, como cualquier persona autorizada, pueda acceder a dichos informes; los cuales deben estar disponibles y accesibles mediante un navegador de Internet a cualquier instante.

Al personal técnico y administrativo también le resulta muy útil reemplazar los archivos de base de datos que tienen en Microsoft Access, los archivos de formatos que tienen en Microsoft Excel, y en medios físicos (hojas); para llevar el control de inventario del material electrónico que se encuentra en sus bodegas, ubicadas en Santo Domingo de los Tsáchilas y Quito.

Debido a que existe un manejo erróneo de éstos archivos y no están actualizados ni disponibles cuando se requiere, sería bastante útil integrar en la parte del administrador de éste sistema, una sección donde se pueda hacer las consultas básicas para la base de datos de productos. Que, a través de un navegador se pueda dar de alta, bajas o modificaciones usando una base de datos única, actualizada y siempre disponible, validando datos de entrada para llevar un control de coherencia dentro de ésta información; que también lleve a cabo todos los detalles del proceso para la realización de requisiciones de material¹ electrónico, con la cual se pueda automatizar los cálculos que se deben realizar.

Es notorio además, la mínima automatización de algunos cálculos (la mayoría son hechos en Microsoft Excel), es por esto que, se persigue

¹ Requisiciones de Materiales: Solicitud escrita que usualmente se envía para informar acerca de una necesidad.

automatizar los procesos de apoyo contable de la empresa, para la generación de reportes contables, tales como: comprobantes de ingreso, comprobantes de egreso y facturas, mismos que están relacionados directamente con el proceso de generación de Órdenes de trabajo. Basado en esta idea, a futuro, este sistema podrá incorporar nuevas funcionalidades, que complementen en su totalidad la gestión contable dentro de la empresa.

1.3 Objetivos

1.3.1 Objetivo general

Desarrollar un Sistema Web para la empresa Crosstronik Cía. Ltda., mediante la investigación, aplicación de tecnologías actuales, herramientas de software libre y siguiendo una metodología de desarrollo acorde a los requerimientos de éste proyecto, para gestionar archivos digitales, controlar y administrar los movimientos de materiales de cualquier bodega, y llevar un control de la información contable.

1.3.2 Objetivos específicos

- Crear un Sistema Web en lenguaje Java, y utilizando tecnología Primefaces que permita mejorar la presentación e información publicitaria de la empresa, con el fin de captar clientes potenciales y ofertar los servicios y productos que se prestan.

- Utilizar la herramienta DSpace que gestiona archivos de repositorios digitales; para administrar, difundir y facilitar el acceso a través de internet, a aquella información perteneciente a los informes técnicos emitidos semanal y mensualmente.
- Desarrollar el módulo de Inventario para controlar y administrar todos los movimientos de productos almacenados en cualquiera de las bodegas de la empresa.
- Crear el módulo de gestión contable, que administre y controle los documentos contables comerciales como (Facturas) y no comerciales como (Comprobante de Ingreso, Comprobante de Egreso, Órdenes de Trabajo, Cotizaciones y Requisiciones), con el propósito de mantener el control interno tanto de las actividades operativas y no operativas a nivel contable.
- Aplicar la metodología de desarrollo ágil XP, para garantizar una integración de los módulos de forma efectiva, y complementaria con otras ideas desde la perspectiva del negocio, los valores humanos y el trabajo en equipo.

1.4 Justificación

A continuación se detallan las razones que justifican la creación de este proyecto: se realizará un sistema web, para poder acceder desde cualquier computador, con salida a internet, y desde cualquier lugar en que se encuentre

el técnico, representante de la empresa o un usuario cualquiera, a la información disponible vía web, sobre Crosstronik; con el fin de publicitar los productos y servicios que ofrece la empresa, obtener nuevos proyectos de trabajo, y captar clientes potenciales.

Se evitará la demora en el envío, recepción, control de los informes técnicos, para eliminar la duplicidad de tareas cumplidas, que el gerente de la empresa contratista o el técnico encargado; lleven consigo toda la información en su computadora o dispositivo extraíble, mucho menos impresos o almacenados temporalmente en la bandeja de entrada de su correo electrónico, debido a que puede generar problemas, como pérdida de documentos.

Con la implementación de un módulo para Repositorio Digital de Informes Técnicos, dentro del sitio web, se logrará que la información de dichos informes, esté siempre disponible para el personal que cuenta con los privilegios necesarios y pueda hacer uso de ésta información; de tal forma que, el gerente o técnico, no tenga que buscar entre grandes cantidades de correos que están dentro de la bandeja de entrada de su correo electrónico, para poder revisar la información de algún informe que necesite.

Se logrará mantener enlazadas las actuales sucursales de la empresa; ubicadas en las ciudades de Quito y Santo Domingo de los Tsáchilas.

Mediante la sección de manejo de Inventario, el personal técnico, o administrativo de la empresa que cuente con los privilegios de acceso necesarios, tendrá disponible dentro del mismo Sistema web las opciones para: ingresar registros, dar de alta, bajas o modificaciones usando una base de datos, evitando así, que la toma de datos sea realizada por medios físicos

- hojas de papel -, o a través de hojas electrónicas elaboradas en Excel.

Además, se automatizará el proceso para la realización de Requisiciones de Material, donde el técnico podrá llenar el formato perteneciente al conjunto de herramienta, material o equipo que necesita obtener de alguna bodega. Esto facilitará un control efectivo del existente físico de todos y cada uno de los elementos almacenados en bodega, permitirá anticipar la reposición de materiales y evitar así retrasos en el despacho de todos los pedidos por falta de stock.

Teniendo en cuenta que el nivel de automatización en los procesos contables iniciales dentro de la empresa, es bastante escaso, se eliminará la realización manual de reportes contables (comprobantes de ingreso, egreso, facturas) que tienen relación directa al momento de realizar una Orden de Trabajo; simplificando la generación y administración de éstos.

1.5 Alcance

El Sistema Web para la empresa Crosstronik, permitirá:

- Administrar Usuarios por primera vez, si es requerido:
- Ingresar, borrar, actualizar datos de usuario.

Control de accesos:

- Validar el ingreso del usuario a tres secciones específicas: Repositorio Digital de Informes Técnicos, Inventarios y Gestión Contable.

Gestionar el Repositorio de Archivos Digitales (Informes Técnicos):

- Utilizar DSpace, Repositorio de Contenidos Digitales de código abierto; que permitirá:
- Almacenar y gestionar archivos digitales (documentos en formato pdf) pertenecientes a los informes técnicos generados semanal y mensualmente.
- Distribuir los recursos digitales de la empresa (informes técnicos) sobre la web a través de un sistema de búsqueda y recuperación.
- Preservar contenidos digitales a largo plazo.

Gestión Contable:

- Automatizar las tareas que la empresa cumple para el control interno de las actividades operativas y no operativas a nivel contable, como por ejemplo: generación de comprobantes de ingreso, comprobantes de egreso y facturas.

Control de Inventario:

- Controlar las actividades (entrada y salida de productos) en cualquier bodega de la empresa.
- Asignar distintos permisos de usuario por secciones y definir qué movimientos se pueden realizar en cada una; de manera que se controlen mejor las existencias.
- Dar de alta, bajas, o modificaciones, llevar un Kárdex/historial de los productos pertenecientes a bodega, de cualquier ubicación.

- Generar el formato de Órdenes De Trabajo, Cotización De Materiales, y Requisiciones de Bodega, donde el personal de las diferentes áreas registre los requerimientos del trabajo a realizar, los elementos y cantidades que se desean adquirir y que se emplearán para la realización de la orden de trabajo en curso.
- Control de Inventario, estará relacionado con el módulo de Contabilidad, al momento de generarse una Orden de Trabajo.

El sistema será implementado acorde a los requerimientos solicitados por la empresa Crosstronik Cía. Ltda. Se pretende además que el sistema en mención sea escalable desde el punto de vista del desarrollador, es decir, que pueda incorporar nuevas funcionalidades que a futuro sean requeridas.

1.6 Factibilidad

1.6.1 Factibilidad Técnica

Los desarrolladores tienen los conocimientos necesarios y suficientes acerca de: aplicaciones web y el motor de bases de datos, para realizar el presente proyecto de tesis; en cuanto a las herramientas de desarrollo a utilizar se necesita realizar una mayor investigación con respecto a la herramienta DSpace dado que no existe ningún curso de capacitación actualmente en el país.

1.6.2 Factibilidad Económica

El presupuesto estimado para el sistema se muestra a continuación:

Tabla 1.1: Descripción del Hardware

Material	Valor
2 PCs	\$2400
1 impresora	\$100.00
Subtotal	\$2500.00

Tabla 1.2: Descripción del Software

Material	Valor
PostgreSQL	\$0.00
Eclipse	\$0.00
Visual Paradigm for UML	\$0.00
DSpace	\$0.00
Primefaces	\$0.00
Subtotal	\$0.00

Tabla 1.3: Descripción recursos humanos

Desarrollador	Valor
Cruz Caiza Diana	\$318.00x12
Zhamungui Oviedo Christian	\$318.00x12
Subtotal	\$7632.00

Tabla 1.4: Descripción suministros de oficina

Material	Cantidad	Valor
Papel Bond	2000	\$20.00
Cartuchos Impresora	3	\$60.00
Internet	Mensual	\$20.00
Varios		\$30.00
Subtotal		\$130.00

Tabla 1.5: Descripción servicio Hosting y Dominio

Recurso	Valor
Dominio	\$20.00
Hosting Jboss AS 6.1, PostreSQL 8.4	\$64.95
Subtotal	\$84.95

Tabla 1.6: Descripción Total

Recurso	Valor
Hardware	\$2500.00
Software	\$0.00
Recursos Humanos	\$7632.00
Suministros de oficina	\$130.00
Servicio Hosting y Dominio	\$84.95
Total	\$10346.95

Con respecto al costo de hardware, se entregarán los equipos dentro de la empresa, por lo cual no es necesario adquirirlos y no representaría un gasto para el desarrollo del proyecto.

1.6.3 Factibilidad Operativa

Crosstronik Cía. Ltda., como auspiciante de éste proyecto de tesis, facilitará toda la información que se requiera, y colaborará con todos los aspectos necesarios para la satisfactoria culminación del mismo.

Además garantiza el cumplimiento en fechas y tiempos estimados para las entrevistas, de manera que la retroalimentación se pueda realizar de forma continua.

1.6.4 Factibilidad Operacional

Con respecto a las interfaces del sistema, estas serán estandarizadas y estéticamente agradables para el usuario de manera que se reduzcan los

tiempos de capacitación para cada tipo de usuario que estará disponible en el sistema; y dado que es un software de tipo especializado y de uso diario, reduzca efectos de cansancio mental y visual.

1.6.5 Factibilidad Tecnológica

Para la realización del presente sistema se requieren los siguientes equipos:

- Dos computadoras con al menos 80 GB de Disco Duro y 3 GB de memoria RAM.
- Una impresora

Los programas que se utilizarán para el desarrollo del sistema son:

- JDK (Java Development Kit) 1.7
- Eclipse Índigo 3.7 SR2
- JBoss AS 6.1
- PostgreSQL 8.4
- Primefaces 3.2
- DSpace 1.8.2
- Jasper Reports 4.7.1

1.6.6 Factibilidad Legal

Dado que los programas que se van a utilizar son Open Source existe una viabilidad actual desde el punto de vista de procedimientos y procesos a seguir para utilizar dichos programas. Se ha corroborado que todos los programas a utilizar para el desarrollo del sistema actualmente no presentan ningún impedimento jurídico en el país, por tanto los autores liberan de cualquier responsabilidad a la empresa por cambio en las leyes de dichos programas a futuro.

1.7 Metodología

Realizado el análisis comparativo de algunas metodologías de Desarrollo Ágil; Extreme Programming (X.P) resulta ser la más idónea para el desarrollo de este sistema, debido a que se verán aplicadas las características fundamentales que plantea la metodología, como: el desarrollo iterativo e incremental, realización de pruebas unitarias continuas, programación por parejas, interacción frecuente del equipo de trabajo con el cliente/usuario, corrección de errores antes de añadir nueva funcionalidad, y la refactorización de código. En el capítulo 2 de este documento, se hace un análisis a detalle sobre la metodología seleccionada.

CAPÍTULO 2: MARCO TEÓRICO

2.1 Sistemas en Ambientes Web

2.1.1 Desarrollo Aplicaciones Web

El desarrollo de una Aplicación Web, consiste en una aplicación de software que se codifica en un lenguaje que es soportado por los navegadores, dispone de muchos componentes web; que sumados con capas de persistencia construyen aplicaciones robustas, escalables y que aceleran los tiempos de desarrollo. Además garantizan la disminución de recursos en instalación y configuración –donde la instalación del sistema se realiza en un servidor, no siendo necesario instalarlo en cada terminal que lo va a utilizar-, así como el soporte de múltiples usuarios y tareas concurrentes que es fundamental para cualquier empresa con esta cadena de valor.

2.1.2 Tecnologías para implementar Aplicaciones Web

Existe una gama de tecnologías y productos para el desarrollo de aplicaciones web. Sin embargo, lo que se debe buscar, es hacer uso de herramientas y plataformas que garanticen un desarrollo y mantenimiento estándar, un óptimo periodo de vida útil, y que además su vigencia en el mercado tecnológico esté garantizada a largo plazo. A continuación se muestran algunos ejemplos de tecnologías tanto a nivel de software libre como para software propietario para justificar el uso de cada uno.

2.1.2.1 Plataformas Tecnológicas

2.1.2.1.1 Sistema Operativo del Servidor

Tabla 2.1: Ejemplo Sistema Operativo del Servidor – Software Libre

Tipo de Software	Libre
Producto	Linux distros: Red Hat, Suse, Centos, Ubuntu
Requisitos	Ninguno

Tabla 2.2: Ejemplo Sistema Operativo del Servidor – Software Propietario

Tipo de Software	Propietario
Producto	Windows Server 2003 o superior; Sun Solaris 9 o superior
Requisitos	Ninguno

2.1.2.1.2 Motor de Base de Datos Relacional

Tabla 2.3: Ejemplo Motor BDD Relacional MySql – SW Libre

Tipo de Software	Libre
Producto	MySQL
Requisitos	Ninguno
Condiciones	Utilizar siempre que: <ul style="list-style-type: none">- Se necesite hacer aplicaciones ligeras de respuesta rápida. Volumen de información menor que 1Gb.- Se cuente con pocos recursos en el servidor de aplicaciones y/o en el servidor de Base de Datos

Tabla 2.4: Ejemplo Motor BDD Relacional PostgreSQL – SW Libre

Tipo de Software	Libre
Producto	PostgreSQL
Requisitos	Ninguno
Condiciones	Utilizar siempre que: <ul style="list-style-type: none">- La aplicación sea medianamente compleja y se procese un gran volumen de datos. Se quiera usar store-procedures, triggers.

Tabla 2.5: Ejemplo Motor BDD Relacional Oracle– SW Propietario

Tipo de Software	Propietario
Producto	Oracle
Requisitos	Se cuenta con la licencia de uso.
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones complejas a nivel empresarial. - Se tenga que manejar un volumen ilimitado de datos. - El sistema requiera una rápida recuperación de datos, tenga que soportar una carga de datos muy grande y que necesite soporte permanentemente.

Tabla 2.6: Ejemplo Motor BDD Relacional SQL Server – SW Propietario

Tipo de Software	Propietario
Producto	Microsoft SQL Server
Requisitos	Se cuenta con S.O Windows y con las licencias de uso.
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite desarrollar aplicaciones complejas bajo Windows. - Se tenga maneje un alto volumen de información.

2.1.2.1.3 Servidor Web, Servidor de Aplicaciones JEE

Tabla 2.7: Ejemplo Servidor Web – SW Libre

Tipo de Software	Libre
Producto	Jetty, Apache Tomcat
Requisitos	Ninguno
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones Java, que requieran un contenedor web que es exclusivamente para servlets.

Tabla 2.8: Ejemplo Servidor de Aplicaciones – SW Libre

Tipo de Software	Libre
Producto	Jboss, Glassfish
Requisitos	Ninguno
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones Java EE, que requieran un contenedor web y un contenedor de EJBs.

Tabla 2.9: Ejemplo Servidor de Aplicaciones – SW Propietario

Tipo de Software	Propietario
Producto	WebLogic, WebSphere, EAServer
Requisitos	Se cuenta con las licencias de uso.
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones JEE, que requiere contenedor web y de EJBs.

2.1.2.1.4 Entorno de Desarrollo Integrado (IDE)

Tabla 2.10: Ejemplo IDE – SW Libre

Tipo de Software	Libre
Producto	Netbeans 6.5 o superior, Eclipse Índigo, JDeveloper, IntelliJ IDEA Community Edition
Requisitos	Ninguno.
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones Java

Tabla 2.11: Ejemplo IDE – SW Propietario

Tipo de Software	Propietario
Producto	MyEclipse, JBuilder Enterprise, IntelliJ IDEA Ultimate Edition
Requisitos	Se cuenta con las licencias de uso.
Condiciones	Utilizar siempre que: <ul style="list-style-type: none"> - Se necesite hacer aplicaciones Java

2.2 Introducción a las metodologías de Desarrollo Ágil

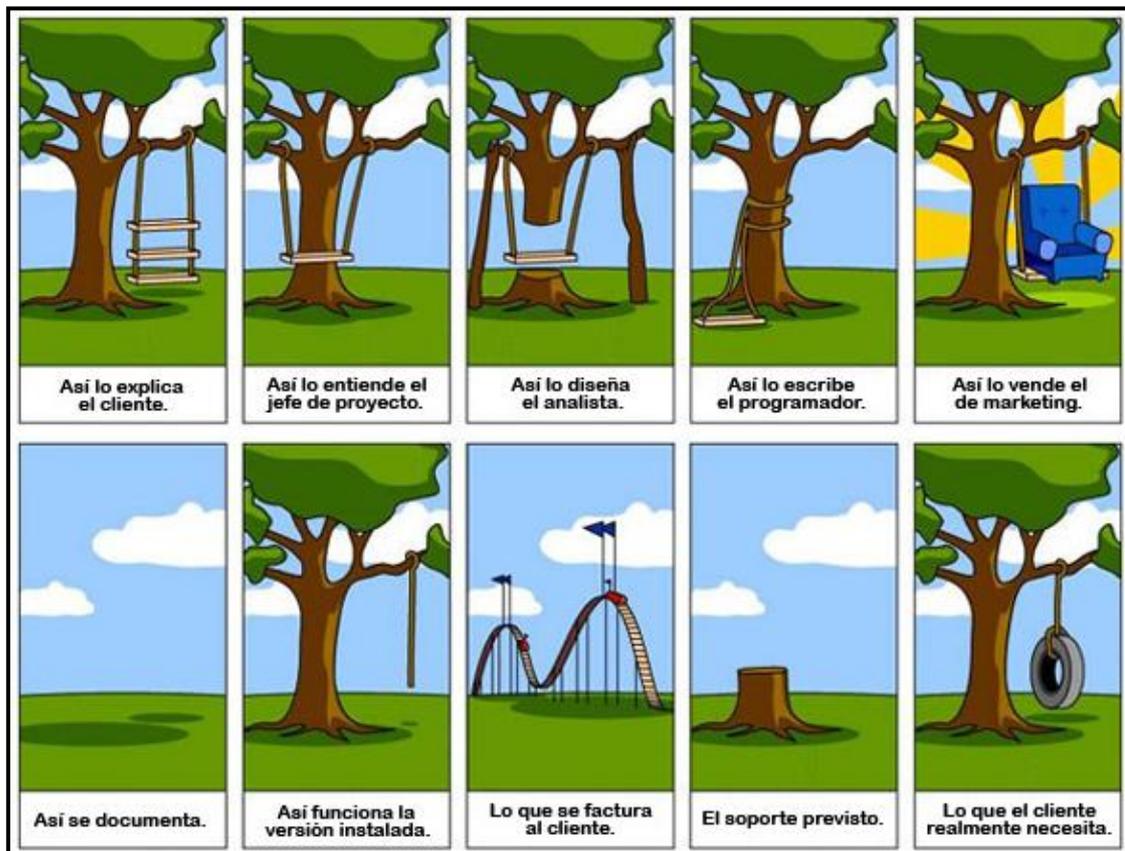


Figura 2.1: La realidad del desarrollo de Software²

La figura 2.1, muestra con humor, pero de forma precisa, las diversas situaciones que se dan en un gran número de proyectos de desarrollo de software.

¿Por qué es difícil desarrollar software? Una respuesta puntual a esa pregunta, es que hay perspectivas distintas al interpretar la realidad, lo que realmente requiere el cliente.

² Figura. La realidad sobre el desarrollo de software. Adaptado de "Modelo de Desarrollo Productivo", por Contreras E. Recuperado de <http://richzendy.org/docs/DesarrolloProductivo/>

Esto hace que el desarrollo de un proyecto de software sea riesgoso y difícil de controlar, por eso es necesario tener un plano en que apoyarse. Es decir, que el desarrollador, debe llevar una metodología de por medio, que le ayude a saber cómo identificar las necesidades o problemas existentes, qué criterios usar para llegar a decidir la solución computarizada que se debe aplicar en el proyecto de software, entre otros.

Actualmente, existe una gran variedad de propuestas metodológicas que incluyen diversas herramientas en el proceso de desarrollo. Principalmente, estas metodologías se dividen en dos: las Metodologías Ágiles y las Tradicionales o Clásicas.

En este apartado, se hablará solo de las metodologías ágiles pues estas han sido la base fundamental para el éxito de este proyecto.

El desarrollo ágil de Software hace uso directo del Manifiesto Ágil de Software, mismo que sigue estos principios que Beck (2001) afirma:

- La mayor prioridad es satisfacer al cliente mediante la entrega temprana y continua de software con valor.
- Es aceptable que los requisitos cambien, incluso en etapas tardías del desarrollo. Los procesos Ágiles aprovechan el cambio para proporcionar ventaja competitiva al cliente.
- Entrega software funcional frecuentemente, entre dos semanas y dos meses, con preferencia al periodo de tiempo más corto posible.
- Los responsables de negocio y los desarrolladores deben trabajar juntos de forma cotidiana durante todo el proyecto.
- Los proyectos se desarrollan en torno a individuos motivados. Hay que darles el entorno y el apoyo que necesitan, y confiarles la ejecución del trabajo.
- El método más eficiente y efectivo de comunicar información al equipo de desarrollo y entre sus miembros es la conversación cara a cara.
- El software funcionando es la medida principal de progreso.

- Los procesos Ágiles promueven el desarrollo sostenible. Los promotores, desarrolladores y usuarios deben ser capaces de mantener un ritmo constante de forma indefinida.
- La atención continua a la excelencia técnica y al buen diseño mejora la Agilidad.
- La simplicidad, o el arte de minimizar la cantidad de trabajo no realizado, es esencial.
- Las mejores arquitecturas, requisitos y diseños emergen de equipos auto-organizados.³

Según el Manifiesto se valora:

- Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas. La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.
- Más el software que funciona que la documentación exhaustiva. La regla a seguir es “no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante” Beck (2001). Estos documentos deben ser cortos y centrarse en lo fundamental.
- Responder a los cambios más que seguir estrictamente un plan. La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo,

³ Adaptado de "Extreme Programming Explained: Embrace Change." Second Edition. [PDF]. Boston. 1999, por Beck K.

etc.) determina también éxito o fracaso del mismo. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta.

En el 2001, Kent Beck y otros autores de enfoques similares proponen los Principios ágiles:

Individuos e interacciones	por sobre	Procesos y herramientas
Software Funcional		Documentación exhaustiva
Colaboración con el cliente		Negociación de contratos
Responder al cambio		Seguir un plan

Figura 2.2: Principios Ágiles⁴

Una de las cualidades más destacables en una metodología ágil es su sencillez, tanto en su aprendizaje como en su aplicación, reduciéndose así los costos de implantación en un equipo de desarrollo.

Para la selección de la metodología de desarrollo a aplicar; se han tomado en cuenta las características de éste proyecto, el nivel de conocimiento que tiene el grupo de trabajo, y en base a éstos y otros aspectos (que serán analizados a través de un cuadro comparativo) se describirán tres metodologías: Microsoft Solution Framework (MSF), Programación Extrema (XP), y Proceso Unificado Ágil (AUP). Mismas que, se consideraron, idóneas para este proyecto.

⁴ Figura 5. Principios Ágiles. Adaptado de "Introducción a los Métodos Ágiles", por Villena A. 2010. Recuperado de <http://www.slideshare.net/chileagil/introduccion-gil-a-extreme-programming-webprendedor08-350127>.

2.2.1 Análisis de las Metodologías ágiles de Desarrollo MSF, AUP, X.P

2.2.1.1 Microsoft Solution Framework (MSF)

M.S.F. representa un resumen de las mejores prácticas que han aparecido en cuanto a administración de proyectos. Se la puede ver como una metodología rígida de administración de proyectos, pero es una serie de modelos que si se los aplica en un tipo adecuado de plan puede adaptarse a cualquier proyecto de tecnología de información; sin embargo es demasiado largo, debido a que cada una de sus fases requieren una extensa documentación, por lo que no es recomendable para proyectos que funcionen en base a tiempo o para proyectos cortos, sino para proyectos largos y que necesitan una visión clara del proyecto desde el inicio de éste; además es primordial para poder aplicar ésta metodología que se tengan claros los requerimientos del cliente desde el principio, pues cada retroalimentación con esta metodología representa un alto coste. El modelo de proceso de M.S.F. se compone de 5 fases o etapas iterativas:

- Visión y Alcances.
- Planificación.
- Desarrollo.
- Estabilización.
- Implantación.

Al final de cada fase se tienen que lograr alcances específicos –definidos por hitos- y que generan entregables para agregar valor al proyecto.

2.2.1.1.1 Principales ventajas de MSF:

- Puede ser utilizada en un proyecto de desarrollo sobre cualquier tecnología.
- No es requerido alcanzar un punto específico en una de las fases para poder avanzar a la siguiente.
- Crea una disciplina de análisis de riesgos que evoluciona a lo largo del proyecto.
- Permite la reutilización de componentes ya desarrollados en ciclos anteriores.

2.2.1.1.2 Principales desventajas de MSF:

- Existe dependencia al administrador de proyectos, ya que es el responsable directo de la interacción con el cliente y el grupo de trabajo; para tomar decisiones en caso de cambios o el tratamiento de nuevos requerimientos.
- Está basada en tecnología Microsoft, por lo que trata de obligar a utilizar sus propias herramientas. Generando dependencias de tecnologías propietarias.
- Demasiada documentación en sus fases.
- Los precios de licencias, capacitación y soporte de Microsoft son costosos.

2.2.1.2 Extreme Programming

La Programación Extrema está basada en la comunicación, la simplicidad, y la reutilización continua de código. Es considerada la más destacada de los procesos ágiles de desarrollo de software. Se diferencia de las metodologías tradicionales principalmente en que pone más énfasis en la adaptabilidad.

Los defensores de XP consideran que los cambios de requisitos sobre la marcha son un aspecto natural, inevitable e incluso deseable del desarrollo de proyectos.

Creer que ser capaz de adaptarse a los cambios de requisitos en cualquier punto de la vida del proyecto es una aproximación mejor y más realista que intentar definir todos los requisitos al comienzo del proyecto e invertir esfuerzos después en controlar los cambios en los requisitos. Las fases que se siguen en la metodología XP son las siguientes:

- Planificación
- Diseño
- Desarrollo
- Pruebas
- Finalización del Proyecto

Características fundamentales de la metodología:

- Desarrollo iterativo e incremental: pequeñas mejoras, unas tras otras.
- Pruebas unitarias continuas, frecuentemente repetidas y automatizadas.

- Programación por parejas.
- Frecuente interacción del equipo de desarrollo con el cliente o usuario.
- Corrección de todos los errores antes de añadir nueva funcionalidad.
- Hacer entregas frecuentes.
- Refactorización del código, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento.
- Simplicidad en el código: La programación extrema apuesta que es más sencillo hacer algo simple y tener un poco de trabajo extra para cambiarlo si se requiere, que realizar algo complicado y quizás nunca utilizarlo.

2.2.1.2.1 Principales ventajas de X.P.:

- El manejo, coste del cambio no depende de la fase o etapa de desarrollo.
- Comunicación constante entre usuarios y desarrolladores.
- Retroalimentación frecuente entre el equipo de desarrollo, y el cliente.
- Simplicidad en el análisis de la solución y codificación de los módulos del sistema.

2.2.1.2.2 Principales desventajas de X.P.:

- Debido a la falta de documentación no se puede tener un control completo de las iteraciones que se han ido realizando durante todo el proceso de desarrollo.
- El uso de una metodología Ágil, como X.P. sirve más para proyectos sencillos, de alcance corto; pues hace énfasis en el empleo del menor tiempo posible y no generar mucha documentación. Sin embargo, esto representa una desventaja ya que el documentar, puede servir para el control y seguimiento de lo que realiza el equipo de desarrollo.

2.2.1.3 Agil Unified Process (AUP)

La metodología AUP es una versión simplificada del Proceso Unificado de Rational (RUP); que describe de manera más simple la forma de desarrollar aplicaciones de software usando técnicas ágiles. Las disciplinas de AUP son:

- Modelado
- Implementación
- Prueba
- Despliegue
- Administración de la configuración
- Administración o gerencia del Proyecto

- Entorno

AUP es ágil, porque está basada en los siguientes principios:

- El personal sabe lo que está haciendo. La gente no va a leer de forma detallada el proceso de documentación, pero algunos quieren una orientación de alto nivel y/o formación de vez en cuando
- Simplicidad. Todo se describe concisamente.
- Agilidad. El ajuste a los valores y principios de la Alianza Ágil.
- Centrarse en actividades de alto valor. La atención se centra en las actividades que se ve que son esenciales para el de desarrollo.

2.2.1.3.1 Principales ventajas de AUP.:

- Metodología robusta con muchos artefactos y disciplinas por elegir.
- Prioridades establecidas sobre la base de mayor riesgo.
- La documentación ayuda a comunicarse en entornos distribuidos.

2.2.1.3.2 Principales desventajas de AUP.:

- La documentación es mucho más formal que en la mayoría de metodologías ágiles.
- Mínima atención a la dinámica del equipo de trabajo.

2.3 Comparación entre las Metodologías candidatas

A continuación se realiza un análisis comparativo entre algunas metodologías ágiles, de manera que se justifique el porqué se optó por la utilización de X.P para este proyecto.

2.3.1 Diferencias entre Metodología Tradicionales y Ágiles

Tabla 2.12: Comparación entre las Metodologías Ágiles y las Tradicionales⁵

Metodologías Tradicionales	Metodologías Ágiles
Basadas en normas provenientes de estándares seguidos por el entorno de desarrollo	Basadas en heurísticas provenientes de prácticas de producción de código
Cierta resistencia a los cambios	Especialmente preparados para cambios durante el proyecto
Impuestas externamente	Impuestas internamente por el equipo
Proceso mucho más controlado, con numerosas políticas o normas	Proceso menos controlado, con pocos principios.
Más artefactos	Pocos Artefactos
Más roles	Pocos roles
Grupos grandes y posiblemente distribuidos	Grupos pequeños (<10 integrantes) y trabajando en el mismo sitio
La arquitectura del software es esencial y se expresa mediante modelos	Menos énfasis en la arquitectura del software
Existe un contrato prefijado	No existe contrato tradicional o al menos es bastante flexible

⁵ Tabla 1. Diferencia entre metodologías ágiles y no ágiles. Adaptado de "Metodologías Ágiles en el Desarrollo de Software". Por Canós J, Letelier P. Y Penadés M. 2011. Valencia, España. Recuperado de www.willydev.net/descargas/prev/ToDoAgil.Pdf.

Tabla 2.13: Matriz de comparación entre las Metodologías candidatas

CRITERIOS	METODOLOGÍAS		
	Microsoft Solutions Framework (MSF)	Programación Extrema (XP)	Proceso Unificado Ágil (AUP)
Pocos roles y flexibles	2	3	1
Iterativa	1	3	1
Entorno amplio de proyectos de Software	2	3	2
Permite desarrollar software sobre cualquier tecnología	2	2	3
Nivel de conocimiento del grupo de trabajo	0	3	1
Soporte de Orientación a Objetos	3	3	3
Afinidad con el sistema	2	3	2
No presenta Resistencia a los cambios	1	3	1
Proceso Controlado	2	2	2
Comunicación con el cliente	1	3	1
TOTAL:	16	28	17

Valores:

- Nivel Alto = 3
- Nivel Medio = 2
- Nivel Bajo = 1
- Nivel Nulo = 0

Realizado el análisis comparativo entre las tres metodologías de desarrollo, se ha seleccionado Extreme Programming como determinante para orientar el proyecto. X.P. será bastante útil durante el desarrollo de éste proyecto, dado que lo que se quiere es evitar un diseño rígido del software, que cuando el cliente en la etapa final solicite algún cambio no sea difícil de realizar, y que, en el caso de que se lo hiciera, no alteraría muchos aspectos con respecto a lo contemplado desde un principio. Durante el desarrollo de este sistema, la realidad va ser otra; pues se tiene que ofrecer al cliente la posibilidad de conseguir un sistema acorde a sus requerimientos (que muchas veces no los tienen claros), por lo que, se irá mostrando el proyecto a tiempo, para ir

cambiándolo y poder retroceder a una fase para rediseñarlo a su gusto. Es decir, existirá una retroalimentación concreta y frecuente entre el equipo de desarrollo, el cliente y los usuarios finales.

Esta metodología se acopla perfectamente al proyecto, pues: el equipo de desarrollo es pequeño, se necesita reducir tiempos de desarrollo en base a simplificar el diseño, y la disminución del número de procesos involucrados.

2.4 Descripción de la metodología seleccionada

2.4.1 Metodología X.P (eXtreme Programming)

2.4.1.1 Introducción a la metodología X.P

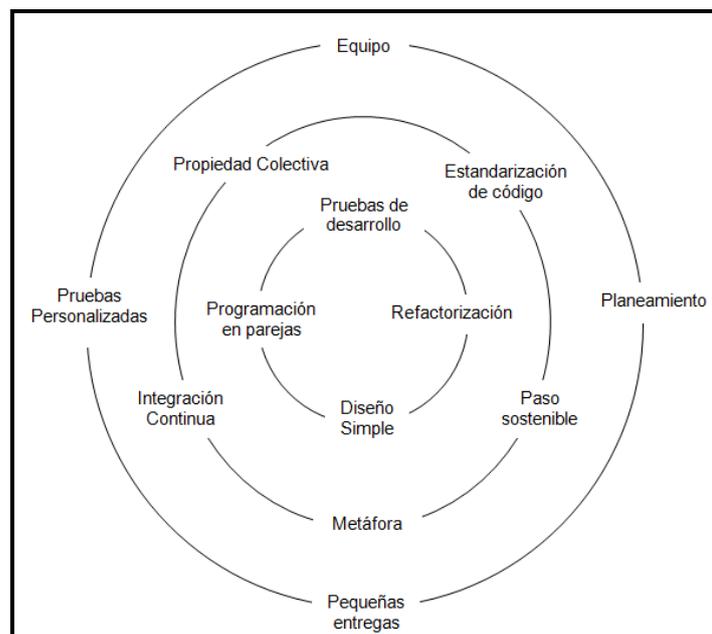


Figura 2.3: Prácticas X.P⁶

⁶ Figura. Procedimiento o conjunto de prácticas de Extreme Programming. Adaptado de “*Metodologías acordes con Agile Manifiesto*”, por Balzar E. A. Recuperado de <http://gmodulo.sourceforge.net/docs/html/manual/ch02s04.html>

“Todo en el software cambia. Los requisitos cambian. El diseño cambia. El negocio cambia. La tecnología cambia. El equipo cambia. Los miembros del equipo cambian. El problema no es el cambio en sí mismo, puesto que sabemos que el cambio va a suceder; el problema es la incapacidad de adaptarnos a dicho cambio cuando éste tiene lugar.”⁷ Kent Beck (1999)

Extreme Programming o Programación Extrema (X.P) es una metodología ligera para el desarrollo de software, enunciada por Kent Beck, Ward Cunningham, y Ron Jeffries. Esta metodología persigue dos objetivos:

El objetivo principal es satisfacer los requerimientos del cliente (Implica responder de forma rápida a los cambios en las necesidades del cliente, incluso cuando éstos cambios se produzcan al final del ciclo de vida del software)

El segundo objetivo es potenciar al máximo el trabajo en equipo (tanto Jefes de Proyecto, como desarrolladores y clientes, forman parte del equipo encargado de la implementación del software).

El término “extrema” se debe a la idea planteada por Kent Beck: “Llevar las buenas prácticas de Ingeniería de Software al extremo”. Por ejemplo:

Tabla 2.14: Lo “extremo” de X.P

Buenas Prácticas	Prácticas Extremas
Sistema de pruebas estructurado	Desarrollo guiado por pruebas durante todo el desarrollo del proyecto.
Revisiones de código	Programación en parejas
Software funcionando totalmente	Entregas incrementales e integración continua
Tener alineado al cliente	Cliente in situ

⁷ Adaptado de “Extreme Programming Explained: Embrace Change.” Second Edition. [PDF]. Boston. 1999, por Beck K.

- Si revisar el código es bueno, entonces en X.P se revisará el código de manera exhaustiva, valiéndose de la programación en pareja.
- Si realizar pruebas es bueno, entonces en X.P se realizarán pruebas continuamente, a través de las pruebas unitarias que realiza el desarrollador, hay que añadir a este aspecto, las pruebas que realiza el cliente (pruebas de aceptación).
- Si diseñar la solución del proyecto es bueno, entonces X.P lo hace parte del trabajo diario del equipo de desarrollo, valiéndose de la Refactorización.
- Si desarrollar realizando iteraciones cortas a lo largo del proyecto es bueno, entonces X.P plantea la realización de iteraciones que en un corto periodo de tiempo.

Esta metodología está formada por cuatro partes fundamentales: valores, principios, actividades y prácticas; que se ejecutan durante todo el ciclo de desarrollo del proyecto.

ESPACIO EN BLANCO INTENCIONAL

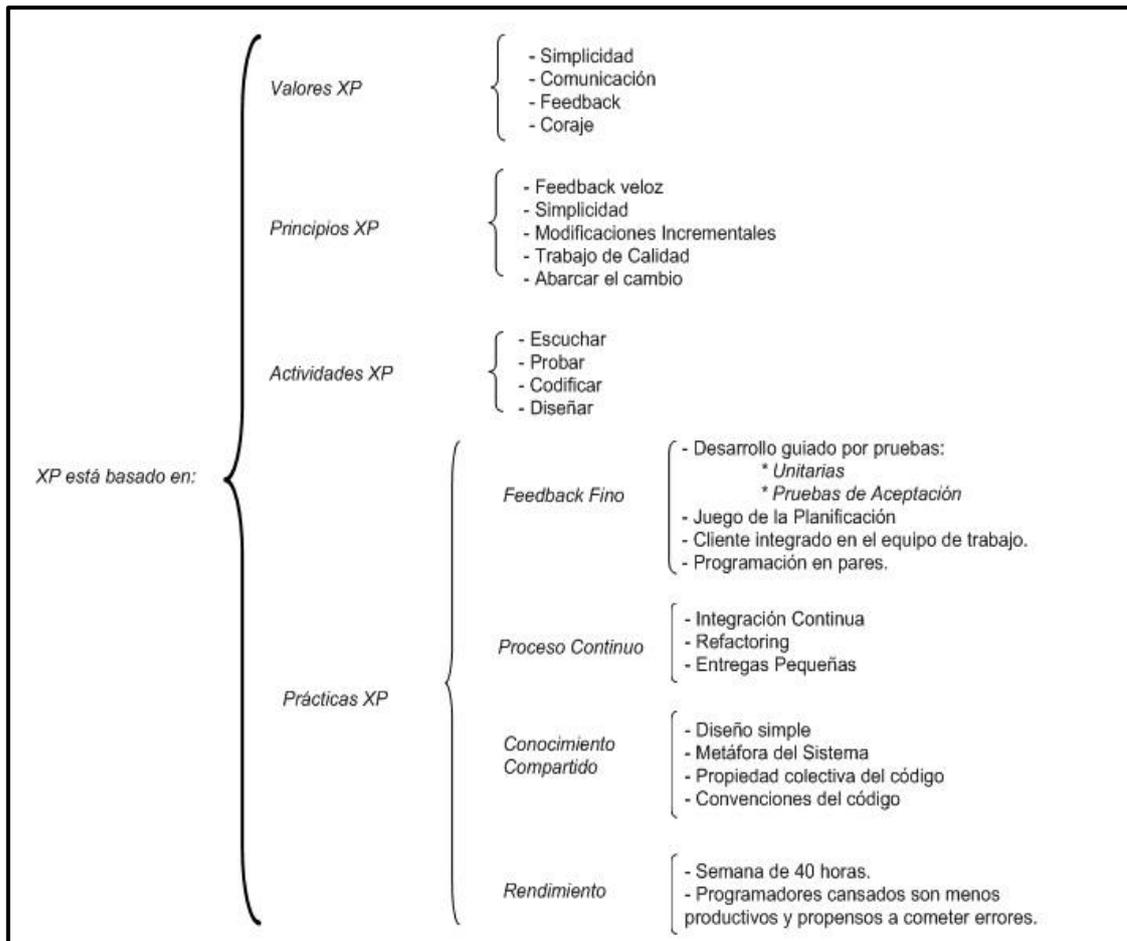


Figura 2.4: Partes fundamentales en las que se basa X.P

ESPACIO EN BLANCO INTENCIONAL

2.4.1.2 Valores de X.P

En la siguiente tabla se muestran los cinco valores de X.P:

Tabla 2.15: Valores de X.P

Valores	Descripción
Simplicidad	Siempre se intenta simplificar el diseño con lo cual se agiliza el desarrollo y facilita el mantenimiento. Para mantener la simplicidad es necesario refactorizar el código a medida que este va creciendo, también se aplica el valor de simplicidad al momento de documentar, haciéndolo en su justa medida.
Comunicación	Constituye un valor fundamental de esta metodología. Es necesario establecer una comunicación con el cliente ya que también formará parte del equipo de desarrollo, además con la ayuda de la programación en pareja se logra mayor interacción entre los desarrolladores del proyecto.
Retroalimentación	Se ve aplicada en distintas dimensiones o etapas del desarrollo de software: Retroalimentación del Sistema: Al realizar pruebas unitarias o de integración, el desarrollador realiza una retroalimentación directa con el resto del equipo, indicando el estado que tiene el sistema tras haber realizado algún cambio. Retroalimentación del Cliente: Las pruebas de aceptación son escritas por el cliente y por los testers. Con esta práctica, ambos miembros del equipo reciben una retroalimentación concreta. Retroalimentación del equipo de desarrollo: Si el cliente viene al equipo de desarrollo con nuevos requerimientos, el equipo estima el tiempo que tardará en implementarlos.
Coraje	El desarrollador debe ser capaz de trabajar rápidamente y rediseñar si es necesario, teniendo como soporte las pruebas de unidad y de aceptación. El desarrollador tiene la libertad para reconstruir código, modificarlo o desechar código obsoleto sin tomar en cuenta cuánto esfuerzo le costó realizarlo; si con ello se asegura que futuros cambios se implementarán fácilmente.
Respeto	El respeto tiene como base los 4 valores ya mencionados. Los miembros del equipo de desarrollo se respetan en el sentido de que no deben realizar cambios al sistema que haga fallar las pruebas de unidad existentes, o retraiga el trabajo de sus compañeros.

2.4.1.3 Principios de X.P

Esta metodología está tiene cinco principios fundamentales, que toman como base los valores antes descritos.

Tabla 2.16: Principios de X.P

Principio	Descripción
Feedback veloz	Aplicación de pequeños ciclos de retroalimentación.
Asunción de la Simplicidad	Ver al problema como una pieza fácil de resolver.
Modificaciones incrementales	Las modificaciones incrementales resuelven problemas realizando cambios pequeños. Se aplica a las fases de: planificación, desarrollo y diseño del proyecto.
Trabajo de Calidad	X.P eleva la calidad del código generado, por medio de pruebas unitarias y de integración antes de realizar una entrega.
Abarcar el cambio	Los Tener distintas posibles viables, mientras se resuelven problemas que son críticos para la entrega.

2.4.1.4 Actividades de X.P

Tabla 2.17: Actividades de X.P

Actividad	Descripción
Escuchar	Escuchar a los clientes y principalmente cuales son los problemas de su negocio, se debe mantener siempre una escucha activa, además de explicar lo que es fácil y difícil de obtener, y la realimentación entre ambos permite a todos a entender los problemas.
Probar	Todas las características del software deben ser demostradas mediante pruebas. Las pruebas brindan la oportunidad de saber si lo que se implementa es lo que en un principio se entendió con las historias de usuario.
Codificar	Sin código fuente no hay programa. Por tanto se necesita codificar y crear nuestras ideas a través del código, ya que al basarse XP en la reutilización esta carecería de sentido si no hay código que reutilizar.
Diseñar	El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben ser sencillos, si alguna parte del sistema es de desarrollo complejo, dividirla en varias. Si hay fallos en el diseño o malos diseños, estos deben de ser corregidos cuanto antes.

2.4.1.5 Prácticas de X.P

Las actividades de X.P. se ven reflejadas en la ejecución de las doce prácticas. A continuación se muestra de forma resumida las 12 prácticas en las que está basada la metodología X.P.

Tabla 2.18: Prácticas de X.P

Práctica	Descripción
Pequeñas entregas	Entrega continua de prototipos del sistema.
Planificación	El usuario escribe sus necesidades, y define las actividades que realizará en el sistema. Se crean las Historias de Usuario ⁸ . Entre 15 y 45 historias de usuario –dependiendo de la complejidad del sistema-, se consideran suficientes para formar el Plan de Iteración, que define los tiempos de entrega de la aplicación para recibir retroalimentación por parte del usuario.
Metáfora	Términos y lenguaje utilizado para describir partes importantes del proyecto.
Diseño simple	El diseño crea una estructura que organiza la lógica del sistema, un buen diseño permite que el sistema crezca con cambios en un solo lugar. Los diseños deben ser sencillos, si alguna parte del sistema es de desarrollo complejo, dividirla en varias. Si hay fallos en el diseño o malos diseños, estos deben de ser corregidos cuanto antes.
Pruebas	Pruebas unitarias continuas, frecuentemente repetidas y automatizadas, incluyendo pruebas de regresión. Se aconseja escribir el código de la prueba antes de la codificación. Además se debe establecer un periodo de pruebas de aceptación del programa donde se definirán las entradas al sistema y los resultados esperados de estas entradas.
Refactorización	Refactorización del código, es decir, reescribir ciertas partes del código para aumentar su legibilidad y mantenibilidad pero sin modificar su comportamiento. Se debe aplicar durante todo el proceso X.P. Para mejorar el código sin cambiar la funcionalidad, eliminar la complejidad y el código duplicado.
Programación en pareja	Se recomienda que las tareas de desarrollo se lleven a cabo por dos personas en un mismo puesto. Se supone que la mayor calidad del código escrito de esta manera –el código es revisado y discutido mientras se escribe- es más importante que la posible pérdida de productividad inmediata. Esto puede producir aplicaciones mejores, con iguales o menores costos

⁸ Historia de Usuario: Representación de un Requerimiento de Software en una o dos frases utilizando el lenguaje común del usuario.

Propiedad colectiva	Permite que un miembro del equipo de desarrollo pueda cambiar cualquier parte del código cuando lo amerite.
Integración continua	Con la integración continua se logra que los componentes del sistema sean integrados varias veces al día.
40 horas laborales	X.P exige un horario de trabajo de 40 horas semanales; para el buen desempeño del desarrollador.
Cliente en el sitio	(Cliente integrado) El cliente debe determinar los requerimientos, definir la funcionalidad, priorizar los requerimientos más urgentes, responder a las dudas de los programadores. El cliente estará con el equipo de trabajo durante la realización de todo el proyecto, por lo tanto, se disminuye la cantidad de documentación, junto con los altos costos de su creación y mantenimiento.
Estandarización	Grupo de normas y convenciones que el equipo de desarrollo debe cumplir.

2.4.1.6 Ciclo de Vida de La Metodología X.P

En la siguiente tabla se detalla el ciclo de vida de esta metodología.

Tabla 2.19: Ciclo de Vida de X.P

Práctica	Descripción
Exploración	Corresponde a la inicialización del proyecto. El cliente plantea las Historias de Usuario. El equipo de desarrollo selecciona las herramientas, tecnologías que utilizará en el proyecto y realiza un prototipo de la arquitectura que tendrá el sistema.
Planificación	Se priorizan las Historias de Usuario, se realiza una estimación de la duración requerida para implementar dichas funcionalidades, se organiza la cultura de trabajo y la planificación detallada dentro de cada versión, se crea un plan de entrega, se controla la velocidad del proyecto, se divide el proyecto en iteraciones.
Iteraciones	Esta fase abarca el desarrollo y las pruebas del sistema. Se debe escribir código acorde a los estándares, el cliente debe estar disponible para ayudar al equipo de desarrollo y para formar parte del mismo, se debe desarrollar la unidad de pruebas primero para que ayude a tener una visión acerca del comportamiento del programa, y se deben ejecutar pruebas de aceptación a menudo; hasta que no se hayan pasado estas pruebas, una historia de usuario no se considera lista. Al final de la última iteración el sistema estará listo para entrar en producción.
Producción	Puesta en producción del proyecto de software en el entorno del cliente. Las ideas y sugerencias propuestas son documentadas para una implementación posterior (en la fase de mantenimiento).
Mantenimiento	Mientras una versión del sistema está en Producción, el proyecto X.P debe mantener en funcionamiento el sistema al mismo tiempo que se desarrollan nuevas iteraciones, actualizaciones para mejorarlo.

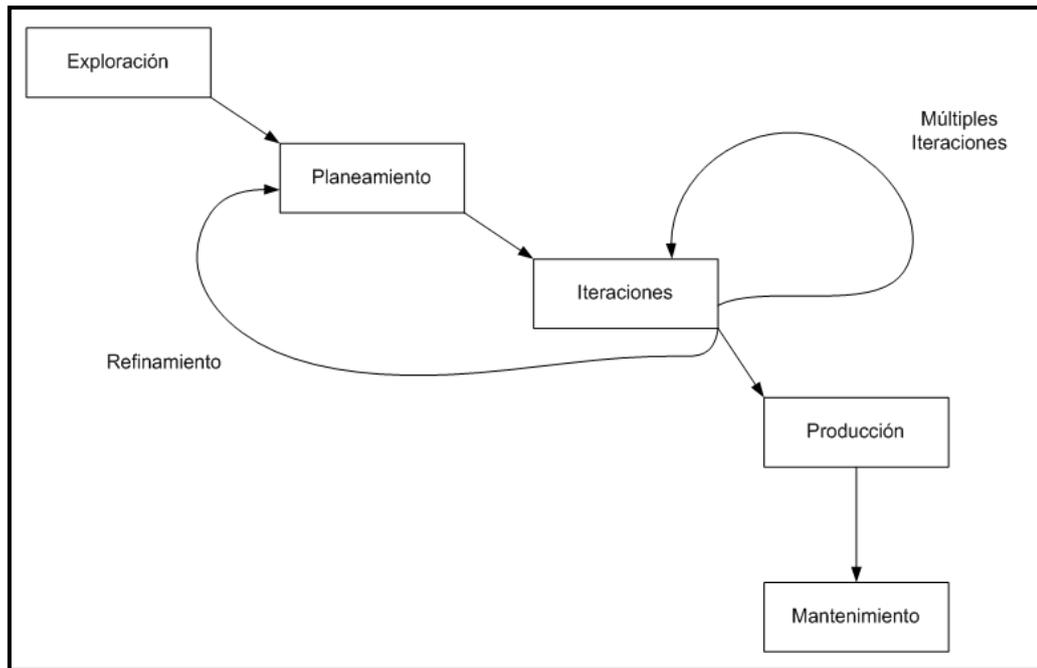


Figura 2.5: Ciclo de vida de un proyecto X.P.⁹

En la Figura 2.5 se ilustra de forma resumida el ciclo de vida de Extreme Programming. Luego de la etapa de Exploración, el equipo empieza con un ciclo de Planeamiento y múltiples iteraciones, hasta que el software se considere listo para entrar a la etapa de Producción, finalmente, el sistema entra en una etapa de mantenimiento, en el que se producen mejoras y actualizaciones al sistema.

Para el caso de este proyecto de tesis, se llegará hasta la fase de Iteraciones, que implica el desarrollo/implementación, pruebas unitarias, de integración y de aceptación del sistema; ya que para la fase de puesta en producción (implantación del sistema), está valorado como un proyecto aparte de la empresa, lo cual fue acordado por ambas partes.

⁹ Figura 1. Ciclo de vida de X.P. Adaptado de "Entrega Ágil Disciplinada", por Ambler S y Lines M. Recuperado de <http://sg.com.mx/revista/entrega-%C3%A1gil-disciplinada>.

Un punto relevante cuando se aplica esta metodología es la planificación del proyecto X.P, misma que consiste en un proceso evolutivo que es refinado a lo largo de todo el ciclo de vida del proyecto. Con la Figura 2.6 se trata de ilustrar este concepto.

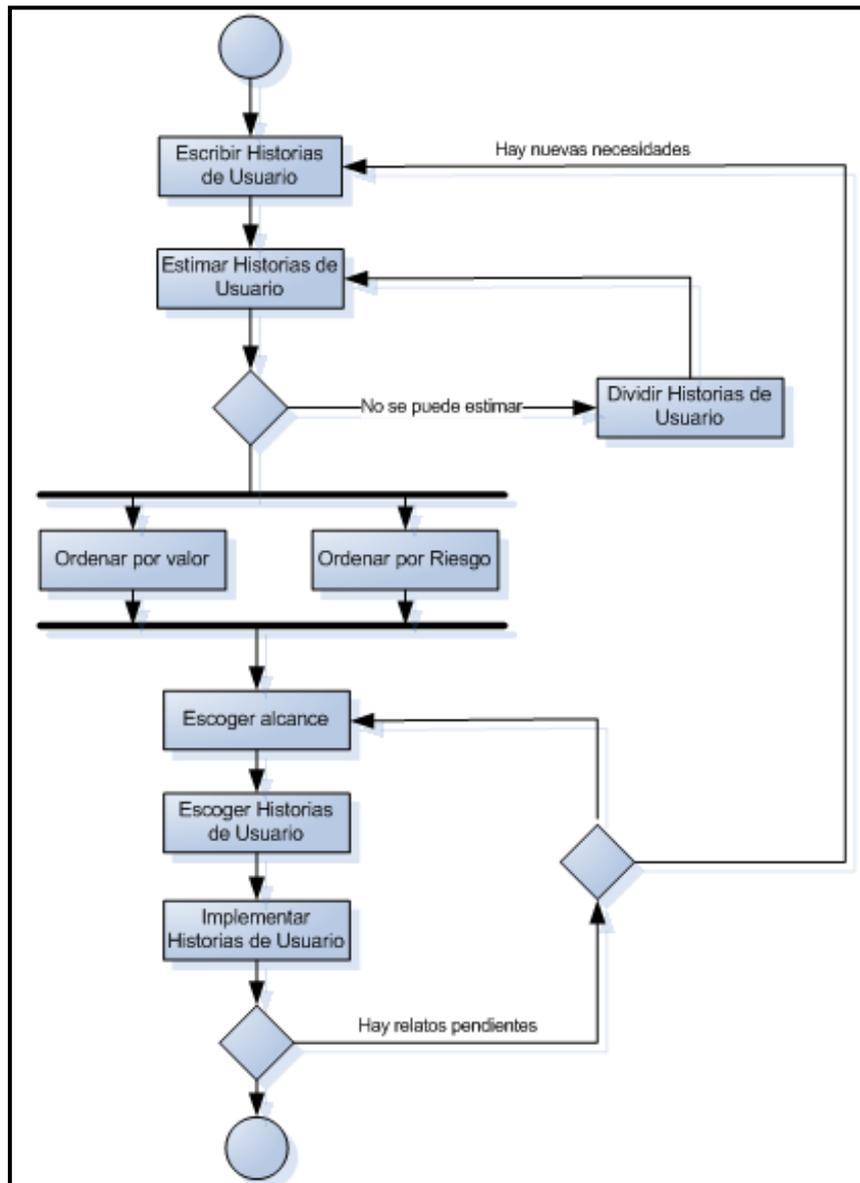


Figura 2.6: El flujo del Juego de la Planificación

2.4.1.7 Roles en X.P

La definición de roles con esta metodología, no son estáticos, dado que en un momento determinado (dependiendo de la necesidad) un integrante del grupo de trabajo puede asumir un cargo distinto, o desempeñar varios cargos al mismo tiempo; y para esto se vale de la colaboración efectiva que asume como compromiso cada uno de los integrantes.

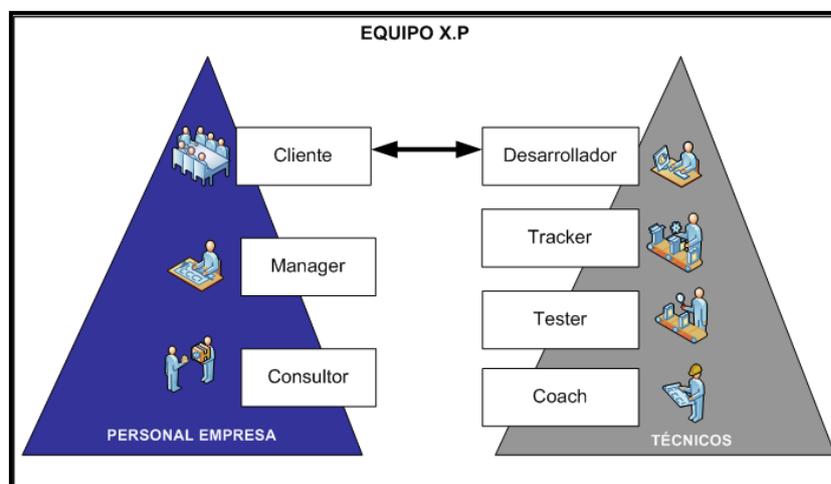


Figura 2.7: Roles en X.P

En la Figura 2.7 se muestran los roles que pueden existir en un proyecto X.P:

Cliente: Persona o grupo de personas que tienen sólidos conocimientos de los procesos de negocio. Encargado de:

- Escribir las Historias de Usuario.
- Escribir/especificar las pruebas de aceptación.
- Elaborar conjuntamente con el equipo de desarrollo el Plan de Entregas de prototipos del sistema.

Manager: Persona encargada de la gestión del proyecto internamente y hacia los clientes. Encargado de:

- Aclarar interferencias desde el punto de vista del usuario que obstruyan el trabajo del equipo de desarrollo.

Consultor: Experto externo, especialista en solventar inquietudes sobre temas técnicos o de negocio.

Desarrollador: Encargado de:

- Estimar las Historias de Usuario.
- Implementar las Historias de Usuario.
- Escribir pruebas unitarias.
- Participar en reuniones para la planificación.

Tracker: Encargado de:

- Usar métricas para medir el avance del proyecto. Por ejemplo: Tiempo de desarrollo / Tiempo Calendario.
- Ayudar al Coach a motivar el cambio de manera gentil y no restrictiva con su equipo de trabajo.

Tester: Encargado de:

- Realizar las pruebas de aceptación conjuntamente con el cliente estableciendo reuniones periódicamente.

Coach: Entrenador. Encargado de:

- Facilitar la comunicación dentro del equipo de desarrollo, de manera que se pueda mantener un correcto vínculo entre éste.
- Explicar el proceso a gerentes de mayor nivel.
- Proveer de golosinas y juguetes 😊.

2.4.1.8 ¿Cuándo no implantar X.P?

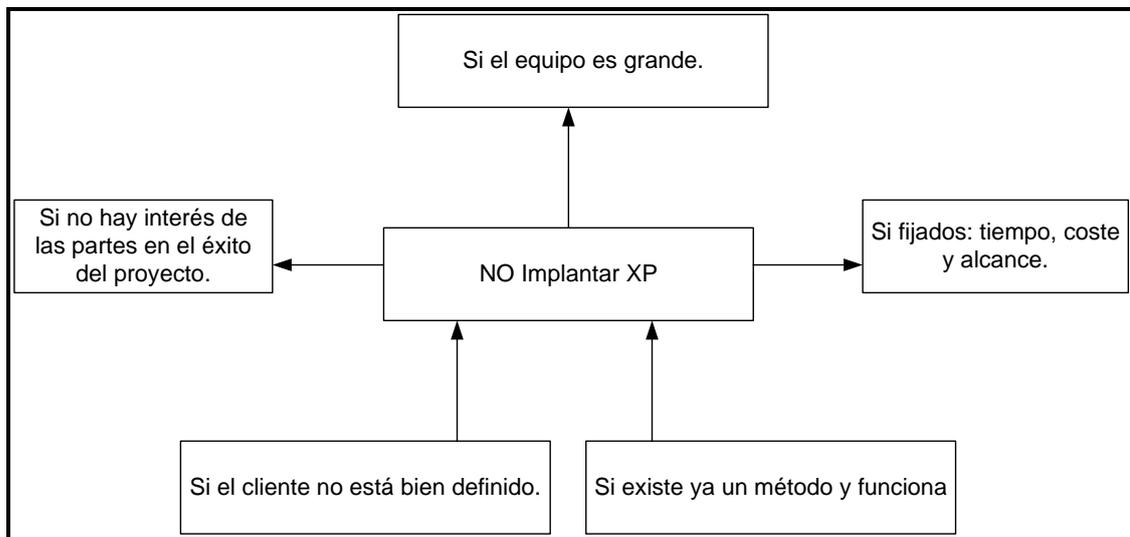


Figura 2.8: Cuando no implantar X.P

2.5 Selección de las herramientas de desarrollo

De manera resumida se presenta en la siguiente tabla todas las herramientas de Software Libre utilizadas para el desarrollo del presente proyecto y su correspondiente descripción.

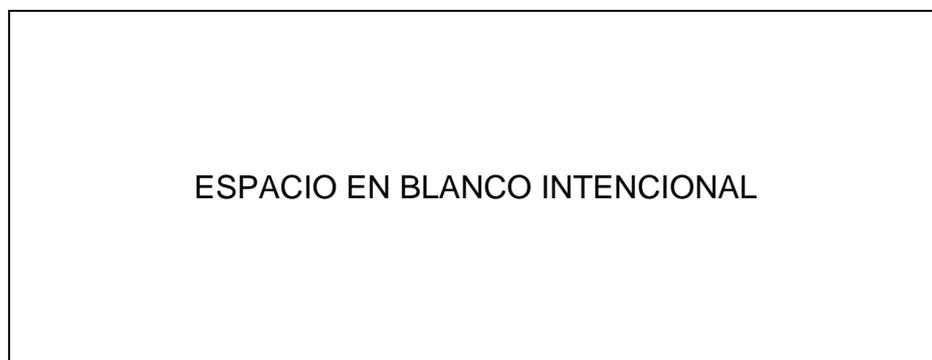


Tabla 2.20: Herramientas de Software Libre utilizadas

Herramienta	Descripción
JDK 1.7: Java Development Kit	Conjunto de herramientas de desarrollo para la creación, depuración y control de programas en lenguaje JAVA. ¹⁰
Eclipse Indigo 3.7 SR2	Es un IDE de código abierto y multiplataforma que proporciona un conjunto completo de herramientas de programación; además de un excelente ambiente para desarrolladores de aplicaciones integradas. ¹¹
JBoss AS 6.1	Es un servidor de aplicaciones desde Java2EE en adelante, de código abierto, que corre sobre cualquier sistema operativo para el que esté disponible la máquina virtual de Java, implementa por defecto las especificaciones: EJB 3.0, JPA 2.0 y Hibernate ¹² .
PostgreSQL 8.4	Es un sistema gestor de bases de datos objeto-relación, de código abierto; utiliza el modelo cliente/servidor y uso de multiprocesos, lo que lo vuelve uno de los más robustos y estables. PostgreSQL maneja sin problema grandes cantidades de datos, además de soportar una alta concurrencia de usuarios ¹³ .
Primefaces 3.2	Es una librería de componentes visuales Open Source para JSF, que permite la creación o personalización de aplicaciones web; funciona de manera conjunta con tecnologías como Ajax y JPA, además permite a la aplicación ser ejecutada en dispositivos móviles sin programación extra.
DSpace 1.8.2	Es un sistema de repositorio digital, de código abierto, que permite recoger material digital, distribuirlos sobre la web a través de un sistema de búsqueda y recuperación; así como preservarlos a largo plazo.
Jasper Reports 4.7.1	Es una herramienta de creación de informes de tipo dinámico y permite generar ficheros en varias extensiones (PDF, HTML, XLS, CSV y XML) y una impresión de una forma simple y flexible ¹⁴ . Puede ser usado en cualquier aplicación Java, se usa comúnmente con un front-end gráfico de código abierto para la edición de informes, en el presente proyecto iReport-4.7.1
Visual Paradigm for UML Standard	Es una herramienta para modelado UML de datos, arquitectura, diagramas de Entidad-Relación entre, otros; de manera fácil y ágil.

¹⁰ Richardson, C., Avondolio D., Scanlon J., Mitchell M., Professional Java JDK. [PDF]. Toronto. 2007

¹¹ Daum, B. Professional Eclipse for Java Developers. [PDF]. Indiana, 2005

¹² Marchioni, F. Jboss AS Development. [PDF]. Toronto 2009

¹³ The PostgreSQL Global Development Group, *PostgreSQL 8.4 Documentation*. [PDF]. Boston. 2010

¹⁴ Jaspersoft, The JasperReports Ultimate Guide. Third Edition. [PDF]. U.S.A. 2010

2.5.1 Descripción de Las Herramientas de Desarrollo

A pesar de que todas las herramientas descritas en la tabla 2.20 tienen un valor trascendental dentro del sistema; resulta exigible que para mejor entendimiento, se amplíe la explicación de las herramientas que han sido base fundamental para la construcción de esta aplicación. Estas son: Eclipse, DSpace y PostgreSQL.

2.5.1.1 Eclipse

Eclipse es un entorno de desarrollo integrado de código abierto multiplataforma para desarrollar “Aplicaciones de Cliente Enriquecido”, opuestas a las aplicaciones “Cliente-liviano” basadas en navegadores. Eclipse es un marco y un conjunto de servicios para construir un entorno de desarrollo a partir de componentes conectados (plug-in).

Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (IDEs), como el IDE de Java llamado Java Development Toolkit (JDT) y el compilador (ECJ) que se entrega como parte de Eclipse (y que son usados también para desarrollar el mismo Eclipse). Sin embargo, también se puede usar para otros tipos de aplicaciones cliente, como BitTorrent o Azureus.

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que

fomenta una Comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios.

Eclipse fue liberado originalmente bajo la Common Public License, pero después fue re-licenciado bajo la Eclipse Public License. La Free Software Foundation ha dicho que ambas licencias son licencias de software libre, pero son incompatibles con Licencia pública general de GNU (GNU GPL). Hasta el momento, la última versión estable de Eclipse es la 4b Juno SR2. Para la realización de este proyecto se utilizó Eclipse Indigo 3.7 SR2

2.5.1.2 DSpace

DSpace es un software de código abierto desarrollado por las Bibliotecas del MIT (Massachusetts Institute of Technology) y Hewlett Packard Labs. Es un sistema de repositorio que permite recoger, describir documentos digitales. Y puede ser personalizado según las necesidades.

El sistema quedó disponible públicamente en Noviembre de 2002 bajo licencia BSD Open Source. Este software fue desarrollado utilizando normas y estándares, por lo tanto, le permite integrarse fácilmente a otros sistemas.

DSpace es un repositorio que preserva contenidos digitales a largo plazo. Es un archivador estandarizado de documentos con un sistema de indexación y búsqueda sobre metadatos (conjunto de atributos que sirven para definir la información que se almacena dentro de un recurso de tal manera que pueda ser catalogado y categorizado) y en el texto completo.

La manera en la que se organiza la información en DSpace pretende reflejar

la estructura de una organización. Cada sitio que utilice DSpace está dividido en Comunidades, las cuales pueden ser divididas también en Subcomunidades que reflejan la típica estructura de la facultad, el departamento, centro de investigación o laboratorio de una universidad.

Las Comunidades contienen Colecciones, las cuales son agrupaciones de contenidos relacionados. Estas Colecciones pueden aparecer en más de una Comunidad.

Cada Colección está compuesta por Ítems, los cuales son los elementos básicos del archivo. Cada Ítem es propiedad de una Colección. Cada Ítem puede aparecer también en otras Colecciones, sin embargo solo puede pertenecer a una Colección.

Los Ítems también pueden estar subdivididos en paquetes de bitstreams. Estos bitstreams son flujos de bits, normalmente archivos. Los bitstreams que de alguna manera están relacionados, como por ejemplo los archivos HTML y las imágenes que componen un documento HTML, están organizados en paquetes.¹⁵ (Chazarra, Requena y Valverde, 2010, p. 17-18).

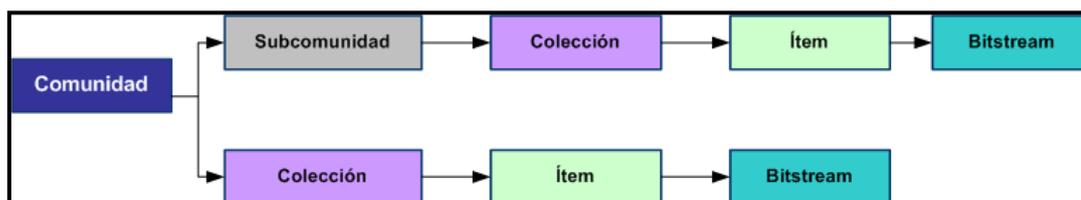


Figura 2.9: Forma de organizar la información en DSpace

En la siguiente figura se puede apreciar el funcionamiento del repositorio:

¹⁵ Adaptado de "Desarrollo de un repositorio de objetos de aprendizaje usando DSpace." Por Chazarra J., Requena V. y Valverde S. Recuperado de <http://eprints.ucm.es/11078/1/MemoriaSI>

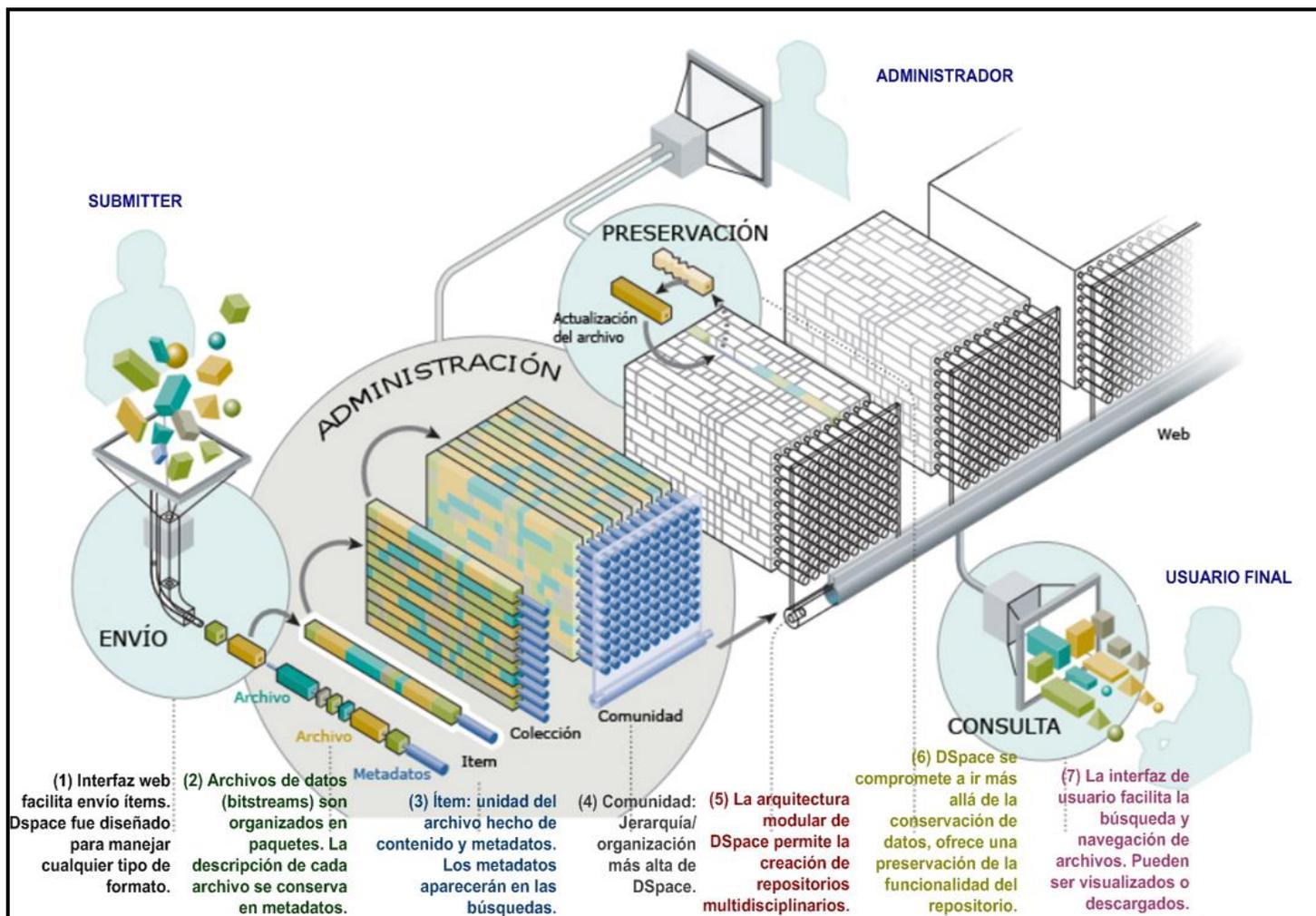


Figura 2.10: Funcionamiento de DSpace

2.5.1.2.1 Beneficios de utilizar DSpace

Visibilidad: Optimiza la difusión de documentos.

Permanencia: Estabilidad en la localización, mediante la asignación de una dirección única en Internet.

Flexibilidad: Soporte de diferentes formatos de archivo, inclusive hojas de cálculo, presentaciones, imágenes, videos, programas, etc.

Almacenamiento seguro: Garantiza la perdurabilidad de los documentos ingresados, mediante mecanismos estándares de preservación digital.

Derechos de Uso: Permite definir el grado de uso que pueden hacer otras personas de los documentos ingresados; determinando si se permite el uso comercial o modificaciones del documento.

Control de acceso: El administrador de una Colección determina el nivel de acceso a los documentos, tratándolos como públicos o privados.

2.5.1.2.2 Ventajas Técnicas de utilizar DSpace

- Código limpio, utiliza estándares.
- Código bien comentado, facilita entender lo que el desarrollador ha codificado, permitiendo modificar/corregir con mayor facilidad.
- Arquitectura en capas.
- Utilización del motor de base de datos PostgreSQL, que es una poderosa, robusta herramienta.

2.5.1.3 PostgreSQL

Es un sistema de gestión de bases de datos objeto-relacional, distribuido bajo licencia BSD¹⁶ y con su código fuente disponible libremente.

PostgreSQL hace uso de un modelo cliente-servidor y utiliza multiprocesos que garantizan la estabilidad del sistema (si uno de los procesos fallara, esto no afectaría el resto de procesos y el sistema continuaría funcionando).

Componentes más importantes de un sistema PostgreSQL:

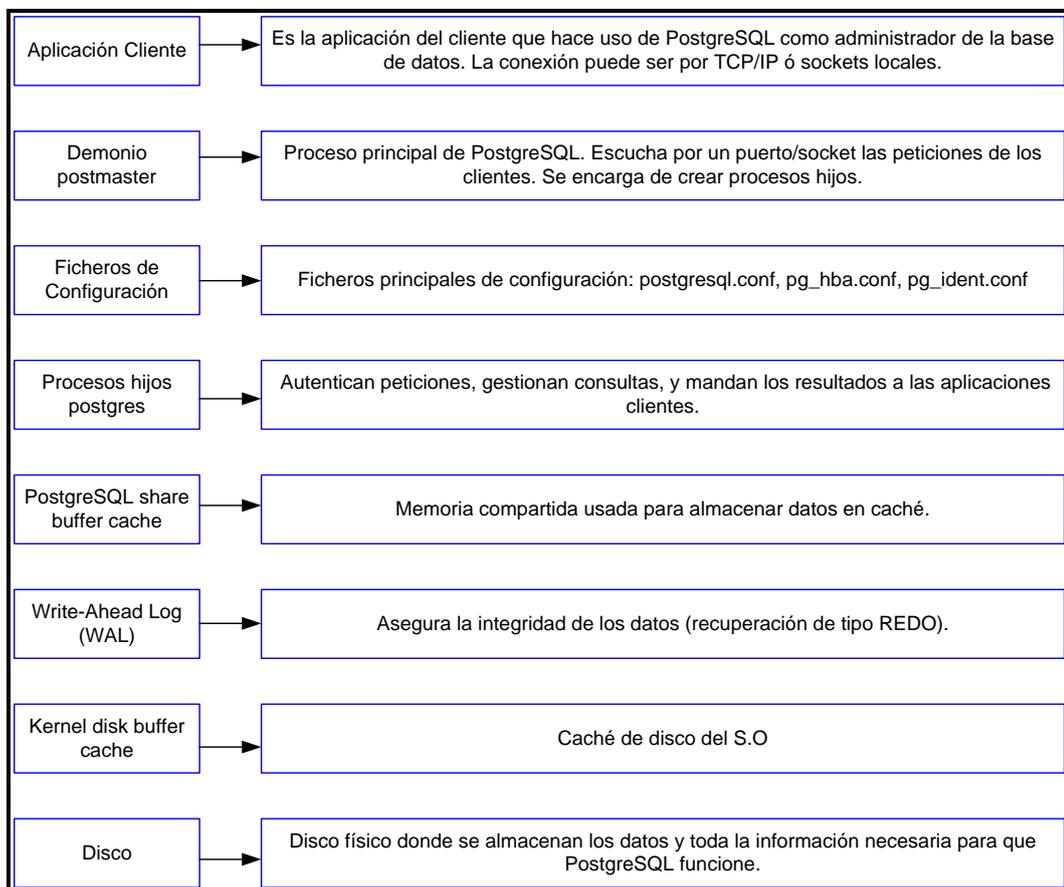


Figura 2.12: Componentes PostgreSQL

¹⁶ Licencia BSD: Licencia de software otorgada principalmente para los sistemas BSD (Berkeley Software Distribution). Esta licencia tiene menos restricciones en comparación con otras como la GPL estando muy cercana al dominio público. La licencia BSD al contrario que la GPL permite el uso del código fuente en software no libre.

2.5.1.3.1 Características PostgreSQL

2.5.1.3.1.1 Características Generales

- Es una base de datos 100% ACID¹⁷
- Integridad referencial
- Disponible para Linux y UNIX en todas sus variantes (AIX, BSD, HP-UX, SGI IRIX, Mac OS X, Solaris, Tru64) y Windows 32/64bit.
- Tablespaces
- Replicación asincrónica/sincrónica/Streaming replication – Hot Standby
- PITR – point in time recovery
- Copias de seguridad en caliente (Online/hot backups)
- Unicode¹⁸
- Multi-Version Concurrency Control (MVCC)
- Múltiples métodos de autenticación
- Documentación completa
- Licencia BSD

¹⁷ ACID: (Atomicity, Consistency, Isolation and Durability). conjunto de características necesarias para que una serie de instrucciones puedan ser consideradas como una transacción.

¹⁸ Unicode: Estándar de codificación diseñado para facilitar el tratamiento informático, transmisión y visualización de textos de múltiples lenguajes y disciplinas técnicas además de textos clásicos de lenguas muertas. El término Unicode proviene de los tres objetivos perseguidos: universalidad, uniformidad y unicidad.

CAPÍTULO 3: ANÁLISIS DEL PROYECTO

El análisis del proyecto informático permite programar el tiempo, los recursos y alcances para cada etapa de desarrollo, de manera que se establezcan los objetivos basados en un plan de entrega previamente analizado. La especificación de requerimientos del sistema permite ver las necesidades y tener objetivos claros. Dichos requerimientos del usuario se encuentran plasmados en las Historias de Usuario. La programación del tiempo y recursos a emplear en cada Historia de Usuario, están proyectados en el plan de entrega.

3.1 Línea Base y Estado del Arte

Luego de una indagación y síntesis de los antecedentes de los problemas que actualmente posee la empresa Crosstronik Cía. Ltda., se determina los siguientes puntos con respecto al estado del arte, estos son:

- La empresa no dispone actualmente de un sistema para ninguno de los procesos de inventario, ni de administración de sus productos; con respecto al módulo de gestión contable, Plan de Cuentas y libros contables los generan con el programa T-Max.
- Actualmente todos los documentos que intervienen en el proceso de generación de Órdenes de Trabajo y Cotizaciones son realizados en papel o en hojas de cálculo con formatos predefinidos por la empresa, no existe ningún tipo de respaldo a parte del documento original. Además

estos procesos muestran falencias con respecto al control y evidencias de su ejecución.

- Los procesos contables se los realiza únicamente los fines de semana, por motivos de falta de personal especializado en el tema, y de una herramienta que permita realizarlos diariamente de manera correcta.
- En la empresa actualmente se ocupa el correo electrónico a forma de repositorio de todos los reportes técnicos que generan sus empleados en todas sus actividades.

3.1.1 Producción de Información vs Necesidades y Uso

Es necesario generar una herramienta que permita mejorar los procesos principales dentro de la cadena de valor de la empresa; por tanto se ve necesario la automatización de dichos procesos sensibles de manera que el usuario pueda estar al tanto del flujo de uno de sus trabajos. Con respecto a las necesidades actuales en el campo contable, la herramienta debe ofrecer al usuario la capacidad de conocer saldos de sus cuentas desde el inicio del año contable de manera que se pueda hacer un seguimiento adecuado a las fases que dichas cuentas tienen y que permitan a los gerentes tomar decisiones correctas en base a los reportes obtenidos del sistema.

Finalmente se establece que la gestión de inventarios necesita redefinir sus procesos pues actualmente no se dispone de ninguna herramienta que permita conocer en tiempo real los stocks de las distintas bodegas.

3.1.2 Capacitación y Aspectos Organizativos

Se ha identificado que dentro del sistema deben existir perfiles, debido principalmente a la organización que tiene la empresa, las distintas responsabilidades que tienen los empleados en esta, y el nivel de conocimiento requerido para poder manejar la información con la que trata cada día.

Es por esto que es necesario realizar una capacitación al personal para el uso del sistema sobre todo a nivel modular ya que a nivel de interfaz el sistema no presentaría ningún inconveniente.

3.2 Apreciación General

El sistema a desarrollar abarca cuatro módulos:

- Módulo administración de usuarios y control de accesos.
- Módulo de Inventario que cubre tanto el control de las entradas/salidas de los productos existentes en cualquier bodega de la empresa, como la automatización de los procesos de generación de Órdenes de Trabajo y Cotizaciones.
- Módulo Contable que cubre la gestión de procesos de apoyo Contable siendo principalmente la generación de Comprobantes de Ingreso, Egreso, Transacciones, Facturas, y Reportes Contables (Libro Diario, Libro Mayor, Balance de Comprobación, Estado de Resultados).
- Módulo para la gestión de documentos –informes técnicos- de forma que sean accesibles en todo momento a través de la web. Para

solventar este requerimiento se utiliza DSpace que es una aplicación, repositorio digital de documentos (libros, informes, etc.) que facilita la publicación, búsqueda, descarga de documentos mediante internet.

3.3 Usuarios del Sistema

Se considera como usuario a la o las personas que se conecten al sistema para hacer uso de los servicios que este les proveerá. Se distinguirán diferentes perfiles o niveles de usuario, y dependiendo del nivel poseerá más o menos privilegios durante su estancia dentro del sistema.

- **Superadministrador:** Este usuario tiene la función de administrar y configurar los privilegios que puede tener un usuario dentro del sistema. El superadministrador tiene acceso y máximos privilegios dentro de los módulos del sistema: Inventario, Gestión Contable, Gestión de Usuarios y control de Accesos.
- **Administrador:** Tiene acceso y máximos privilegios en los módulos de Inventario, Gestión Contable y Repositorio Digital de Informes técnicos.
- **Técnico:** Tiene acceso al módulo de Inventario, secciones: Administración de Clientes, Administración de Proyectos, Requisición de Bodega, Búsqueda Avanzada y Reportes.
- **Bodeguero:** Tiene acceso al módulo de Inventario, secciones: Administración de Producto, Administración de Inventario, Administración de Ubicación, Búsqueda Avanzada y Reportes.
- **Contador:** Tiene máximos privilegios en el módulo de Gestión Contable.

- **Cliente:** Tiene acceso al módulo de Repositorio Digital de Informes técnicos. Puede visualizar y descargar los informes técnicos publicados para la empresa a la que pertenece, no posee privilegios para enviar/subir un archivo.
- **Anónimo:** Tiene acceso restringido al sistema (privilegios mínimos). Visualiza únicamente la página principal de la empresa, para conocer datos informativos.

3.4 Requisitos Funcionales

Los requisitos funcionales expresan la naturaleza del sistema, es decir, la interacción del sistema con su entorno, su funcionamiento y cuál va ser su estado.

Tomando en cuenta lo que plantea la metodología X.P se hizo uso de un formato de Historias de Usuario para especificar los requerimientos en la situación actual.

ESPACIO EN BLANCO INTENCIONAL

3.4.1 Historias de Usuario módulo: Administración de Usuarios

Tabla 3.1: Historia de Usuario I: Administración de Usuarios

HISTORIA DE USUARIO	
Número: 1	Nombre: Administración de Usuarios
Usuario: Superadministrador	Riesgo en desarrollo: Medio
Prioridad en negocio: Alta	Puntos estimados:0.8
Descripción: <ul style="list-style-type: none">- Como superadministrador, debo tener la facultad para crear, modificar, habilitar o deshabilitar usuarios, asignar permisos, que les permitirá ingresar solo a determinadas secciones del sistema, no a todas.	
Observación:	

Tabla 3.2: Historia de Usuario II: Ingreso y utilización del sistema

HISTORIA DE USUARIO	
Número: 2	Nombre: Ingreso y utilización del sistema
Usuario: Todos	Riesgo en desarrollo: Medio
Prioridad en negocio: Alta	Puntos estimados:0.8
Descripción: <ul style="list-style-type: none">- El sistema debe realizar una validación del usuario y de los permisos que le han sido asignados, permitiéndole ingresar a las secciones cuyo perfil establezca.	
Observación: <ul style="list-style-type: none">- El contenido referente a los servicios que presta la empresa, estará disponible en la página principal de la empresa, por lo tanto cualquier usuario podrá visualizarla sin necesidad de validarse.	

ESPACIO EN BLANCO INTENCIONAL

3.4.2 Historias de Usuario módulo: Repositorio digital de informes técnicos

Tabla 3.3: Historia de Usuario III: Publicación de informes técnicos en la web

HISTORIA DE USUARIO	
Número: 3	Nombre: Publicación de informes técnicos a través de la web.
Usuario: Administrador, Cliente	Riesgo en desarrollo: Alto.
Prioridad en negocio: Alta	Puntos estimados:0.8
Descripción: <ul style="list-style-type: none"> - Publicar/subir documentos en formato pdf referentes a los informes técnicos que se generan de forma semanal y mensual; que estén disponibles a través de la página web de la empresa para el personal que cuente con los privilegios de acceso respectivos: <ul style="list-style-type: none"> o Solo el Administrador tendrá permisos para subir, descargar, o visualizar los documentos de cualquier cliente. o El Cliente tendrá permisos de lectura y descarga de los informes técnicos que Crosstronik publique para su empresa. 	
Observación:	

3.4.3 Historias de Usuario módulo: Gestión de Inventario

Tabla 3.4: Historia de Usuario IV: Control de la entrada de productos en bodegas

HISTORIA DE USUARIO	
Número: 4	Nombre: Control de la entrada de productos en cualquier bodega de la empresa
Usuario: Administrador, Bodeguero	Riesgo en desarrollo: Medio
Prioridad en negocio: Media	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Dar de alta, y modificaciones a los registros de materiales, existentes en una base de datos. - La información sobre el manejo de Inventario debe estar actualizada y disponible desde el sitio web de la empresa; para el personal que cuente con los privilegios de acceso respectivos. 	
Observación: Se llevará un control de las transacciones (alta, bajas y modificaciones) de los materiales mediante un kárdex de producto por ubicación.	

Tabla 3.5: Historia de Usuario V: Control de salidas de producto en bodegas

HISTORIA DE USUARIO	
Número: 5	Nombre: Control de la salida de productos en cualquier bodega de la empresa y generación del formato de Requisiciones de Material
Usuario: Administrador, Bodeguero, Técnico	Riesgo en desarrollo: Alta
Prioridad en negocio: Media	Puntos estimados: 0.2
Descripción:	
<ul style="list-style-type: none"> - Generar el formato "Requisición de Bodega", donde se registre la salida de productos de cualquier bodega de la empresa, el motivo por el cual se realiza dicho movimiento y el listado con todos los productos y cantidades solicitados. - En el formato de Requisición de Materiales se detallarán los siguientes campos: Número de requisición, fecha, solicitante, Datos de la bodega, Observaciones; Detalles de los materiales: (Ítem, Código, Descripción, Cantidad, Unidad), Nota y Firma de Responsabilidad. 	
Observación:	

Tabla 3.6: Historia de Usuario VI: Administración Órdenes de Trabajo

HISTORIA DE USUARIO	
Número: 6	Nombre: Administración Órdenes de Trabajo
Usuario: Administrador, Técnico	Riesgo en desarrollo: Alto
Prioridad en negocio: Media	Puntos estimados: 0.2
Descripción:	
<ul style="list-style-type: none"> - Generar el formato perteneciente a una Orden de trabajo; de tal manera que se automatice el recogimiento de la información inicial (requerimientos) para realizar un proyecto. - El formato de Orden de trabajo debe tener los siguientes campos: Número de Orden de Trabajo, Fecha, Tema; Información del cliente (Nombre, e-mail, teléfonos), Observaciones, los archivos que se relacionan con la Orden de Trabajo, Detalles, Costo, Abono, Saldo, Fecha de Entrega, Observaciones de Entrega, y la firma de la persona que entregó la orden. 	
Observación:	
<ul style="list-style-type: none"> - Una Orden de Trabajo no siempre genera una Cotización. 	

ESPACIO EN BLANCO INTENCIONAL

Tabla 3.7: Historia de Usuario VII: Administración Cotizaciones

HISTORIA DE USUARIO	
Número: 7	Nombre: Administración Cotizaciones
Usuario: Administrador, Técnico	Riesgo en desarrollo: Alto
Prioridad en negocio: Media	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Generar el formato perteneciente a una Cotización de Materiales, que estime el costo de un servicio/proyecto que ejecute la empresa para cualquier cliente. - El formato de Cotización debe tener los siguientes campos: Número de Cotización, Fecha, Por (Servicio o productos que se están cotizando), Información del Cliente (Nombre, Dirección, Atención, E-Mail, Teléfonos, Fax/Otros), Detalles, Condiciones Comerciales, Tiempo de Entrega, Forma de Pago, Validez de la Oferta, Observaciones, y la firma de la persona que elabora la cotización. 	
Observación: Una cotización pertenece a una Orden de Trabajo específica.	

3.4.4 Historias de Usuario módulo: Gestión Contable

Tabla 3.8: Historia de Usuario VIII: Elaboración comprobantes de Ingreso

HISTORIA DE USUARIO	
Número: 8	Nombre: Elaboración de Comprobantes de Ingreso
Usuario: Administrador, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Generar de forma automática el Comprobante de Ingreso, donde se describa la fecha, valor, el concepto del valor ingresado, y las firmas de responsabilidad 	
Observación: <ul style="list-style-type: none"> - Este documento contable, puede o no, estar relacionado con una Orden de Trabajo, de manera que se lleve un control de los anticipos que el cliente nos haga por el trabajo que solicita. 	

Tabla 3.9: Historia de Usuario IX: Elaboración comprobantes de Egreso

HISTORIA DE USUARIO	
Número: 9	Nombre: Elaboración de Comprobantes de Egreso
Usuario: Administrador del Sistema, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Generar de forma automática el Comprobante de Ingreso, donde se describa la fecha, a quién se paga, el valor del pago, el concepto del valor que se paga, el valor, y las firmas de responsabilidad 	
Observación:	

Tabla 3.10: Historia de Usuario X: Elaboración de facturas

HISTORIA DE USUARIO	
Número: 10	Nombre: Elaboración de Facturas
Usuario: Administrador del Sistema, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Necesito generar de forma automática la factura acorde al formato que actualmente se maneja en la empresa y los detalles que la conforman. - En la factura se detallarán los siguientes campos: Cabecera: Datos del cliente (Nombre, RUC/C.I, Teléfono, Dirección), Fecha; Cuerpo: (Cantidad, Detalle, Valor Unitario, Valor Total), y en el Pie de la factura: (Subtotal, Valor I.V.A, Total) con las firmas de los responsables. 	
Observación:	

Tabla 3.11: Historia de Usuario XI: Elaboración del Plan de Cuentas

HISTORIA DE USUARIO	
Número: 11	Nombre: Elaboración del Plan de Cuentas
Usuario: Administrador del Sistema, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Que se pueda ingresar el Plan de Cuentas ajustado a las necesidades actuales de la empresa. 	
Observación: <ul style="list-style-type: none"> - Esta plantilla será utilizada en los siguientes documentos: Comprobantes de Ingreso y Egreso, Libro Diario, Mayorización y Balance de Resultados. 	

Tabla 3.12: Historia de Usuario XII: Elaboración del reporte contable Libro Diario

HISTORIA DE USUARIO	
Número: 12	Nombre: Elaboración del reporte contable: Libro Diario
Usuario: Administrador, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: <ul style="list-style-type: none"> - Generar el reporte contable Libro Diario a la fecha actual. 	
Observación:	

Tabla 3.13: Historia de Usuario XIII: Elaboración del reporte Mayor General

HISTORIA DE USUARIO	
Número: 13	Nombre: Elaboración del reporte contable: Mayor General
Usuario: Administrador, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: - Generar el reporte contable Libro Mayor de manera automática.	
Observación:	

Tabla 3.14: Historia de Usuario XIV: Elaboración Balance de Comprobación

HISTORIA DE USUARIO	
Número: 14	Nombre: Elaboración del reporte contable: Balance de Comprobación
Usuario: Administrador, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: - Generar el reporte contable Balance de Comprobación de manera automática.	
Observación:	

Tabla 3.15: Historia de Usuario XV: Elaboración del Estado de Resultados

HISTORIA DE USUARIO	
Número: 15	Nombre: Elaboración del Estado de Resultados
Usuario: Administrador, Contador	Riesgo en desarrollo: Alto
Prioridad en negocio: Baja	Puntos estimados: 0.2
Descripción: - Generar reporte contable Estado de Resultados.	
Observación:	

3.5 Requisitos No Funcionales

Hacen relación a las características del sistema aplicando una visión generalizada del sistema más que a rasgos particulares del mismo.

Estos requerimientos complementan a los requisitos funcionales (los requisitos funcionales definen qué debe hacer un sistema, y los requisitos no

funcionales definen cómo debe ser el sistema).

En esta sección se definirán los requisitos no funcionales que tendrá este sistema.

Se han considerado los siguientes requisitos no funcionales:

- Seguridad: Se garantizará que los usuarios que se conecten al sistema, accedan únicamente a determinados servicios. Esto dependerá del nivel de privilegios (permisos) que le hayan sido asignados por el superadministrador del sistema.
- Disponibilidad: El sistema estará disponible a través de Internet, las 24 horas del día.
- Integridad: El sistema mantendrá tanto la integridad lógica (hace referencia al control de la relación entre las reglas de negocio y la base de datos), como la integridad de datos (control mediante validaciones).

3.6 Definición de Prioridades

Luego de analizar las historias de usuario y los requisitos involucrados en cada una de ellas se define el siguiente listado de prioridades.

Tabla 3.16: Análisis Historias de Usuario – Prioridad Alta

Prioridad Alta
1. Administración de Usuarios
2. Ingreso y utilización del sistema
3. Publicación de informes técnicos a través de la web.
6. Creación del formato de Orden de Trabajo
7. Creación del formato de Cotización

Tabla 3.17: Análisis Historias de Usuario – Prioridad Media

Prioridad Media
4. Control de la entrada de productos en cualquier bodega de la empresa
5. Control de la salida de productos en cualquier bodega de la empresa y generación del formato de Requisiciones de Material

Tabla 3.18: Análisis Historias de Usuario – Prioridad Baja

Prioridad Baja
8. Elaboración de Comprobantes de Ingreso
9. Elaboración de Comprobantes de Egreso
10. Elaboración de Facturas
11. Elaboración del Plan de Cuentas
12. Elaboración del reporte contable: Libro Diario
13. Elaboración del reporte contable: Mayor General
14. Elaboración del reporte contable: Balance de Comprobación
15. Elaboración del Estado de Resultados

3.7 Definición de Iteraciones

Tabla 3.19: Estimación de Iteraciones

No. Historia	Módulos	Nombre de la Historia de Usuario	Iteración
1	Construcción de la Interfaz del Sistema y Gestión de perfiles de usuario.	Administración de Usuarios	1
2		Ingreso y utilización del sistema	2
3	Repositorio digital de informes técnicos	Publicación de informes laborales a través de la web.	3,4,5
4	Gestión de Inventario	Control de la entrada de productos en cualquier bodega de la empresa	6,7
		Control de la salida de productos en cualquier bodega de la empresa y generación del formato de Requisiciones de Material	8,9
5		Creación del formato de Orden de Trabajo	10,11
7		Creación del formato de Cotización	12,13
8	Gestión Contable	Elaboración de Comprobantes de Ingreso	14
9		Elaboración de Comprobantes de Egreso	14
10		Elaboración de Facturas	15
11		Elaboración del Plan de Cuentas	15
12		Elaboración del Libro Diario	16
13		Elaboración del Mayor General	16
14		Elaboración del Balance de Comprobación	16
15		Elaboración del Estado de Resultados	16

CAPÍTULO 4: DISEÑO DEL PROYECTO

Como paso previo al diseño de la proyecto se realizó el análisis de requerimientos –como se describió en el Capítulo 3- mediante el levantamiento de requisitos, utilizando técnicas como reuniones de trabajo; de manera que se pueda entender las necesidades y requerimientos que tiene el usuario final, respecto al software que desea.

Se obtienen cada uno de los requerimientos, tomando en cuenta que la metodología X.P, plantea la utilización de un formato de Historias de Usuario para especificar los requerimientos en la situación actual.

En este capítulo, se diseñan Diagramas de Casos de Uso, Diagrama de clases, Diagrama Entidad – Relación, Diagrama de Componentes y Diagrama de Despliegue; de manera que se represente o visualice de forma gráfica cada uno de los requerimientos postulados en las Historias de Usuario. Adicionalmente se diseñó un Diagrama de flujo de trabajo para representar de manera más clara la parte del sistema referente a los procesos internos que se llevan a cabo en el área técnica, para la generación de Órdenes de Trabajo y Cotizaciones.

Además, en este capítulo, se define la arquitectura que tendrá el sistema.

4.1 Arquitectura y Diseño de Aplicaciones Java EE

Dividir una aplicación en partes que se ejecutan simultáneamente en computadores distintos, es la característica principal de la computación

distribuida.

- Se denomina *nivel* o “tier” a cada una de estas partes; y cada nivel brinda servicios a otros niveles.
- Cada nivel puede dividirse en *capas* o “layers”.

4.1.1 Qué es JEE:

Plataforma para desarrollar aplicaciones empresariales; basadas en componentes, centradas en el servidor, habilitadas para la web y distribuidas en n-capas.

4.1.2 Componentes Java EE

Un componente JEE es una unidad de software funcional que es ensamblada en una aplicación Java EE con clases y archivos relacionados que permiten la comunicación con otros componentes.

Debido a que Java EE utiliza un modelo de aplicaciones distribuidas multicapa, la lógica de la aplicación es dividida en componentes de acuerdo a la función.

- Componentes Capa Cliente: Clientes web o aplicaciones cliente. Corren sobre la máquina cliente.
- Componentes Capa Web: Servlets o páginas dinámicas creadas utilizando tecnología JSP o JSF.

- Componentes Capa Negocio: Componentes más importantes dentro de una aplicación JEE que representan la lógica que soluciona las necesidades de un negocio.
- Componentes EIS: Incluye sistemas de infraestructura empresarial como ERP, sistemas de base de datos, entre otros; que pueden ser accedidos por la aplicación para recuperar información.

4.1.3 Contenedores Java EE

Proveen un entorno de ejecución para componentes Java. Brindan servicios de seguridad, administración del ciclo de vida, transacciones, persistencia de datos, comunicación en la red, etc. Hay 2 tipos de contenedores:

- Contenedores Web: Almacenan componentes de presentación (capa web), como JSP, Servlets.
- Contenedores EJB: Administran la ejecución de componentes EJB.

4.1.4 Patrón Arquitectónico MVC

Los patrones Arquitectónicos proporcionan un esquema de la estructura básica del sistema. El principal ejemplo de este tipo de patrones es MVC (Modelo, Vista, Controlador), divide a la aplicación en 3 componentes principales:

- La vista, muestra el modelo en un formato entendible para el usuario, de manera que éste pueda interactuar con el sistema.
- El modelo, administra la data y aplica la lógica del negocio.
- El controlador, es un mediador que intercambia información (datos, instrucciones) entre la vista y el controlador.

MVC, consiste básicamente en realizar una serie de pasos cuando se construye el sistema, siendo estos:

- Realizar un mapeo objeto-relacional (con alguna herramienta ORM como por ejemplo Hibernate), de manera que a partir de una base de datos se creen las entidades (clases Java) que representan las tablas que se utilizarán en la aplicación.
- Codificar la lógica de negocio, es decir, crear los métodos/funciones para procesar las solicitudes del usuario, y hacer que las entidades persistan en la base de datos.
- Diseñar las vistas que permitan al usuario interactuar con la aplicación.

4.1.5 Capas en una Aplicación Java EE

“La expresión n-layers suele mezclarse con n-Tier. Las dos son arquitecturas en capas, pero n-Tier hace referencia a la parte física de la distribución. Es decir, los equipos en donde se encuentran las aplicaciones. Un ejemplo clásico de la arquitectura de 3 capas (3-Tier) es aquella que está distribuida en tres localizaciones específicas: la máquina del cliente, la máquina del servidor de negocio, y la máquina de la Base de Datos. De igual manera, la arquitectura de n-layers hace referencia a una división, pero esta vez a una

división lógica de la aplicación. Es decir que las 3 capas de este ejemplo podrían estar en una misma máquina, o en varias.”¹⁹ Garzón (2011)

Como se puede ver en la figura 4.1, en una arquitectura multicapas, algunas capas se subdividen en capas más especializadas, sin embargo, este tipo de arquitectura, sigue siendo una extensión de la arquitectura MVC.

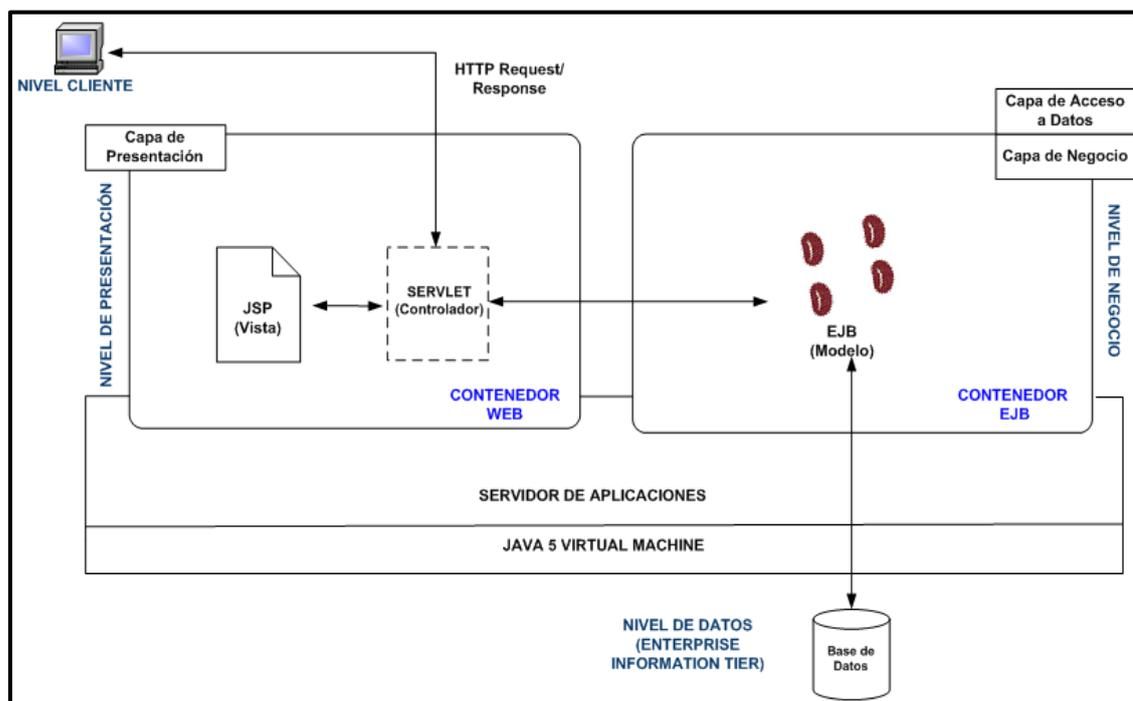


Figura 4.1: Arquitectura aplicación JEE²⁰

4.2 Arquitectura del Sistema Web

El presente sistema se diseñó como una aplicación Java EE, por tanto, tiene una arquitectura multi-capas, que es una extensión de la arquitectura MVC (Modelo Vista Controlador). Para mejor entendimiento, ver la Figura 4.2.

¹⁹ Adaptado de: "Un acercamiento a hibernate espacial.", por Garzón A. 2011. Recuperado de http://www.icde.org.co/web/ide_gig/blogs/-/blogs/un-acercamiento-a-hibernate-spatial.

²⁰ Figura. Arquitectura aplicación JEE. Adaptado de "Arquitectura y diseño de aplicaciones JAVA EE", por Gavidia C. Recuperado de <http://www.slideshare.net/cptanalatriste/arquitectura-y-diseo-de-aplicaciones-java-ee>

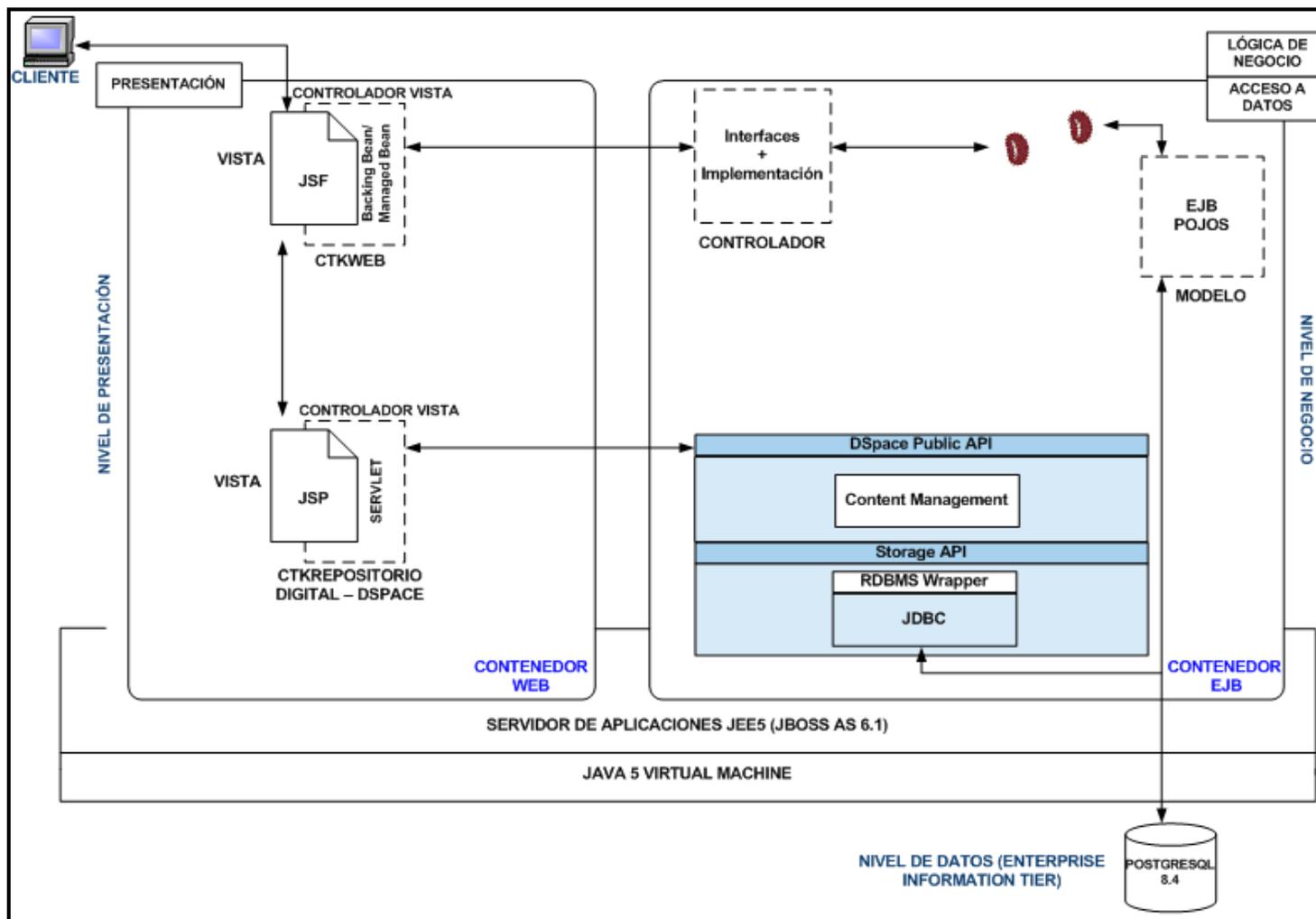
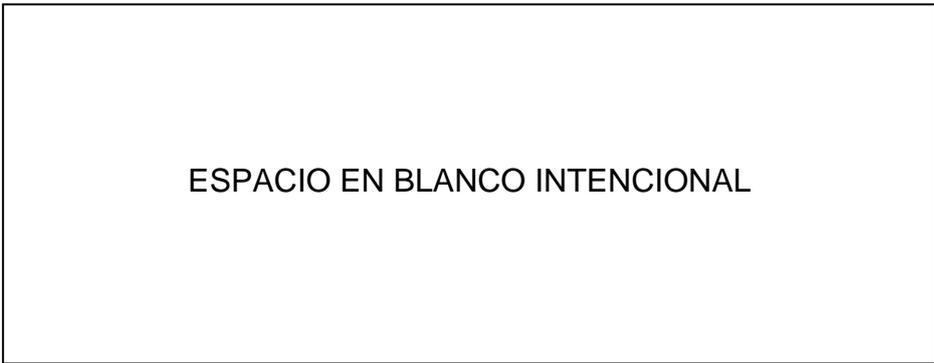


Figura 4.2: Arquitectura del Sistema Web CTKSYS

Dentro de varios conceptos sobre aplicaciones distribuidas, que muchas veces resulta complicado entender por completo; se ha tratado de ver al sistema de la forma más simple posible, por lo que se ha dividido el gran bloque de la aplicación en partes más pequeñas (donde los componentes están diseminados de diferentes formas y niveles) que se comuniquen entre sí para darle a la aplicación mayor flexibilidad y sobre todo escalabilidad.

Al momento de construir la aplicación se ha usado la arquitectura n-capas, misma que está basada en el patrón arquitectónico MVC y que es una especialización de la arquitectura 3 capas.

En la primera capa, se encuentra el aspecto gráfico de la aplicación (vista y controlador de la vista), en la segunda capa se encuentra la lógica de negocio (interfaces, clases implementadoras de dichas interfaces, etc) y en la tercera se encuentran los datos almacenados de nuestra aplicación; para hacer una descripción muy general del tema.



ESPACIO EN BLANCO INTENCIONAL

4.3 Interfaces

4.3.1 Página Principal

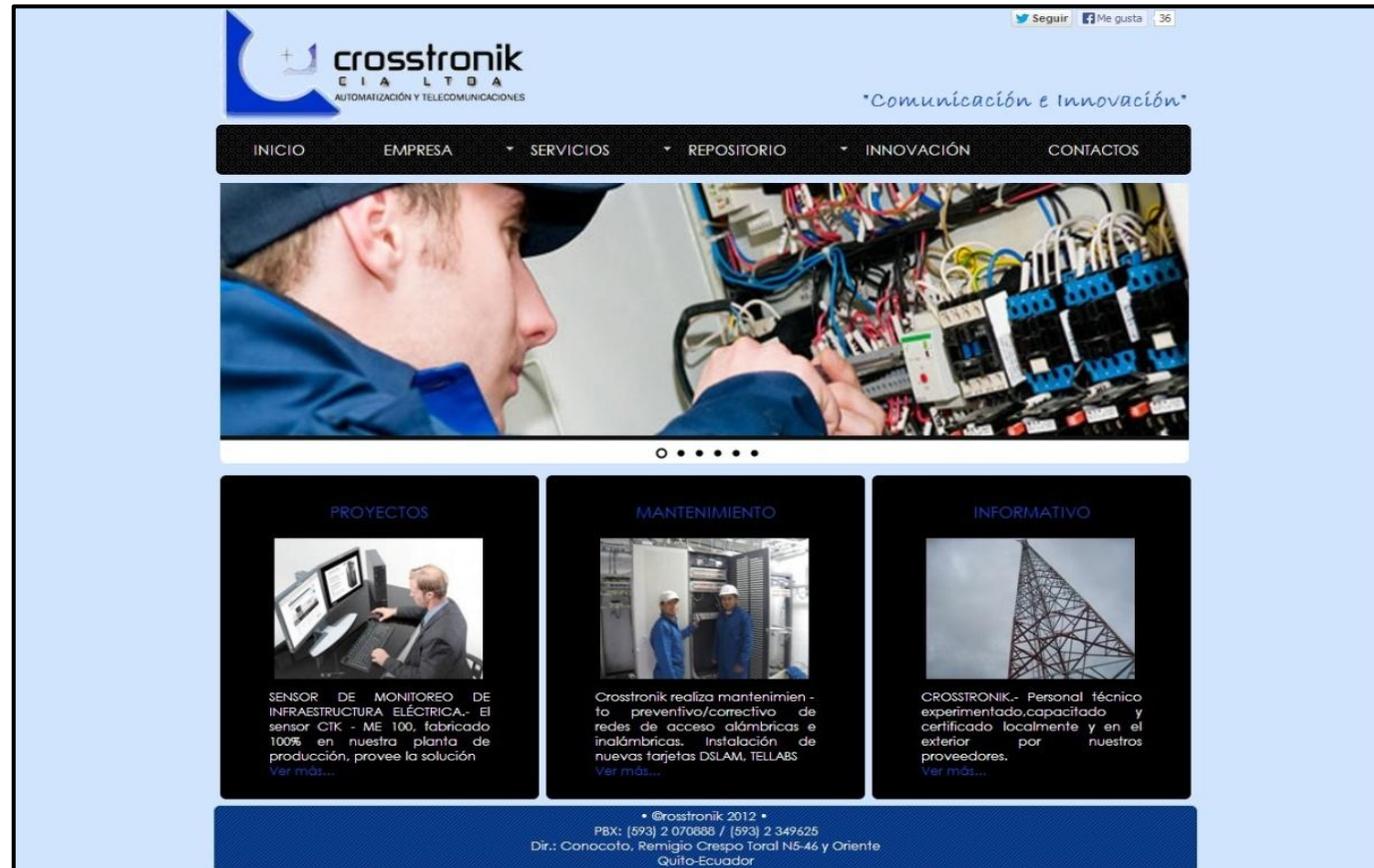


Figura 4.3: Pantalla página de inicio CTKSYS

4.3.2 Login

Home

CTK Login

CTK Login

Usuario:

Password:

Login

crosstronik
CIA LTDA
AUTOMATIZACIÓN Y TELECOMUNICACIONES

Figura 4.4: Pantalla de Login

4.4 Portal Repositorio digital de Informes técnicos

The screenshot shows the home page of the Crosstronik digital repository. At the top left is the logo for Crosstronik E.I.A. LTDA, with the tagline 'AUTOMATIZACIÓN Y TELECOMUNICACIONES'. Below the logo, the user is logged in as 'dcruz' with a 'Salir' link. The main content area is titled 'Repositorio CTK' and 'Comunidades y colecciones'. It features a list of communities: 'Huawei' and 'TVCable', with 'TVCable' having two sub-items: 'INFORMES STO. DGO. ABRIL 2013' and 'INFORMES STO. DGO. MAYO 2013'. A 'Herramientas de Administrador' box contains a 'Crear una comunidad' button. The footer includes a 'W3C XHTML 1.0' logo and copyright information for DSpace (© 2002-2010 MIT y Hewlett-Packard) and the Crosstronik repository.

Registrado como dcruz ([Salir](#))

Servicios

- [Ir Repositorio CTK](#) usuarios autorizados
- [Editar perfil](#)
- [Administrar](#)
- [Ayuda](#)

Repositorio CTK

Comunidades y colecciones

A continuación se muestra un listado de todas las comunidades, subcomunidades y colecciones. Haga clic sobre un nombre para ver su página principal.

- [Huawei](#)
- [TVCable](#)
 - [INFORMES STO. DGO. ABRIL 2013](#)
 - [INFORMES STO. DGO. MAYO 2013](#)

Herramientas de Administrador

[Crear una comunidad](#)

W3C XHTML 1.0

Software DSpace Copyright © 2002-2010 MIT y Hewlett-Packard - Repositorio Crosstronik

Figura 4.5: Pantalla de Inicio Repositorio CTK

4.5 Interfaces Módulo de Inventario

4.5.1 Movimientos – Ingresos

MÓDULO-INVENTARIO Bienvenid(a) Diego Cruz

Datos Generales Detalles Producto Ubicación

Cabecera de Ingreso:

INGRESO Nro.
8

Información General

Fecha: 9/05/13

Observación:

Buscar Nro. Ingreso: Guardar Cancelar Generar Nuevo Mov. Ingreso

→ Next

Figura 4.6: Pantalla Movimientos de Inventario – Ingresos

4.5.2 Movimientos – Egresos

MÓDULO-INVENTARIO Bienvenid(a) Diego Cruz

Datos Generales Detalles de Requisición Ubicación

Cabecera de Requisición:

REQUISICIÓN Nro.
0000008

Información General

Fecha: 9/05/13

Observación:

Buscar Nro. Requisición: 00000000

Guardar Cancelar

Generar Nueva Requisición

→ Next

©rosstronik 2012

Figura 4.7: Pantalla Movimientos de Inventario – Egresos/Requisiciones

4.5.3 Administración Productos

MÓDULO - INVENTARIO Bienvenid(a) Diego Cruz

Administración Producto

Código CTK *: Producto *: Otro Código:

Marca: Modelo: Serial:

Parte No.: Estado: Costo:

Área: Categoría *: Unidad *:

Observación:

LISTADO - PRODUCTOS

1 2 3 4 >> >>> 10

#	Cód. CTK	Producto	Otro Cód.	Características							Área	Observaciones	Operaciones
				Categoría	Marca	Modelo	Serial	Num. Parte	Unidad	Estado			
1	CTK00001	Laptop	n/a	Equipo	HP	PROBOOK 4420S	X1573740	n/a	pzs	Bueno	Administrativa	n/a	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
2	CTK00002	Laptop	n/a	Equipo	Toshiba	SATELLITE PRO	YA571921Q	n/a	pzs	Bueno	Administrativa	Hacer mantenimiento de anti-virus	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
3	CTK00003	Monitor	n/a	Equipo	LG	C17LC-0	602DYGSZ786	n/a	pzs	Bueno	Administrativa	n/a	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
4	CTK00004	Laptop	n/a	Equipo	HP	PROBOOK 4430S	CNU2022NMN	n/a	pzs	Bueno	Técnica	Se adquirió con recurso económico propio de Edison, no con recursos de CTK	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>

©rosstronik 2012

Figura 4.8: Pantalla Administración de Productos

4.5.4 Administración Bodegas

Administración - Bodega

Ciudad*: Nombre Bodega*:

Dirección: Teléfono:

LISTADO - BODEGAS

Nombre	Dirección	Teléfono	Ubicación	Operaciones
Bodega CTK - UIO	Conocoto	2070888	UIO	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
Bodega CTK - STO. DGO.	Calle Cuenca y Quito, Edificio Armijos Oficina 305	2754006	STO. DGO.	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
Bodega ECO - UIO	Conocoto	2349625	UIO	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
Bodega ECO - STO. DGO.	Cooperativa Las Palmas, calle Bolivia 207 y Padre Germán Maya	0994186596	STO. DGO.	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>
Laboratorio CTK - UIO	Conocoto	2070888	UIO	<input type="button" value="Eliminar"/> <input type="button" value="Editar"/>

Num. Total de Bodegas: **5**

©rosstronik 2012

Figura 4.9: Pantalla Administración de Bodegas

4.5.5 Administración Órdenes de Trabajo

Orden de Trabajo

Orden de Trabajo

ORDEN DE TRABAJO Nro. **3**

Fecha: 9/05/13 Estado:

Tema:

CLIENTE

Nombre: E-mail: Teléfono:

OBSERVACIONES

Nro. Factura:

Comp. Ingreso:

Forma de Pago: Efectivo

ARCHIVOS RELACIONADOS

COTIZACIÓN:

ADM- LIQUIDACIÓN VIÁTICOS:

TEC-INFORME:

ADM-SOLICITUD COMPRA:

DETALLES

Guardar Volver

Generar PDF

©crosstronik 2012

Figura 4.10: Pantalla Administración de Órdenes de Trabajo

4.5.6 Administración Cotizaciones

Orden de Trabajo Cotización Visualizar O.T. Visualizar Cotización Bienvenid@ Diego Cruz

Datos Generales Detalle-Materiales Detalle-Mano de Obra Total Materiales y Mano de Obra Condiciones Comerciales

Cotización

 **COTIZACIÓN Nro. 1**
Fecha: 9/05/13
Por: Limpieza interna de un servidor

CLIENTE

Atención:	Samuel Rivas	Empresa:	SURATEL TVCABLE
Dirección:	Eloy Alfaro y de las Higueras	E-mail:	srivas@tvcable.com.ec
Teléfono:	6002400	Fax:	6002400

Guardar

→ Next

©rosstronik 2012

Figura 4.11: Pantalla Administración de Cotizaciones

4.6 Interfaces Módulo Contable

4.6.1 Administración Comprobantes de Ingreso, Egreso y Transacciones

MÓDULO - CONTABLE Bienvenid(a) Lizeth Cruz

Datos Generales Detalles de Comprobante Finalización

Cabecera de Comprobante

COMPROBANTE DE INGRESO
00000005

Información General:

Recibido de*: TIPO: COMPROBANTE DE INGRESO

Efectivo: USD*:

Cheque Nº: Banco:

Por concepto*:

Anticipo: Relacionar con O. de Trabajo:

COMPROBANTE DE INGRESO
COMPROBANTE DE EGRESO
REGISTRO DE TRANSACCIÓN

Buscar
Nro. Comprobante: 00000000

Guardar Cancelar

Nuevo

→ Next

©rosstronik 2012

Figura 4.12: Pantalla Administración Comprobantes

4.6.2 Administración Facturas

MÓDULO - CONTABLE Bienvenid(a) Diego Cruz

Datos Generales Detalles de Factura Finalización

Cabecera de Factura

FACTURA
00000002

Información General:

CLIENTE*:

FECHA: 9/05/13

R.U.C./I:

TELÉFONO:

DIRECCIÓN:

Relacionar con una Orden de Trabajo:

Buscar Nro. Factura: 000000000

©rosstronik 2012

Figura 4.13: Pantalla Administración de Facturas

4.6.3 Reportes Contables

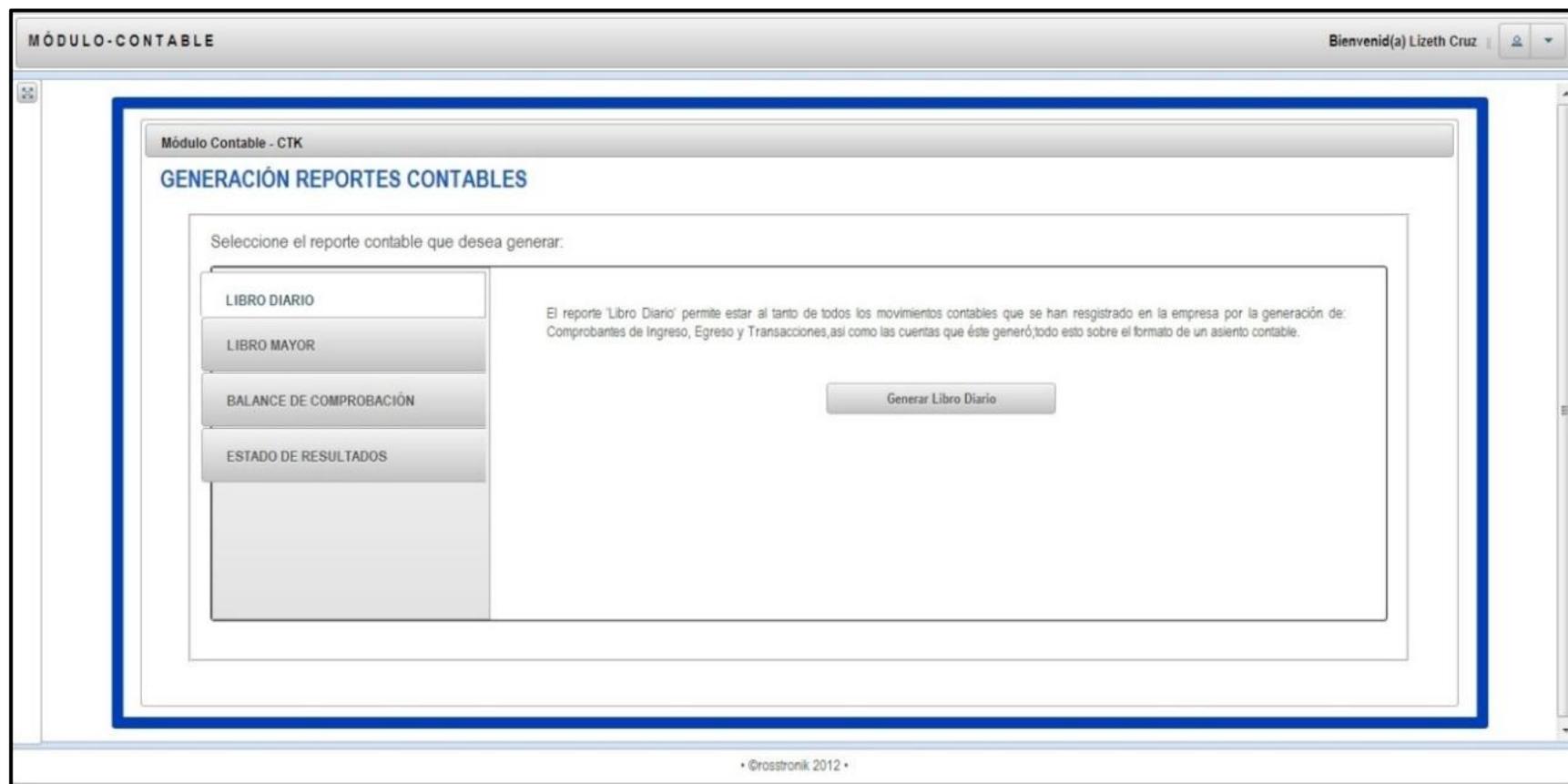


Figura 4.14: Pantalla Reportes Contables

4.7 Casos de Uso

4.7.1 Diagrama de Casos de Uso módulo I: Gestión de Usuarios

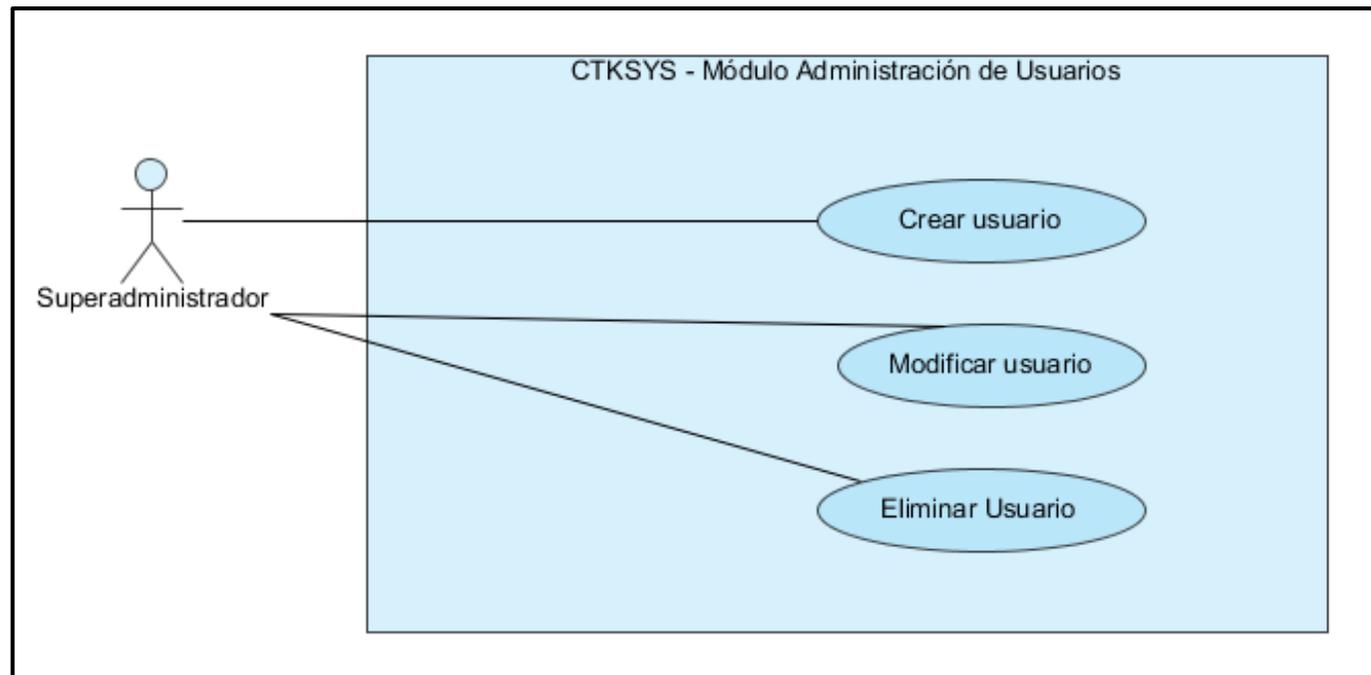


Figura 4.15: Diagrama C.U módulo Gestión de Usuarios

4.7.1.1 Descripción de C.U módulo I: Gestión de Usuarios

Nombre		Crear Usuario
Identificador		CU-01
Actor		Superadministrador
Descripción		El sistema deberá permitir al Superadministrador del sistema, crear un usuario para que haga uso del sistema.
Precondición		Usuario debe estar autenticado como Superadministrador y acceder al módulo de Gestión de Usuarios.
Postcondición		Usuario creado y habilitado dentro del sistema. Dependiendo del tipo de cargo asignado, el usuario podrá hacer uso de los módulos/secciones existentes en el sistema.
Paso	Acción	
	Actor	Sistema
1	Ingresar la siguiente información (Los campos requeridos están marcados con un *): <ul style="list-style-type: none"> • Nombre*: Nombre completo del usuario. • Nombre Usuario*: Nombre identificativo del usuario. •Cargo*: <Seleccionar entre: Administrador, Técnico, Bodeguero, Contador>. •Correo electrónico. •Contraseña* •Habilitado. <checkbox> Campo utilizado para dar de alta o baja a un usuario. 	Verifica que los campos ingresados sean correctos.
2	Da click en el botón Guardar	Informa el resultado de la acción realizada y muestra en el Listado – Usuarios los datos del usuario ingresado.
	2a	Si los campos requeridos no han sido ingresados el sistema deberá mostrar un mensaje de error.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Modificar Usuario	
Identificador	CU-02	
Actor	Superadministrador	
Descripción	El sistema deberá permitir al Superadministrador del sistema, modificar, un usuario existente dentro del sistema. Según se describe en el siguiente caso de uso:	
Precondición	Usuario debe estar autenticado como Superadministrador y acceder al módulo de Gestión de Usuarios.	
Postcondición	Usuario modificado listo para hacer uso del sistema.	
Paso	Acción	
	Actor	Sistema
1	Selecciona el registro perteneciente al usuario que va a modificar del Listado – Usuarios y da click en el botón que tiene un lápiz como ícono.	Muestra un cuadro de diálogo, donde solicita la confirmación de la modificación.
2	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	2a	Si el Superadministrador confirma la modificación de datos, se muestra/carga en pantalla los datos correspondientes al usuario seleccionado.
	2b	Si el Superadministrador no confirma la modificación de datos, el sistema no realiza ninguna acción.
3	Da click en el botón Guardar.	Verifica que los datos hayan sido ingresados correctamente y muestra un mensaje con el resultado de la acción realizada. Aparecerá en el Listado – Usuarios los datos del usuario con los cambios.
	3a	Si los campos requeridos no han sido ingresados el sistema deberá mostrar un mensaje de error.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Eliminar Usuario	
Identificador	CU-03	
Actor	Superadministrador	
Descripción	El sistema deberá permitir al Superadministrador del sistema, eliminar, un usuario existente dentro del sistema. Según se describe en el siguiente caso de uso:	
Precondición	Usuario debe estar autenticado como Superadministrador y acceder al módulo de Gestión de Usuarios.	
Postcondición	Usuario eliminado del sistema.	
Paso	Acción	
	Actor	Sistema
1	Selecciona el registro perteneciente al usuario que va a eliminar del Listado – Usuarios y dar click en el botón que tiene un tachó de basura como ícono.	Muestra un cuadro de diálogo, donde solicita la confirmación de la eliminación.
2	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	2a	Si el Superadministrador confirma la eliminación de los datos, se muestra un mensaje con el resultado de la acción realizada y el registro desaparecerá en el Listado – Usuarios.
	2b	Si el Superadministrador no confirma la eliminación de datos, el sistema no realiza ninguna acción.

ESPACIO EN BLANCO INTENCIONAL

4.7.2 Diagrama de C.U módulo II: Gestión de Informes Técnicos

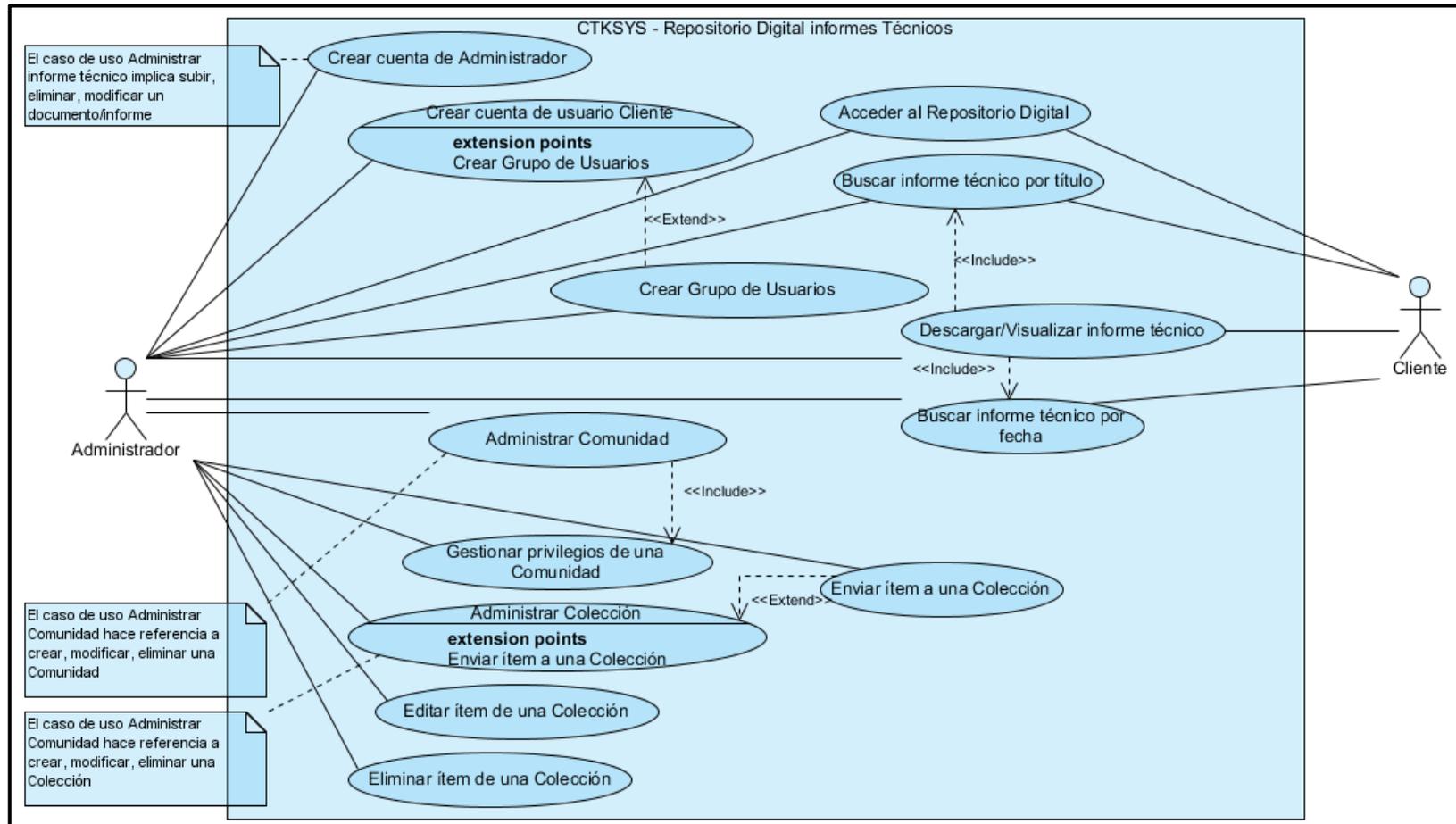


Figura 4.16: Diagrama C.U módulo Gestión de Informes técnicos

4.7.2.1 Descripción de C.U módulo II: Gestión de Informes Técnicos

Nombre		Crear la cuenta de Administrador del repositorio Digital
Identificador		CU-D01
Actor		
Descripción		El sistema deberá permitir la creación de la cuenta de Administrador del repositorio Digital
Precondición		
Postcondición		Cuenta de Administrador creada y lista para ser utilizada.
Paso	Acción	
	Actor	Sistema
1	Abrir el editor de comandos o CMD.	Se muestra la ventana comandos.
2	En la ventana de comandos ubicarse en la carpeta de instalación de DSpace, y digitando > cd C:\DSpace\bin. Ejecutar la siguiente orden: C:\DSpace\bin>DSpace dsrun org.DSpace.administer.CreateAdministrator. Ir ingresando los datos que se van solicitando. La información referente al mail, puede llenarse con un nombre de usuario cualquiera o la dirección de correo electrónico del administrador. Es un campo único (no podrá repetirse cuando se creen nuevos usuarios).	El sistema solicita la confirmación de los datos ingresados para poder concluir con el proceso de creación de la cuenta de Administrador inicial.
3	Digitar la letra y si los datos ingresados son los correctos, o la letra n si existiese algún dato erróneo.	
	3a	Si se confirma que la información que se ingresa, es la correcta, se informa que la cuenta de Administrador ha sido creada correctamente.
	3b	Si no se confirma que la información ingresada es correcta, se solicita volver a ingresar los datos.

ESPACIO EN BLANCO INTENCIONAL

Nombre		Crear grupo de Usuarios
Identificador		CU-D02
Actor		Administrador
Descripción		El sistema deberá permitir al usuario Administrador la creación de un grupo de usuarios.
Precondición		Tener cuenta de Administrador.
Postcondición		Usuario forma parte de un grupo de usuarios.
Paso	Acción	
	Actor	Sistema
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	En el menú izquierdo seleccionar la opción Administrar.	Se muestra la pantalla de Administración.
3	En el menú izquierdo seleccionar la opción Grupos.	Se muestra el Editor de Grupo.
4	Ingresar la siguiente información: Nombre: nombre representativo del grupo de usuarios a crear. Dar click en el botón Seleccionar usuarios.	El sistema verifica los datos ingresados.
4 ^a	Si el Nombre de Grupo no ha sido ingresado, se asigna un nombre por defecto. Se muestra una pantalla emergente con el listado de todos los usuarios.	
5	Escoger el usuario del listado de usuarios, que será añadido al grupo que se acabó de crear, dando click en el botón Añadir.	El usuario/cliente aparecerá en el listado Miembros Epersona.
6	Para añadir un usuario a un Grupo ya existente: dar click en el botón Editar del listado de Grupos que se muestran.	Se muestra Editar grupo: nombre del Grupo ya existente.
7	Dar click en el botón Seleccionar usuarios	Se muestra una pantalla emergente con el listado de todos los usuarios existentes.
8	Escoger el usuario que se añadirá al grupo, dando click en el botón Añadir.	El usuario/cliente aparecerá en el listado Miembros Epersona..
9	Dar click en el botón Seleccionar usuarios	Se muestra una pantalla emergente con el listado de todos los usuarios existentes.
10	Para retirar un usuario de un Grupo: dar click en el botón Editar del listado de Grupos que se muestran.	Se muestra Editar grupo: nombre del Grupo.
11	Para retirar un usuario de un Grupo: dar click en el botón Editar del listado de Grupos que se muestran.	Se muestra Editar grupo: nombre del Grupo.
12	Escoger el usuario que se eliminará del grupo, dando click en el botón Borrar Selección.	El usuario/cliente desaparecerá del listado Miembros Epersona.
13	Para editar un Grupo ya existente: Escoger el grupo a modificar, dando click en el botón Editar del listado de Grupos ya creados.	Se muestra Editar grupo: nombre del Grupo ya existente.

14	Dependiendo de lo que se necesite, se puede modificar el Nombre del grupo, o los Miembros Epersona (usuarios) que pertenecen al mismo. Para la realización de esta última acción, se deben seguir los pasos ya descritos en los puntos: 7, 8, 9, 10.	Se muestra en pantalla los nuevos cambios.
15	Para eliminar un Grupo: Escoger el grupo a eliminar, dando click en el botón Borrar del listado de Grupos ya creados.	Se muestra la página Delete Group: nombre del Grupo. Con botones de Eliminar o Cancelar.
16	Dar click en el botón Eliminar.	El grupo eliminado desaparece del listado de Grupos.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Crear, modificar, eliminar una Comunidad.	
Identificador	CU-D03	
Actor	Administrador	
Descripción	El sistema deberá permitir al usuario Administrador crear, modificar, eliminar una Comunidad.	
Precondición	Tener una cuenta con privilegios de Administrador, y distintas cuentas de usuario creadas.	
Postcondición	Comunidad lista para generar Colecciones.	
Paso	Acción	Sistema
	Actor	
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	Acceder a la sección Crear Comunidad: Seleccionando la opción Administrar – Comunidades/Colecciones del menú izquierdo ó dando click en el botón Crear Comunidad ubicado en la parte superior derecha de la pantalla principal del Repositorio CTK.	El sistema muestra la sección Crear Comunidad.
3	Ingresar los datos principales, de la Comunidad a crear. Dar click en el botón Crear.	Se muestra la página de inicio de la Comunidad. Con un panel Herramientas de Administrador, con botones para Editar, Crear una Colección, Exportar metadata
4	Dar click en el botón Editar para modificar una Comunidad.	Se muestra la página Editar Comunidad. Con la información correspondientes a la Comunidad, cargada.
5	Hacer los cambios requeridos. Dar click en el botón Actualizar.	Se muestra la página de inicio de la Comunidad. Con los cambios realizados.
6	Dar click en el botón Editar para borrar una Comunidad.	Se muestra la página Editar Comunidad. Con la información correspondientes a la Comunidad cargada. Y un botón Borrar esta Comunidad.
7	Dar click en el botón Borrar esta Comunidad.	Se muestra la página de confirmación Borrar Comunidad.
8	Dar click en el botón Borrar.	La Comunidad desaparece del listado Comunidades y colecciones que se visualizan en la página principal del Repositorio CTK.

Nombre	Gestionar los privilegios de una Comunidad	
Identificador	CU-D04	
Actor	Administrador	
Descripción	El sistema deberá permitir al usuario Administrador gestionar los privilegios de una Comunidad.	
Precondición	Tener una cuenta con privilegios de Administrador, y haber creado una Comunidad.	
Postcondición	Restricción de accesos/visualización de una Comunidad para uno o un grupo específico de usuarios.	
Paso	Acción	Sistema
	Actor	
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	En el menú izquierdo seleccionar la opción Administrar.	Se muestra la pantalla de Administración.
3	En el menú izquierdo seleccionar la opción Autorización.	Se muestra la pantalla Administrar privilegios de autorización.
4	Dar click en el botón Gestionar los privilegios de una Comunidad.	Se muestra la pantalla Comunidades:
5	Seleccionar una Comunidad del listado de Comunidades y dar click en el botón Editar privilegios.	Se muestra la pantalla Privilegios para la Comunidad “Nombre de la Comunidad”.
6	Del listado de Grupos, eliminar el grupo Anonymous, que está establecido por defecto.	El grupo Anonymous desaparecerá del listado.
7	Dar click en el botón Añadir nuevos privilegios.	Se muestra la página Editar privilegios para Community número de la Comunidad:, y un listado de los grupos que se han creado en el sistema.
8	Dar click en el botón Borrar.	La Comunidad desaparece del listado Comunidades y colecciones que se visualizan en la página principal del Repositorio CTK.
9	En esta sección, es donde se escoge el grupo de usuarios y los privilegios que tendrán dentro de esta Comunidad.	Se muestra la página Privilegios para la Comunidad “Nombre de la Comunidad”. En el listado aparece el grupo que tendrá acceso a esta Comunidad y la Acción (READ).
10	Para editar un privilegio: Seleccionar el privilegio que se desea editar, dando click en el botón Editar del listado que se presenta en la pantalla Privilegios para la Comunidad	Se muestra en la página Privilegios para la Comunidad “Nombre de la Comunidad”, el listado de privilegios.
11	Seguir lo descrito en el paso 5.	Se muestra la página Privilegio
12	Para eliminar un privilegio: Dar click en el botón Eliminar del listado que se presenta en la pantalla Privilegios para la Comunidad	El privilegio desaparece del listado de privilegios.

Nombre	Crear, modificar, eliminar una Colección.	
Identificador	CU-D05	
Actor	Administrador	
Descripción	El sistema deberá permitir al usuario Administrador crear, modificar, eliminar una Colección.	
Precondición	Tener una cuenta con privilegios de Administrador, y haber creado una Comunidad.	
Postcondición	Colección administrada lista para que se puedan añadir Ítems.	
Paso	Acción	
	Actor	Sistema
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	Seleccionar la Comunidad donde se desea crear la Colección del listado Comunidades y colecciones.	Se muestra la página de inicio de la Comunidad seleccionada. Con un panel Herramientas de Administrador, con botones.
3	Dar click en el botón Crear una Colección disponible en el panel Herramientas de Administrador ubicado en el lado superior derecho de la página.	Se muestra el wizard (conjunto de pasos a seguir) –Describe la Colección-, con dos opciones seleccionadas por defecto.
4	Dar click en el botón Siguiente.	Describe la Colección, con los campos descriptivos de la Colección a ingresar.
5	Ingresar los datos principales, de la Colección a crear: • Nombre, • Descripción corta, • Texto introductorio, • Licencia. Dar click en el botón Siguiente.	Se muestra la página: Autorización para enviar Ítems.
6	Dar click en el botón Siguiente.	Se muestra la página Editar Colección.
7	En la sección Procedimientos de envío – Submitters dar click en el botón Borrar.	En la sección Procedimiento de envío, el botón Borrar desaparecerá de la vista.
8	Para editar una Colección. Seleccionar una Colección del listado de Comunidades y Colecciones. Dar click en el botón Editar, ubicado en el panel Herramientas de Administrador.	Se muestra la página Editar Colección. Con la información correspondiente a la Colección que se seleccionó, cargada.
9	Realizar los cambios necesarios, teniendo presente lo descrito en el punto 5.	Se muestra la página de inicio de la Comunidad y la Colección donde se visualizan los cambios realizados.
10	Para eliminar una Colección. Dar click en el botón Editar, ubicado en el panel Herramientas de Administrador ubicado en el lado superior derecho de la Página de inicio de la Colección	Se muestra la página de confirmación Borrar Colección: número de la Colección, con los botones Borrar y Cancelar.
11	Dar click en el botón Eliminar.	La Colección eliminada, desaparece del listado de Colecciones.

Nombre	Enviar un Ítem a una Colección.	
Identificador	CU-D06	
Actor	Administrador	
Descripción	El sistema deberá permitir al usuario Administrador enviar un Ítem a una Colección.	
Precondición	Tener una cuenta con privilegios de Administrador, y haber creado una Colección.	
Postcondición	Informe técnico subido a una Colección.	
Paso	Acción	Sistema
	Actor	Sistema
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	Seleccionar la Colección donde se desea subir el informe del listado Comunidades y colecciones.	Se muestra la Página de inicio de la Comunidad Y un botón central Enviar un Ítem en esta Colección.
3	Dar click en el botón Enviar un Ítem en esta Colección.	Se muestra el wizard (conjunto de pasos a seguir) –Describe la Colección-, con dos opciones seleccionadas por defecto.
4	Dar click en el botón Siguiente.	Se muestra un wizard (Paso 1: Envío: describa su Ítem) que irá guiando al Administrador, a lo largo del proceso de subida de un Ítem.
5	Marcar la casilla o casillas que correspondan al Ítem que se va enviar. Si ninguna opción aplica para el envío actual, no marcar ninguna (no son campos requeridos). Dar click en Siguiente	Se muestra el paso 2. Envío: describa el Ítem.
6	Ingresar los datos principales, del informe: • Título (es un campo requerido), • Tipo (No es un campo requerido). Dar click en el botón Siguiente.	Se muestra el paso 3. Envío: describa el Ítem.
7	Ingresar los datos descriptivos adicionales del informe: •Palabras clave (palabras apropiadas que serán utilizadas como patrón de búsqueda para este informe) • Resumen. • Descripción. Dar click en el botón Siguiente.	Se muestra el paso 4. Envío: Subir un fichero
8	Dar click en el botón Examinar y seleccionar el archivo perteneciente al informe que se va a subir. * El nombre con el que se guarde al archivo no debe llevar tildes ni ñ o Ñ. No se puede subir un archivo en blanco.	Se muestra el paso 4. Envío: El fichero se ha subido correctamente
9	En caso de no existir ningún problema con el archivo subido, dar click en el botón Siguiente, caso contrario dar click en los botones Haga click aquí si el fichero no es correcto o en el botón Haga click aquí si el fichero no es correcto; dependiendo del	Se muestra el paso 5. Verificar

	requerimiento.	
10	Revisar la información que ha sido ingresada. Si es la correcta, dar click en el botón Siguiente, caso contrario dar click en alguno de los diferentes botones, dependiendo de su requerimiento.	Se muestra el paso 6. Licencia
11	Dar click en el botón Acepto la licencia.	Se muestra la página de Envío: envío finalizado!
12	Eliminar el campo URI del informe: Ubicarse en la Página de inicio de la Colección, donde se hizo la subida del Ítem. Dar click en el botón Exportar metadata, para revisar el ID interno o el handle perteneciente al informe subido. Este ID/Handle, se utilizará para poder buscar el Ítem y poder modificar el campo. Esta acción se realizará desde la sección Ítems, como se detalla en CU-D07.	Se descarga automáticamente un archivo en Excel con la información descriptiva de todos los Ítems subidos dentro de la Colección seleccionada.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Editar o borrar un Ítem de una Colección.	
Identificador	CU-D07	
Actor	Administrador	
Descripción	El sistema deberá permitir al usuario Administrador editar o borrar un Ítem de una Colección.	
Precondición	Tener una cuenta con privilegios de Administrador.	
Postcondición	Informe técnico modificado o eliminado el Repositorio	
Paso	Acción	Sistema
	Actor	
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador.	Se muestra la pantalla principal del Repositorio CTK.
2	En el menú izquierdo seleccionar la opción Administrar.	Se muestra la pantalla de Administración.
3	En el menú izquierdo seleccionar la opción Ítems.	Se muestra la pantalla Editar o borrar Ítem, con un recuadro para buscar el Ítem a modificar/eliminar por su número de handle o por su ID interno.
4	Editar Ítem: para eliminar algún campo descriptivo de un informe, como en este caso, el campo URI, seguir los siguientes pasos: Verificar en el archivo de Excel que se generó cuando se dio click en el botón Exportar metadata, el dato referente al ID del fichero que se va a modificar. Dar click en el botón Buscar.	Se muestra la pantalla Editar Ítem.
5	Seleccionar del listado de metadata perteneciente a ese Ítem, el registro de identifier uri, dando click en el botón Borrar.	El campo eliminado desaparece del listado de datos descriptivos del Ítem/informe.
6	Para eliminar el Ítem. Verificar en el archivo de Excel que se generó cuando se dio click en el botón Exportar metadata, el dato referente al ID del fichero que se va a borrar. Dar click en el botón Buscar.	Se muestra la pantalla Editar Ítem
7	Dar click en el botón Eliminar.	Se muestra la pantalla de confirmación Borrar Ítem.
8	Dar click en el botón Borrar.	El Ítem desaparece del listado de Ítems existentes en la Colección.

Nombre	Descargar/Visualizar Informe Técnico	
Identificador	CU-D08	
Actor	Administrador, Cliente	
Descripción	El sistema deberá permitir al usuario Administrador, cliente descargar o visualizar un informe técnico subido a una Colección.	
Precondición	Haberse logueado en dentro del sistema.	
Postcondición	Descarga/Visualización de un informe técnico por el usuario cliente.	
Paso	Acción	
	Actor	Sistema
1	Acceder a la página principal de la empresa. Sección Repositorio – Informes. Loguearse como Administrador o usuario Cliente.	Se muestra la pantalla principal del Repositorio CTK.
2	El usuario cliente debe visualizar únicamente la Comunidad para la cual se le concedieron los privilegios. Para descargar o visualizar un informe técnico seguir los siguientes pasos: - Dar click en el nombre de alguna Colección.	Se muestra la Página de inicio de la Colección, con un recuadro central para búsquedas. Y un panel al lado derecho, de acceso rápido hacia los envíos recientes.
3	Existen varias opciones para poder visualizar/descargar un informe publicado para una Colección: Opción I: Dar click en el título de un Ítem del Listado de Envíos recientes (aparece en el lado derecho de la Página de inicio de la Colección).	Se muestra una descripción completa del informe que va a descargar o visualizar. Y un recuadro centrado en la parte inferior de la página con opciones para Visualizar/abrir el documento.
4	Dar click en el link Visualizar/Abrir o en el link referente al nombre del fichero.	Dependiendo del explorador que se esté utilizando, la descarga del fichero se hará de distintas formas: mediante el uso de una ventana emergente para la descarga, o la descarga directa del fichero sin pedir confirmación.
5	Opción II: Utilizar el recuadro de Búsquedas. – Buscar por: Ingresar un criterio de búsqueda en la caja de texto y dar click en el botón Ir.	Se muestra una descripción completa del informe que va a descargar o visualizar. Y un recuadro centrado en la parte inferior de la página con opciones para Visualizar/abrir el documento.
6	Para descargar el informe realizar lo descrito en el paso 4.	
7	Para buscar el fichero, utilizar los comodines para ordenar los ficheros por Título, en Orden ASC/DSC y dar click en el botón Update. Seleccionar el Ítem.	Se muestra una descripción completa del informe que va a descargar o visualizar. Y un recuadro centrado en la parte inferior de la página con opciones para Visualizar/abrir el documento.
8	Para descargar el informe realizar lo descrito en el paso 4.	

9	Opción IV: Utilizar el recuadro de Búsquedas. – Listar por: click en el botón Fecha Publicación para listar los ficheros ordenados por fecha.	Se muestra la página Buscar por Fecha de publicación. Con el listado completo de los ficheros pertenecientes a esa Colección.
10	Para buscar el fichero, utilizar los comodines para ordenar los ficheros por Fecha Publicación, en Orden ASC/DSC y dar click en el botón Update. Seleccionar el Ítem.	Se muestra una descripción completa del informe que va a descargar o visualizar. Y un recuadro centrado en la parte inferior de la página con opciones para Visualizar/abrir el documento.
11	Para descargar el informe realizar lo descrito en el paso 4.	

ESPACIO EN BLANCO INTENCIONAL

4.7.3 Diagrama de C.U módulo III: Gestión de Inventario

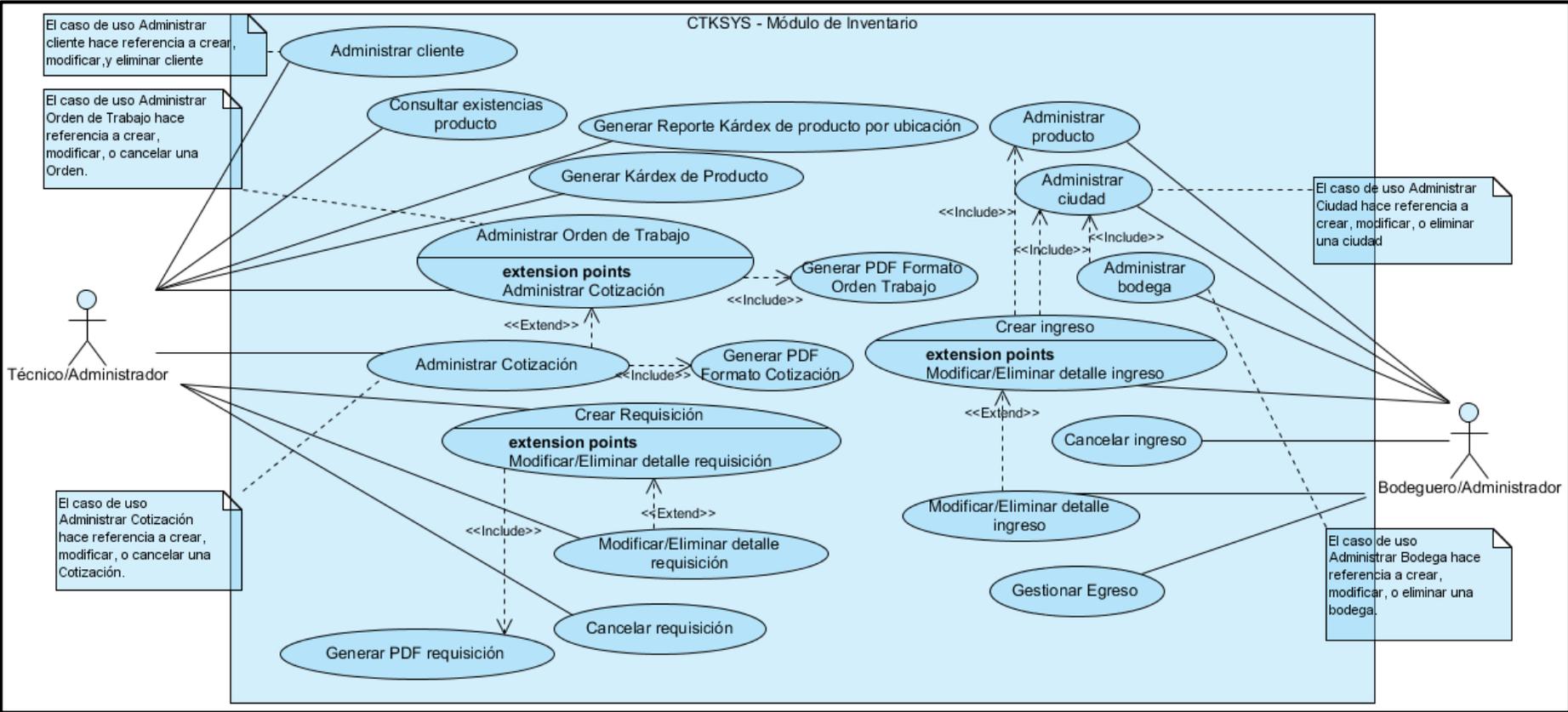


Figura 4.17: Diagrama C.U módulo Gestión de Inventario

4.7.3.1 Descripción de C.U módulo III: Gestión de Inventario

Nombre		Administrar Producto	
Identificador		CU-I01	
Actor		Administrador, Bodeguero	
Descripción		El sistema deberá permitir al usuario Bodeguero o Administrador crear, modificar, eliminar, buscar por código, buscar por nombre, buscar por área a un producto. Según se describe en el siguiente caso de uso:	
Precondición		Usuario debe estar autenticado bajo perfil de Bodeguero o Administrador acceder al módulo de Inventario, sección Administrar Producto.	
Postcondición		Producto administrado y listo para ser utilizado.	
Paso	Acción		
	Actor	Sistema	
1	Da click en el botón Ver Cod. CTK, que proporciona la información del último código que ha sido ingresado. Para verificar el código del siguiente producto a ingresar.	Despliega un cuadro de diálogo con la información del último código CTK que ha sido ingresado.	
2	Ingresar la siguiente información en el formulario: (Los campos requeridos están marcados con un *) <ul style="list-style-type: none"> • Código CTK*: <Acorde al último código CTK ingresado en el sistema>. • Producto*: Nombre identificativo del producto. • Otro Código: Posible segundo código identificador de producto. No es un campo requerido. • Marca. • Modelo. • Serial. • Parte No.: Número de parte. • Estado:<Seleccionar entre: Bueno, Dañado, Defectuoso>. • Costo. • Área: <Seleccionar entre: Administrativa, Técnica, General>. • Categoría*:<Seleccionar entre: Herramienta, Material, Equipo, Otros>. • Unidad*: <Seleccionar la unidad> • Observación. 	Verifica que los datos hayan sido ingresados correctamente.	
3	Da click en el botón Guardar	Informa el resultado de la acción realizada y muestra el producto ingresado con sus campos identificativos, en el Listado – Productos para su utilización.	
	3a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.	
4	Requiere modificar un producto: selecciona el producto del Listado – Productos, y da click en el botón que tiene un lápiz como ícono.	El sistema muestra un cuadro de diálogo, donde solicita la confirmación de la modificación.	
5	Selecciona del cuadro de diálogo emergente, la acción a realizar.	.	

	5a	Si el usuario confirma, se muestra/carga en pantalla los datos correspondientes al producto seleccionado.
	5b	Si el usuario no confirma, el sistema no realiza ninguna acción.
6		<p>Dar click en el botón Modificar.</p> <p>El sistema verifica que los datos hayan sido ingresados correctamente, muestra un mensaje con el resultado de la acción realizada y el registro aparecerá en el Listado – Productos con los cambios.</p>
	6a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
7		<p>Requiere eliminar un producto: selecciona el producto del Listado – Productos, y da click en el botón que tiene un tacho de basura como ícono.</p> <p>El sistema muestra un cuadro de diálogo, donde solicita la confirmación de la eliminación.</p>
8		<p>Seleccionar del cuadro de diálogo emergente, la acción a realizar.</p>
	8a	Si el usuario confirma, se muestra un mensaje con el resultado de la acción realizada y el registro desaparecerá en el Listado – Productos.
	8b	Si el usuario no confirma el sistema no realiza ninguna acción.
9		<p>Requiere buscar por código: Ingresar el código CTK y da click en el botón Buscar por Código.</p> <p>El sistema muestra un mensaje con el resultado de la búsqueda generada, y el registro del producto aparecerá en el Listado – Productos. Caso contrario el Listado – Productos no tendrá ningún dato.</p>
10		<p>Requiere buscar por nombre: Ingresar el nombre de un producto o una parte del mismo, y da click en el botón Buscar por Nombre.</p> <p>El sistema muestra un mensaje con el resultado de la búsqueda generada, y el registro del producto aparecerá en el Listado – Productos. Caso contrario el Listado – Productos no tendrá ningún dato.</p>
11		<p>Requiere buscar por área: Selecciona un área del combo Área, y da click en el botón Buscar por Área.</p> <p>El sistema muestra un mensaje con el resultado de la búsqueda generada, y el registro del producto aparecerá en el Listado – Productos. Caso contrario el Listado – Productos no tendrá ningún dato.</p>
12		<p>Requiere ver el listado de todos los productos que han sido ingresados: Da click en el botón Listar Productos.</p> <p>El sistema muestra un mensaje con el resultado de la acción generada, y se llenará el Listado – Productos con todos los productos.</p>
13		<p>Da click en el botón Limpiar Formulario, para limpiar los campos y el Listado – Productos.</p> <p>El sistema muestra un mensaje con el resultado de la acción generada, se limpian todos los campos y el Listado – Productos.</p>

Nombre	Administrar Bodegas	
Identificador	CU-I02	
Actor	Administrador, Bodeguero	
Descripción	El sistema deberá permitir al usuario Bodeguero o Administrador crear, modificar o eliminar las distintas bodegas de la empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Bodeguero o Administrador, acceder al módulo de Inventario, sección Administración Ubicación – Administrar Bodega.	
Postcondición	Listado de Bodegas listo para ser utilizado.	
Paso	Acción	
	Actor	Sistema
1	Ingresar la siguiente información en el formulario: (Los campos mandatorios están marcados con un *) •Ciudad* : <<Combo>> Seleccionar una ciudad del listado. •Nombre Bodega: Nombre identificativo de la bodega a crear. •Dirección. •Teléfono. Dar click en el botón Guardar.	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada y muestra en el Listado – Ciudades los datos de la ciudad ingresado.
	1a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	1a	Si la información ingresada es correcta, el sistema informa el resultado de la acción realizada y muestra el registro en el Listado – Bodegas.
3	Requiere modificar una bodega: Selecciona el registro perteneciente a la bodega que va a modificar del Listado – Bodegas y da click en el botón que tiene un lápiz como ícono.	Muestra un cuadro de diálogo, donde solicita la confirmación de la modificación.
4	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	4a	Si el usuario confirma, se muestra/carga en pantalla los datos correspondientes a la bodega seleccionada.
	4b	Si el usuario no confirma, el sistema no realiza ninguna acción.
5	Da click en el botón Guardar.	Verifica que los datos hayan sido ingresados correctamente. Muestra un mensaje con el resultado de la acción realizada, y se actualiza la información.
6	Requiere eliminar una bodega: Selecciona el registro perteneciente a la bodega que va a eliminar del Listado – Bodegas y da click en el botón con el tacho de basura como ícono.	Muestra un cuadro de diálogo, donde solicita la confirmación de la eliminación.
7	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	7a	Si el usuario confirma la eliminación de los datos, se muestra un mensaje con el resultado de la acción realizada y el registro desaparecerá en el Listado – Bodegas.
	7b	Si el usuario no confirma, el sistema no realiza ninguna acción.

Nombre		Administrar Ingreso
Identificador		CU-I03
Actor		Administrador, Bodeguero
Descripción		El sistema deberá permitir al usuario Bodeguero o Administrador crear, modificar o cancelar ingresos dentro de las distintas bodegas de la empresa.
Precondición		Usuario debe estar autenticado bajo perfil de Bodeguero o Administrador, acceder al módulo de Inventario, sección Administración Inventario.
Postcondición		Ingresos administrados e inventario actualizado.
Paso	Acción	
	Actor	Sistema
1	Seleccionar de la opción: Ingresos, el submenú Crear Ingresos	Carga el formulario de Cabecera de Ingreso.
2	Ingresar la siguiente información en el formulario: (No existen campos mandatorios) • Fecha. •Observación. Dar click en el botón Guardar.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	2b	Si no se guarda la cabecera, el sistema no permitirá que se avance para guardar los detalles del ingreso.
3	Se procede a buscar el producto a ingresar; la búsqueda puede realizarse por: Código CTK o nombre.	Carga los datos del producto seleccionado.
4	Ingresar la siguiente información en el formulario de Detalle del ingreso: (Todos los campos son mandatorios) •Ubicación: Nombre de la ciudad donde se va a registrar el ingreso. •Bodega: Nombre de la bodega en la que se va a registrar el ingreso. •Cantidad a ingresar: cantidad del producto que se va a ingresar. Dar click en el botón Guardar Detalle.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle-Ingreso. Informa el resultado de la acción realizada.
	4a	Para agregar otro detalle al mismo ingreso, dar click en el botón Generar Nuevo Detalle. Volver al paso 3.
	4b	Para crear otro ingreso, dar click en el botón Generar Nuevo Ingreso. Volver al paso 2.
5	Para agregar nuevos detalles en un ingreso existente, buscamos el ingreso por su número. Dar click en el botón con un ícono en forma de lupa. Volver al paso 2.	Se carga la cabecera y detalles del ingreso buscado.
6	Para cancelar un ingreso, buscamos el ingreso por su número. Dar click en el botón Cancelar.	Muestra un cuadro de diálogo, donde solicita la confirmación de la eliminación.

	6a	Si el usuario confirma la cancelación del ingreso, se muestra un mensaje con el resultado de la acción realizada.	
	6b	Si el usuario no confirma la cancelación de datos, el sistema no realiza ninguna acción.	
7		Para modificar o eliminar el detalle de un ingreso registrado previamente, Seleccionar de la opción: Ingresos, el submenú Modificar/Eliminar Detalles Ingreso.	Carga el formulario Modificar/Eliminar Detalles de Ingreso.
8		Se procede a buscar el ingreso; la búsqueda se hace por el campo Nro. Movimiento.	Carga los datos de la lista de la cabecera y detalles del movimiento buscado.
9		Para modificar un detalle del ingreso correspondiente. Seleccionarlo de la lista de detalles. Dar click en el botón con un ícono en forma de lápiz.	
	9.a	Si el usuario confirma la modificación, se cargan los campos respectivos. Dar click en el botón Modificar. Se actualiza el listado de detalles de ingreso.	
	9.b	Si el usuario no confirma la modificación, el sistema no realiza ninguna acción.	
10		Para eliminar un detalle del ingreso correspondiente. Seleccionarlo de la lista de detalles. Dar click en el botón con un ícono en forma de papelera.	
	10.a	Si el usuario confirma la eliminación, se registra ésta y se actualiza el listado de detalles de ingreso.	
	10.b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.	

ESPACIO EN BLANCO INTENCIONAL

Nombre		Administrar Requisición
Identificador		CU-I04
Actor		Administrador, Técnico.
Descripción		El sistema deberá permitir al usuario Técnico o Administrador crear, modificar o cancelar una requisición de las distintas bodegas de la empresa.
Precondición		Usuario debe estar autenticado bajo perfil de Técnico o Administrador, acceder al módulo de Inventario, sección Requisición de Producto.
Postcondición		Requisiciones administradas e inventario actualizado.
Paso	Acción	
	Actor	Sistema
1	Seleccionar la opción Crear Requisición.	Carga el formulario de Cabecera de Requisición.
2	Ingresa la siguiente información en el formulario: (No existen campos mandatorios) • Fecha. •Observación. Dar click en el botón Guardar.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	2b	Si no se guarda la cabecera, el sistema no permitirá que se avance para guardar los detalles de la requisición.
3	Se procede a buscar el producto para la requisición; la búsqueda puede realizarse por: Código CTK o nombre.	Carga los datos del producto seleccionado.
4	Ingresa la siguiente información en el formulario de Detalle del ingreso: (Todos los campos son mandatorios) •Ubicación: Nombre de la ciudad donde se va a registrar el ingreso. •Bodega: Nombre de la bodega en la que se va a registrar el ingreso. •Cantidad a solicitar: cantidad del producto que se va a solicitar. Dar click en el botón Guardar Detalle.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle-Requisición. Informa el resultado de la acción realizada.
	4a	Para agregar otro detalle a la misma requisición, dar click en el botón Generar Nuevo Detalle. Volver al paso 3.
	4b	Para crear otra requisición, dar click en el botón Generar Nueva Requisición. Volver al paso 2.
5	Para agregar nuevos detalles en una requisición existente, lo buscamos por su número. Dar click en el botón con un ícono en forma de lupa. Volver al paso 2.	Se carga la cabecera y detalles del ingreso buscado.
6	Para cancelar una requisición, la buscamos por su número. Dar click en el botón Cancelar.	Muestra un cuadro de diálogo, donde solicita la confirmación de la eliminación.

	6a	Si el usuario confirma la cancelación de la requisición, se muestra un mensaje con el resultado de la acción realizada.	
	6b	Si el usuario no confirma la cancelación de la requisición, el sistema no realiza ninguna acción.	
7		Para modificar o eliminar el detalle de una requisición registrada previamente, seleccionar la opción: Modificar/Eliminar Detalles Requisición.	Carga el formulario Modificar/Eliminar Detalles de Requisición.
8		Se procede a buscar la requisición; la búsqueda se hace por el campo Nro. Requisición.	Carga los datos de la lista de la cabecera y detalles de la requisición buscada.
9		Para modificar un detalle de la requisición correspondiente, seleccionarlo de la lista de detalles. Dar click en el botón con un ícono en forma de lápiz.	
	9a	Si el usuario confirma la modificación, se cargan los campos respectivos. Dar click en el botón Modificar. Se actualiza el listado de detalles de requisición.	
	9b	Si el usuario no confirma la modificación, el sistema no realiza ninguna acción.	
10		Para eliminar un detalle de la requisición correspondiente, seleccionarlo de la lista de detalles. Dar click en el botón con un ícono en forma de papelera.	
	10a	Si el usuario confirma la eliminación, se registra ésta y se actualiza el listado de detalles de requisición.	
	10b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.	

ESPACIO EN BLANCO INTENCIONAL

Nombre	Gestionar Egreso	
Identificador	CU-I05	
Actor	Administrador, Bodeguero.	
Descripción	El sistema deberá permitir al usuario Técnico o Administrador registrar la entrega del producto de una de las distintas bodegas de la empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Bodeguero o Administrador, acceder al módulo de Inventario, sección Administración Inventario.	
Postcondición	Entrega de producto registrada.	
Paso	Acción	
	Actor	Sistema
1	Seleccionar de la opción: Egresos, el submenú Gestionar Egreso.	Carga el formulario de Gestionar Egreso.
2	Ingresar la siguiente información en el formulario: (Todos los campos son mandatorios) •Nro. Requisición. •Ubicación •Bodega/Cliente. Dar click en el botón Buscar.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	2b	Si se encuentra una requisición coincidente, se carga la lista de detalles de productos por entregar de la requisición buscada.
3	Para registrar la entrega de producto, seleccionar el detalle a entregar. Dar click en el botón Entregar.	
	3a	Si el usuario confirma la entrega del producto, se muestra un mensaje con el resultado de la acción realizada y se actualiza la lista de detalles de requisición por entregar.
	3b	Si el usuario no confirma la entrega del producto, el sistema no realiza ninguna acción.
4	Para actualizar el estado de una requisición, seleccionar de la opción: Egresos, el submenú: Actualizar Requisiciones.	Carga el formulario Actualizar Requisiciones.
5	Ingresar la siguiente información en el formulario: (Todos los campos son mandatorios) •Ubicación •Bodega/Cliente. Dar click en el botón Buscar.	
	9a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	9b	Si se encuentra algún detalle de requisición coincidente, se carga la lista de detalles de requisiciones aún sin retirar.
10	Para cancelar un detalle de la requisición correspondiente, seleccionarlo de la lista de detalles de requisiciones aún sin retirar. Dar click en el botón cancelar.	
	10a	Si el usuario confirma la cancelación, se registra ésta y se actualiza el listado de detalles de requisición.
	10b	Si el usuario no confirma la cancelación, el sistema no realiza ninguna acción.

Nombre	Generar Kárdex de Producto	
Identificador	CU-I06	
Actor	Administrador, Técnico, Bodeguero	
Descripción	El sistema deberá permitir al usuario Bodeguero, Técnico o Administrador generar el Kárdex de producto en base a las bodegas donde se encuentre inventariado.	
Precondición	Usuario debe estar autenticado bajo perfil de Bodeguero, Técnico o Administrador acceder al módulo de Inventario, sección Reportes.	
Postcondición	Kárdex de productos que coinciden con el/los campos de búsqueda.	
Paso	Acción	
	Actor	Sistema
1	Ingresar al menos uno de los siguientes campos: Código CTK: Código interno del producto dentro de la empresa. Ubicación: Ubicación del Producto. Bodega/Cliente: Bodega en la que se encuentra el producto. Presionar el botón Visualizar Kárdex.	Muestra el listado de productos que coincidan con los campos de búsqueda.
	1a	Si la búsqueda no devuelve ningún registro, se mostrará el mensaje correspondiente para el usuario.
	1b	Si no se ingresa ningún campo de búsqueda se mostrará el mensaje para el usuario.
	1c	Si solo se ingresa el campo Código CTK; la única forma de visualizar el kárdex será mediante un archivo de formato PDF.
2	Para realizar una nueva búsqueda, presionar el botón Limpiar Campos, y volver al paso 1.	El sistema limpia todos los campos de búsqueda y la tabla Listado-Inventario.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Consultar existencias producto.	
Identificador	CU-I07	
Actor	Administrador, Técnico, Bodeguero	
Descripción	El sistema deberá permitir al usuario Bodeguero, Técnico o Administrador realizar una consulta de todos los productos que cumplan con uno/varios de los campos de búsqueda que se ingresen.	
Precondición	Usuario debe estar autenticado bajo perfil de Bodeguero, Técnico o Administrador acceder al módulo de Inventario, sección Búsquedas.	
Postcondición	Listado de productos que coinciden con el/los campos de búsqueda.	
Paso	Acción	
	Actor	Sistema
1	Ingresar al menos uno de los siguientes campos: Código CTK: Código interno del producto dentro de la empresa. Producto: Descripción parcial o completa del Producto. Otro Código: Identificativo adicional del producto Marca: Marca del Producto Modelo: Modelo del Producto Serial: Serial del Producto Estado: Estado del Producto(Bueno, Dañado, Defectuoso) Área: Área a la que pertenece el producto(Administrativa, Técnica, General) Categoría: Categoría del Producto (Herramienta, Material, Equipo, Otro) Ubicación: Ubicación del Producto. Bodega/Cliente: Bodega en la que se encuentra el producto. Presionar el botón Buscar para verificar la existencia de alguna existencia.	Muestra el listado de productos que coincidan con los campos de búsqueda.
	1a	Si la búsqueda no devuelve ningún registro, se mostrará el mensaje correspondiente para el usuario.
	1b	Si no se ingresa ningún campo de búsqueda se mostrará el mensaje para el usuario.
2	Para realizar una nueva búsqueda , presionar el botón Limpiar Campos, y volver al paso 1	El sistema limpia todos los campos de búsqueda y la tabla Listado-Inventario.

Nombre	Administrar Orden de Trabajo	
Identificador	CU-I08	
Actor	Administrador, Técnico.	
Descripción	<p>El sistema deberá permitir al usuario Técnico o Administrador: crear/modificar/visualizar únicamente una Orden de Trabajo a la vez.</p> <p>Deberá permitir al usuario Administrador aprobar una Orden de Trabajo.</p> <p>Deberá permitir al usuario Técnico o Administrador: registrar la realización y entregar una Orden de Trabajo.</p>	
Precondición	<p>Usuario debe estar autenticado bajo perfil de Técnico o Administrador, acceder al módulo de Inventario, sección Administración Proyectos.</p> <p>Para modificar o aprobar una Orden de Trabajo, es necesario que haya sido creada anteriormente.</p> <p>Para registrar la realización de una Orden de Trabajo, es necesario que haya sido aprobada o confirmada su cotización anteriormente.</p>	
Postcondición	Orden de Trabajo actualizada.	
Paso	Acción	
	Actor	Sistema
1	Seleccionar la opción Crear/Modificar Orden de Trabajo del botón Orden de Trabajo ubicado en el menú superior.	Muestra el formulario "Administrar Orden de Trabajo" con el botón para Crear una Nueva Orden de Trabajo, y; un campo de búsqueda por tema junto con el listado de Órdenes de Trabajo Modificables.
2	Para crear una nueva Orden de Trabajo, presionar el botón correspondiente.	Carga la plantilla de la Orden de Trabajo limpia, con el número de Orden que corresponde, la fecha actual y el campo Costo predefinido con el valor de 1.0
3	<p>Ingresar la siguiente información en el formulario: (Los campos mandatorios están marcados con un *)</p> <ul style="list-style-type: none"> • Fecha •Tema*: Tema de la Orden de Trabajo. •Nombre*: Seleccionar el Cliente/Contacto que solicita la realización de la Orden de Trabajo; y presionar <<ENTER>> o dar click sobre el botón con un ícono en forma de una lupa, para buscar los campos de este. •Forma de Pago: <<COMBO>> seleccionar la forma de pago. •ADM-LIQUIDACIÓN VIÁTICOS •TEC-INFORME •ADM-SOLICITUD COMPRA •DETALLES •Costo: ingresar el costo total de la Orden de Trabajo, se deja el valor por default (1.0) si se requiere cotización. Ver CU-I09. •Fecha Entrega*: seleccionar fecha de entrega de la Orden de Trabajo. •Observaciones <p>Dar click en el botón Guardar</p>	<p>Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido.</p> <p>Informa el resultado de la acción realizada.</p>

	3a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.	
4		Para generar el PDF de la Orden de Trabajo, dar click en el botón Generar PDF.	Genera el PDF de la Orden de Trabajo.
5		Para modificar una Orden de Trabajo, seleccione una de la lista, o buscarla en base a su Tema o Nro. De Orden, y dar click en el botón en forma de lápiz.	Muestra el Listado- Ordenes de Trabajo Modificables completo o filtrado por un campo de búsqueda.
6		Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	6a	Si el usuario confirma la modificación de la Orden de Trabajo, se carga el formulario de la Orden de Trabajo con los datos de su última actualización. Volver al paso 3.	
	6b	Si el usuario no confirma la modificación de la Orden de Trabajo, el sistema no realiza ninguna acción.	
7		Para Aprobar una Orden de Trabajo; seleccionar la opción Órdenes de Trabajo por Aprobar, del botón Orden de Trabajo, ubicado en el menú superior.	Muestra el formulario "Aprobar Orden de Trabajo" con las opciones de búsqueda por Nro. De Orden o Tema, y el listado de Órdenes de Trabajo por Aprobar.
8		Seleccionar la Orden de Trabajo a aprobar de la lista, o buscarla en base a su Tema o Nro. De Orden, y dar click en el botón correspondiente.	Muestra el Listado- Órdenes de Trabajo por Aprobar completo, o filtrado por un campo de búsqueda.
	8a	Para aprobar la realización de la Orden de Trabajo, dar click sobre el botón con un ícono en forma de un visto, de la orden correspondiente. Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	8a.1	Si el usuario confirma la realización de la Orden de Trabajo, se registra ésta, y se actualiza el Listado- Órdenes por Aprobar.	
	8a.2	Si el usuario no confirma la realización de la Orden de Trabajo, el sistema no realiza ninguna acción.	
	8b	Para rechazar la realización de la Orden de Trabajo, dar click sobre el botón con un ícono en forma de X, de la orden correspondiente. Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	8ba	Si el usuario confirma que se rechaza la Orden de Trabajo, se registra ésta, y se actualiza el Listado- Órdenes por Aprobar.	
	8bb	Si el usuario no confirma que se rechaza la Orden de Trabajo, el sistema no realiza ninguna acción.	
9		Para cotizar una Orden de Trabajo; ver CU-0.	
10		Para registrar la Realización de una orden de Trabajo; seleccionar la opción Registrar Órdenes Realizadas del botón Orden de Trabajo ubicado en el menú superior.	Muestra el formulario "Realización Orden de Trabajo" con las opciones de búsqueda por Nro. De Orden o Tema, y el listado de Órdenes de Trabajo por Realizar.
11		Seleccionar la Orden de Trabajo que se realizó, de la lista o buscarla en base a su Tema o Nro. De Orden, y dar click en el botón con un ícono en forma de visto.	Muestra el Listado- Órdenes de Trabajo por Realizar completo, o filtrado por un campo de búsqueda.
12		Selecciona del cuadro de diálogo emergente, la acción a realizar.	

	12 a	Si el usuario confirma la realización de la Orden de Trabajo, se registra ésta, y se actualiza el Listado- Órdenes por Realizar.	
	12 b	Si el usuario no confirma la realización de la Orden de Trabajo, el sistema no realiza ninguna acción.	
13		Para registrar la Entrega de una orden de Trabajo; seleccionar la opción Entregar Orden de Trabajo, del botón Orden de Trabajo ubicado en el menú superior.	Muestra el formulario "Entrega Orden de Trabajo" con las opciones de búsqueda por Nro. De Orden o Tema, y el listado de Órdenes de Trabajo por Entregar.
14		Seleccionar la Orden de Trabajo que se va a entregar, de la lista, o buscarla en base a su Tema o Nro. De Orden, y dar click en el botón con un ícono en forma de visto.	Muestra el Listado- Órdenes de Trabajo por Realizar completo, o filtrado por un campo de búsqueda.
15		Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	15 a	Si el usuario confirma la entrega de la Orden de Trabajo, se registra ésta, y se actualiza el Listado- Órdenes por Entregar.	
	15 b	Si el usuario no confirma la entrega de la Orden de Trabajo, el sistema no realiza ninguna acción.	
16		Para visualizar una Orden de Trabajo en cualquiera de sus etapas; seleccionar la opción Visualizar O.T., ubicado en el menú superior.	Muestra el formulario "Visualización Orden de Trabajo" con las opciones de búsqueda por Nro. De Orden o Tema, y el listado de Órdenes de Trabajo Existentes.
17		Seleccionar la Orden de Trabajo que se desea visualizar, de la lista, o buscarla en base a su Tema o Nro. De Orden.	Muestra el Listado- Órdenes de Trabajo Existentes completo, o filtrado por un campo de búsqueda.
18		Para generar el formato de la Orden de Trabajo dar click sobre el link PDF.	Genera el PDF de la Orden de Trabajo.
19		Para visualizar la Orden de Trabajo, dar click en el botón con un ícono en forma de lupa. Seleccionar del cuadro de diálogo emergente, la acción a realizar.	
	19 a	Si el usuario confirma la visualización de la Orden de Trabajo, se carga el formulario de ésta con los últimos datos actualizados.	
	19 b	Si el usuario no confirma la visualización de la Orden de Trabajo, el sistema no realiza ninguna acción.	

Nombre		Administrar Cotización
Identificador		CU-I09
Actor		Administrador, Técnico.
Descripción		El sistema deberá permitir al usuario Técnico o Administrador: crear/modificar/visualizar únicamente una Cotización a la vez. Deberá permitir al usuario Administrador aprobar una Cotización.
Precondición		Usuario debe estar autenticado bajo perfil de Técnico o Administrador, acceder al módulo de Inventario, sección Administración Proyectos. Para modificar o aprobar una Cotización, es necesario que haya sido creada anteriormente.
Postcondición		Cotización actualizada.
Paso	Acción	
	Actor	Sistema
1	Seleccionar la opción Órdenes de Trabajo por Cotizar, del botón Orden de Trabajo ubicado en el menú superior.	Muestra el formulario "Administrar Orden de Trabajo" con las opciones de búsqueda por Nro. De Orden o Tema, y el listado de Órdenes de Trabajo por Cotizar.
2	Para crear una nueva Orden de Trabajo, presionar el botón correspondiente.	Carga la plantilla de la Orden de Trabajo limpia, con el número de Orden que corresponde, la fecha actual y el campo Costo predefinido con el valor de 1.0.
3	Seleccionar la Orden de Trabajo a cotizar de la lista, o buscarla en base a su Tema o Nro. De Orden.	Muestra el Listado – Órdenes de Trabajo por Cotizar completo, o filtrado por un campo de búsqueda.
	3a	Para aprobar la cotización de la Orden de Trabajo, dar click sobre el botón Cotizar, de la Orden correspondiente. Selecciona del cuadro de diálogo emergente, la acción a realizar.
	3a.1	Si el usuario confirma la cotización de la Orden de Trabajo, se carga el formulario para la Cotización.
	3a.2	Si el usuario no confirma la cotización de la Orden de Trabajo, el sistema no realiza ninguna acción.
4	Ingresar la siguiente información en el formulario de la Cabecera de la cotización: (Los campos mandatorios están marcados con un *) • Fecha*: Seleccionar fecha de creación de la Orden de Trabajo. Dar click en el botón Guardar	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	4a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	4b	Si no se guarda la cabecera no se permitirá, avanzar al ingreso de detalles de la cotización.

5	Ingresar la siguiente información en el formulario de Ingreso de Detalle-Materiales de la cotización: (Todos los campos son mandatorios) •Descripción: Nombre del Material requerido para la Orden de Trabajo. •Cantidad: Número de Material necesario para la Orden de Trabajo. •Precio Unitario: Costo Unitario del Material. Dar click en el botón con un ícono en forma de visto.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle-Materiales y calcula el precio total del detalle ingresado. Informa el resultado de la acción realizada.
	5a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
6	Para modificar un detalle de material, se selecciona uno de la lista de Detalle-Materiales dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
7	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	7a	Si el usuario confirma la modificación del detalle de materiales, se cargan los campos en el formulario. Volver al paso 5.
	7b	Si el usuario no confirma la modificación del detalle de materiales, el sistema no realiza ninguna acción.
8	Para eliminar un detalle de material, se selecciona uno de la lista de Detalle-Materiales dando click en el botón con un ícono en forma de papelera.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
	8a	Si el usuario confirma la eliminación del detalle de materiales, se efectúa la acción y se actualiza el listado de detalles de materiales.
	8b	Si el usuario no confirma la modificación del detalle de materiales, el sistema no realiza ninguna acción.
9	Ingresar la siguiente información en el formulario de Ingreso de Detalle-Mano de Obra de la cotización: (Todos los campos son mandatorios) •Descripción: Explicación de la actividad que se realizará para la Orden de Trabajo. •Cantidad: Número de veces que se ejecutará la actividad en la Orden de Trabajo. •Precio Unitario: Costo Unitario de la actividad que se realizará en la Orden de Trabajo. Dar click en el botón con un ícono en forma de visto.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle-Mano de Obra y calcula el precio total del detalle ingresado. Informa el resultado de la acción realizada.
	9a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
10	Para modificar un detalle de mano de obra, se selecciona uno de la lista de Detalle-Mano de Obra dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.

11	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	11a	Si el usuario confirma la modificación del detalle de mano de obra, se cargan los campos en el formulario. Volver al paso 9.
	11b	Si el usuario no confirma la modificación del detalle de mano de obra, el sistema no realiza ninguna acción.
12	Para eliminar un detalle de mano de obra, se selecciona uno de la lista de Detalle-Mano de Obra dando click en el botón con un ícono en forma de papelera.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
	12a	Si el usuario confirma la eliminación del detalle de mano de obra, se efectúa la acción y se actualiza el listado de detalles de mano de obra.
	12b	Si el usuario no confirma la modificación del detalle de mano de obra, el sistema no realiza ninguna acción.
13	Ingresar la siguiente información en el Pie de la cotización: (Los campos requeridos están marcados con un *) •Tiempo de Entrega*: Fecha en la que se entrega la Orden de Trabajo. •Condiciones de Pago*: Contexto en el que se va a cancelar la Orden de Trabajo. •Validez de la Oferta*: Tiempo por el cual los valores de la cotización serán válidos. •Observaciones: Detalle adicional de la cotización. •Nota: Detalle adicional con respecto a materiales o mano de obra que consta en la cotización. Dar click en el botón Guardar.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	13a	Si los campos ingresados no son válidos, el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
14	Para generar el PDF de la Cotización, dar click en el botón Generar PDF.	Genera el PDF de la Cotización.
15	Para modificar una Cotización, seleccionar la opción Modificar Cotización, del botón Cotización ubicado en el menú superior.	Muestra el formulario "Modificar Cotización" con las opciones de búsqueda por Nro. De Cotización o Tema, y el listado de Cotizaciones Modificables.
16	Seleccionar la cotización de la lista de Listado-Cotizaciones Modificables dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
17	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	17a	Si el usuario confirma la modificación de la Cotización, se carga el formulario de la Cotización con los datos de su última actualización. Volver al paso 4.
	17b	Si el usuario no confirma la modificación de la Cotización, el sistema no realiza ninguna acción.
18	Para confirmar o rechazar una Cotización, seleccionar la opción Confirmar/Rechazar	Muestra el formulario "Confirmar/Rechazar Cotización"

	Cotización, del botón Cotización ubicado en el menú superior.	con las opciones de búsqueda por Nro. De Cotización o Tema, y el listado de Cotizaciones por Confirmar.
18a	Para confirmar la Orden de Trabajo, dar click sobre el botón con un ícono en forma de un visto, de la cotización correspondiente. Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	18a.1	Si el usuario confirma la Cotización, se registra ésta, y se actualiza el Listado- Cotizaciones por Confirmar.
	18a.2	Si el usuario no confirma la Cotización, el sistema no realiza ninguna acción.
18b	Para rechazar la Cotización, dar click sobre el botón con un ícono en forma de X, de la orden correspondiente. Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	18b.1	Si el usuario rechaza la Cotización, se registra ésta, y se actualiza el Listado- Cotizaciones por Confirmar.
	18b.2	Si el usuario no rechaza la Cotización, el sistema no realiza ninguna acción.
19	Para visualizar una Cotización en cualquiera de sus etapas; seleccionar la opción Visualizar Cotización, ubicado en el menú superior.	Muestra el formulario "Visualización Cotización" con las opciones de búsqueda por Nro. De Cotización o Tema, y el listado de Cotizaciones Existentes.
20	Seleccionar la Cotización que se desea visualizar, de la lista, o buscarla en base a su Tema o Nro. De Cotización.	Muestra el Listado- Cotizaciones Existentes completo, o filtrado por un campo de búsqueda.
21	Para generar el formato de la Cotización dar click sobre el link PDF.	Genera el PDF de la Orden de Trabajo.
22	Para visualizar la Cotización, dar click en el botón con un ícono en forma de lupa. Seleccionar del cuadro de diálogo emergente, la acción a realizar.	
	22a	Si el usuario confirma la visualización de la Cotización, se carga el formulario de ésta con los últimos datos actualizados.
	22b	Si el usuario no confirma la visualización de la Cotización, el sistema no realiza ninguna acción.

ESPACIO EN BLANCO INTENCIONAL

4.7.4 Diagrama de C.U módulo IV: Gestión Contable

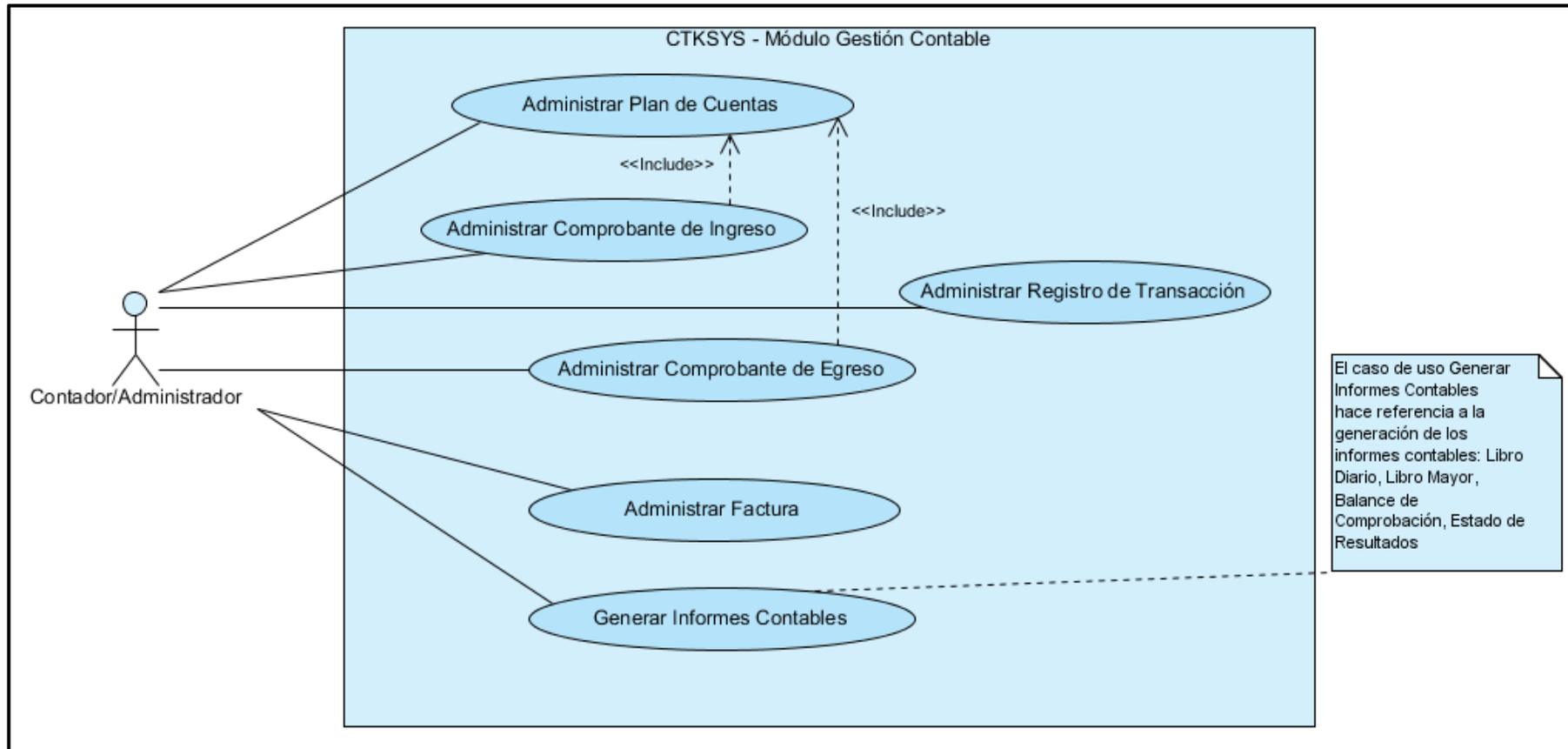


Figura 4.18: Diagrama C.U módulo Gestión Contable

4.7.4.1 Descripción de casos de uso módulo IV: Gestión Contable

Nombre	Administrar Plan de Cuenta	
Identificador	CU-C01	
Actor	Administrador, Contador.	
Descripción	El sistema deberá permitir al usuario Contador o Administrador gestionar el Plan de Cuentas de la Empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Contador o Administrador, acceder al módulo de Gestión Contable, sección Administración Plan de Cuentas.	
Postcondición	Cuenta Contable actualizada y lista para ser usada.	
Paso	Acción	
	Actor	Sistema
1	Ingrese el código de la cuenta padre a buscar. Dar click en el botón con un ícono en forma de lupa.	Carga los campos Código Cuenta Padre y el prefijo de la Cuenta a Ingresar del formulario del Plan de Cuenta.
	1a	Si el campo ingresado no es válido, el sistema deberá mostrar un mensaje de error.
2	Ingresar la siguiente información en el formulario: (Todos los campos son mandatorios) • Código Cuenta: Sufijo de la Cuenta. • Nombre Cuenta: Identificativo de la cuenta en el Plan. Dar click en el botón Guardar.	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
3	Para modificar una cuenta, seleccione una de la lista, o buscarla en base a su cuenta padre, y dar click en el botón en forma de lápiz.	Muestra el Listado- Plan de cuenta Modificables completo o filtrado en base a una cuenta padre.
4	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	4a	Si el usuario confirma la modificación de la cuenta, se carga el formulario de ésta, con los datos de su última actualización. Modificar campos necesario, dar click en el botón Modificar.
	4b	Si el usuario no confirma la modificación de la cuenta, el sistema no realiza ninguna acción.
5	Para Eliminar una cuenta, seleccione una de la lista, o buscarla en base a su cuenta padre, y dar click en el botón en forma de papelera.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
	5a	Si el usuario confirma la eliminación de la cuenta, se ejecuta ésta, y se actualiza el Listado-Plan de Cuentas.
	5b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.

Nombre	Administrar Comprobante de Ingreso	
Identificador	CU-C02	
Actor	Administrador, Contador.	
Descripción	El sistema deberá permitir al usuario Contador o Administrador gestionar comprobantes de ingreso para la empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Contador o Administrador, acceder al módulo de Gestión Contable, sección Administración de Comprobantes.	
Postcondición	Actualización y generación del formato de Comprobante de Ingreso	
Paso	Acción	
	Actor	Sistema
1	Ingresar la siguiente información en el formulario de la Cabecera del comprobante de Ingreso: (Los campos mandatorios están marcados con un *) • Recibido de*: empresa o cliente que va a hacer un ingreso económico hacia la empresa. •Efectivo. •USD*: valor económico por el que se genera el comprobante. •Cheque N°. •Banco Por concepto*: motivo por el que se va a generar el comprobante. Dar click en el botón Guardar	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	1a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	1b	Si no se guarda la cabecera no se permitirá, avanzar al ingreso de detalles del comprobante.
	1c	Si se genera el comprobante por un anticipo relacionado a una Orden de Trabajo se marca la casilla correspondiente y se llena el campo: Relacionar con O. de Trabajo con el número correspondiente.
	1d	Si se genera el comprobante por el registro del pago total de una Orden de Trabajo se llena el campo: Relacionar con O. de Trabajo con el número correspondiente.
2	Ingresar la siguiente información en el formulario de Detalle Comprobante de Ingreso: (Todos los campos son mandatorios) •Código: Código de la Cuenta en el plan. •Debe/Haber: valor del debe o del haber según corresponda. Dar click en el botón con un ícono en forma de visto.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle y calcula la suma de las columnas debe y haber. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos, el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	2b	Si se trata del registro del pago total de una Orden de Trabajo, marcar la casilla correspondiente.
3	Para modificar un detalle del comprobante, se selecciona uno de la lista dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
4	Selecciona del cuadro de diálogo emergente, la acción a realizar.	

	4a	Si el usuario confirma la modificación del detalle, se carga el formulario de éste, con los datos de su última actualización. Volver al paso 2.	
	4b	Si el usuario no confirma la modificación del detalle, el sistema no realiza ninguna acción.	
5		Para Eliminar un detalle del comprobante, se selecciona uno de la lista, y dar click en el botón en forma de papelera.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
6		Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	6a	Si el usuario confirma la eliminación del detalle, se ejecuta ésta, y se actualiza el Listado de detalles.	
	6b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.	
7		Para generar el PDF del Comprobante, dar click en el botón Generar PDF.	Genera el PDF del Comprobante.
		Para modificar un Comprobante, buscarlo por su número y dar click sobre el botón con un ícono en forma de lupa. Volver al paso 1.	Carga el formulario de Comprobante con los datos de su última actualización y el listado de Detalles.
9		Para cancelar un Comprobante, buscarlo por su número y dar click sobre el botón Cancelar.	Registra la cancelación del comprobante. Informa el resultado de la acción realizada.
	9a	Si el comprobante, está relacionado a una Orden de Trabajo, no se podrá efectuar la cancelación y el sistema deberá mostrar un mensaje de error.	

ESPACIO EN BLANCO INTENCIONAL

Nombre	Administrar Comprobante de Egreso	
Identificador	CU-C03	
Actor	Administrador, Contador.	
Descripción	El sistema deberá permitir al usuario Contador o Administrador gestionar comprobantes de egreso para la empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Contador o Administrador, acceder al módulo de Gestión Contable, sección Administración de Comprobantes.	
Postcondición	Actualización y generación del formato de Comprobante de Egreso.	
Paso	Acción	
	Actor	Sistema
1	Ingresar la siguiente información en el formulario de la Cabecera del comprobante de Egreso: (Los campos mandatorios están marcados con un *) <ul style="list-style-type: none"> • Beneficiario*: empresa o cliente para la cual se hace el egreso económico desde la empresa. •Efectivo. •USD*: valor económico por el que se genera el comprobante. •Cheque N°. •Banco. •Retención Nro. •Por concepto*: motivo por el que se va a generar el comprobante. Dar click en el botón Guardar	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	1a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	1b	Si no se guarda la cabecera no se permitirá, avanzar al ingreso de detalles del comprobante.
2	Ingresar la siguiente información en el formulario de Detalle Comprobante de Egreso: (Todos los campos son mandatorios) <ul style="list-style-type: none"> •Código: Código de la Cuenta en el plan. •Debe/Haber: valor del debe o del haber según corresponda. Dar click en el botón con un ícono en forma de visto.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle y calcula la suma de las columnas debe y haber. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos, el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	2b	Si se trata del registro del pago total, marcar la casilla correspondiente.
3	Para modificar un detalle del comprobante, se selecciona uno de la lista dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
4	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	4a	Si el usuario confirma la modificación del detalle, se carga el formulario de éste, con los datos de su última actualización. Volver al paso 2.
	4b	Si el usuario no confirma la modificación del detalle, el sistema no realiza

		ninguna acción.
5	Para Eliminar un detalle del comprobante, se selecciona uno de la lista, y dar click en el botón en forma de papelera.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
6	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	6a	Si el usuario confirma la eliminación del detalle, se ejecuta ésta, y se actualiza el Listado de detalles.
	6b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.
7	Para generar el PDF del Comprobante, dar click en el botón Generar PDF.	Genera el PDF del Comprobante.
8	Para modificar un Comprobante, buscarlo por su número y dar click sobre el botón con un ícono en forma de lupa. Volver al paso 1.	Carga el formulario de Comprobante con los datos de su última actualización y el listado de Detalles.
9	Para cancelar un Comprobante, buscarlo por su número y dar click sobre el botón Cancelar.	Registra la cancelación del comprobante. Informa el resultado de la acción realizada.
	9a	Si el comprobante, está relacionado a una Orden de Trabajo, no se podrá efectuar la cancelación y el sistema mostrará un mensaje de error.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Administrar Factura	
Identificador	CU-C04	
Actor	Administrador, Contador.	
Descripción	El sistema deberá permitir al usuario Contador o Administrador realizar el proceso de facturación en la empresa.	
Precondición	Usuario debe estar autenticado bajo perfil de Contador o Administrador, acceder al módulo de Gestión Contable, sección Facturación.	
Postcondición	Actualización y generación del formato de Facturación.	
Paso	Acción	
	Actor	Sistema
1	Ingresar la siguiente información en el formulario de la Cabecera de la factura: (Los campos mandatorios están marcados con un *) • Cliente*: empresa para la que se genera la factura • Fecha. Dar click en el botón Guardar	Verifica que los campos requeridos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Informa el resultado de la acción realizada.
	1a	Si los campos ingresados no son válidos el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
	1b	Si no se guarda la cabecera no se permitirá, avanzar al ingreso de detalles del comprobante.
	1c	Si se genera la factura por una Orden de Trabajo se llena el campo: "Relacionar con una Orden de Trabajo" con el número correspondiente.
2	Ingresar la siguiente información en el formulario de Detalles de factura: (Todos los campos son mandatorios) • Cantidad: número por el cual se va generar el detalle de la factura. • Descripción: detalle del cuerpo de la factura. • V. Unit.: costo unitario de la descripción. Dar click en el botón con un ícono en forma de visto.	Verifica que los campos hayan sido llenados correctamente, y que el tamaño de los datos esté dentro del rango establecido. Actualiza la lista de Detalle y calcula el campo Valor Total del detalle, y actualiza los campos: Subtotal, I.V.A, Total del pie de la factura. Informa el resultado de la acción realizada.
	2a	Si los campos ingresados no son válidos, el sistema deberá mostrar un mensaje de error por el o los campos mal ingresados.
3	Para modificar un detalle de la factura, se selecciona uno de la lista dando click en el botón con un ícono en forma de lápiz.	Despliega cuadro de diálogo emergente, para confirmación de la acción a realizar.
4	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	4a	Si el usuario confirma la modificación del detalle, se carga el formulario de éste, con los datos de su última actualización. Volver al paso 2.
	4b	Si el usuario no confirma la modificación del detalle, el sistema no realiza ninguna acción.
5	Para Eliminar un detalle de la factura, se selecciona uno de la lista, y dar click en el botón	Despliega cuadro de diálogo emergente, para confirmación de

	en forma de papelera.	la acción a realizar.
6	Selecciona del cuadro de diálogo emergente, la acción a realizar.	
	6a	Si el usuario confirma la eliminación del detalle, se ejecuta ésta, y se actualiza el Listado de detalles.
	6b	Si el usuario no confirma la eliminación, el sistema no realiza ninguna acción.
7	Para generar el PDF de la factura, dar click en el botón Generar PDF.	Genera el PDF del Comprobante.
8	Para modificar una Factura, buscarla por su número y dar click sobre el botón con un ícono en forma de lupa. Volver al paso 1.	Carga el formulario de Factura con los datos de su última actualización y el listado de Detalles.
9	Para cancelar un Comprobante, buscarlo por su número y dar click sobre el botón Cancelar.	Registra la cancelación del comprobante. Informa el resultado de la acción realizada.
	9a	Si la factura, está relacionado a una Orden de Trabajo, no se podrá efectuar la cancelación y el sistema mostrará un mensaje de error.

ESPACIO EN BLANCO INTENCIONAL

Nombre	Generar Informes Contables	
Identificador	CU-C05	
Actor	Administrador, Contador.	
Descripción	El sistema deberá permitir al usuario Contador o Administrador generar los informes contables.	
Precondición	Usuario debe estar autenticado bajo perfil de Contador o Administrador, acceder al módulo de Gestión Contable, sección Reportes.	
Postcondición	Generación de los Informes Contables en formato PDF.	
Paso	Acción	
	Actor	Sistema
1	Seleccionar del formulario de reportes el informe contable que se desea generar.	Muestra el formulario correspondiente a cada informe contable, acompañado de una corta descripción de este.
2	Seleccionar la opción: "Libro Diario". Dar click en el botón: Generar Libro Diario.	Genera el libro diario en formato PDF desde el inicio del año contable a la fecha actual.
3	Seleccionar la opción: "Libro Mayor". Dar click en el botón: Generar Libro Mayor.	Genera el libro mayor en formato PDF desde el inicio del año contable a la fecha actual.
4	Seleccionar la opción: "Balance de Comprobación". Dar click en el botón: Generar Balance de Comprobación.	Genera el balance de comprobación en formato PDF desde el inicio del año contable a la fecha actual.
5	Seleccionar la opción: "Estado de Resultados". Dar click en el botón: Generar Estado de Resultados.	Genera el estado de resultados en formato PDF desde el inicio del año contable a la fecha actual.

ESPACIO EN BLANCO INTENCIONAL

4.8 Diagrama Entidad – Relación CTKSYS (Módulo de Inventario)

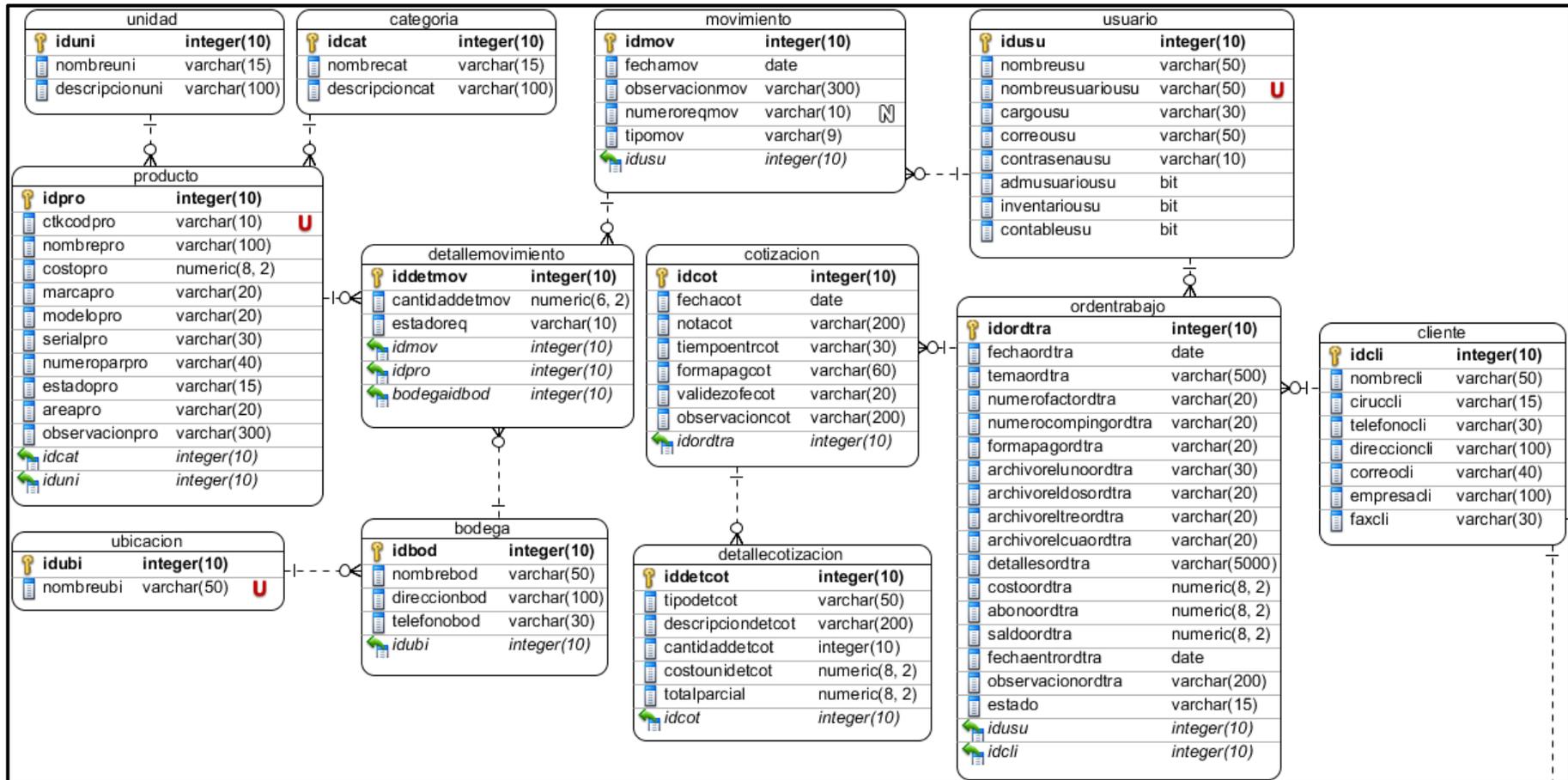


Figura 4.19: Diagrama Entidad Relación CTKWEB (Módulo de Inventario)

4.9 Diagrama Entidad – Relación CTKSYS (Módulo Contable)

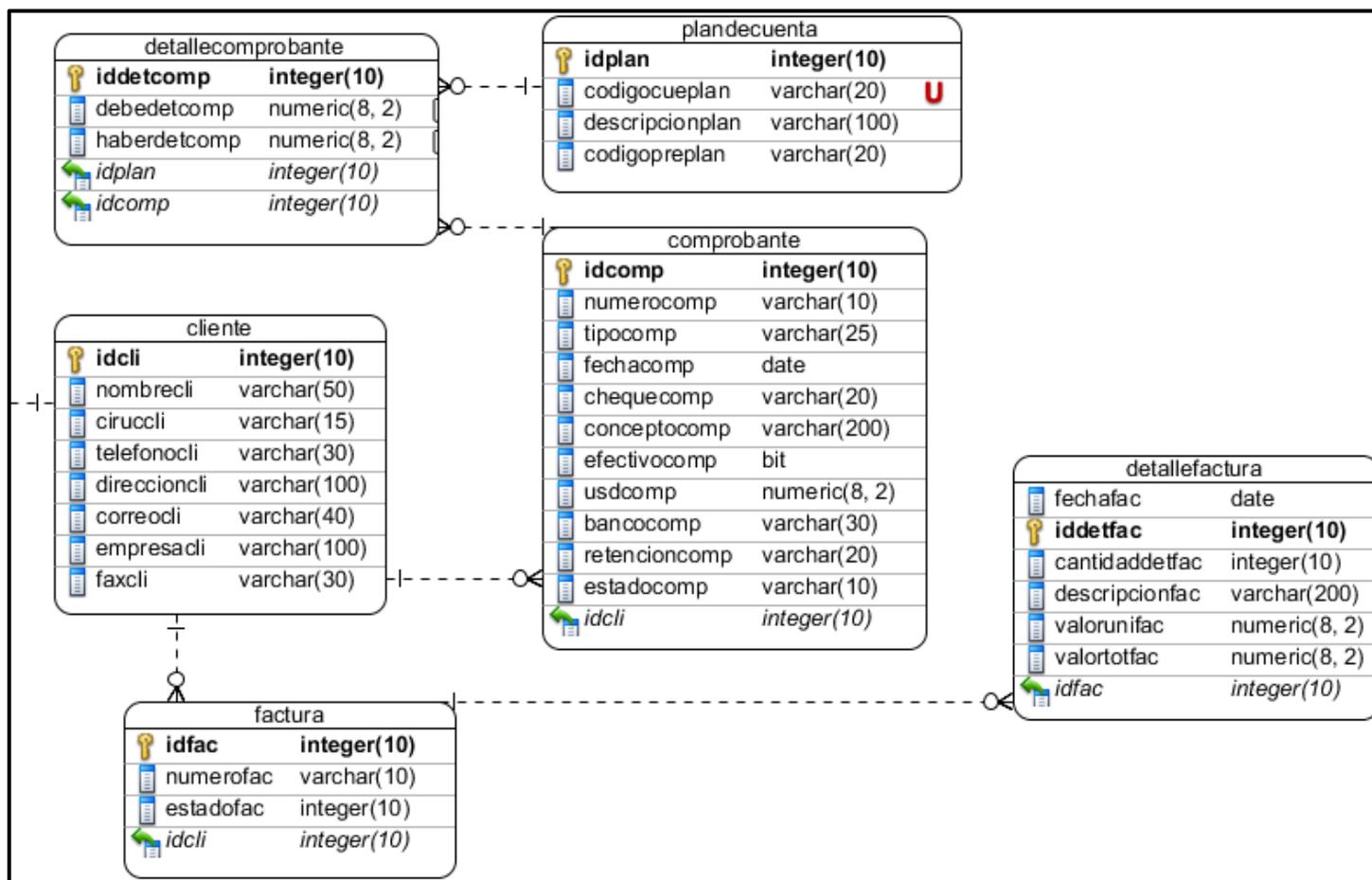


Figura 4.20: Diagrama Entidad Relación CTKWEB (Módulo Contable)

4.10 Diagrama Entidad – Relación CTKREPOSITORIO (Parte I)

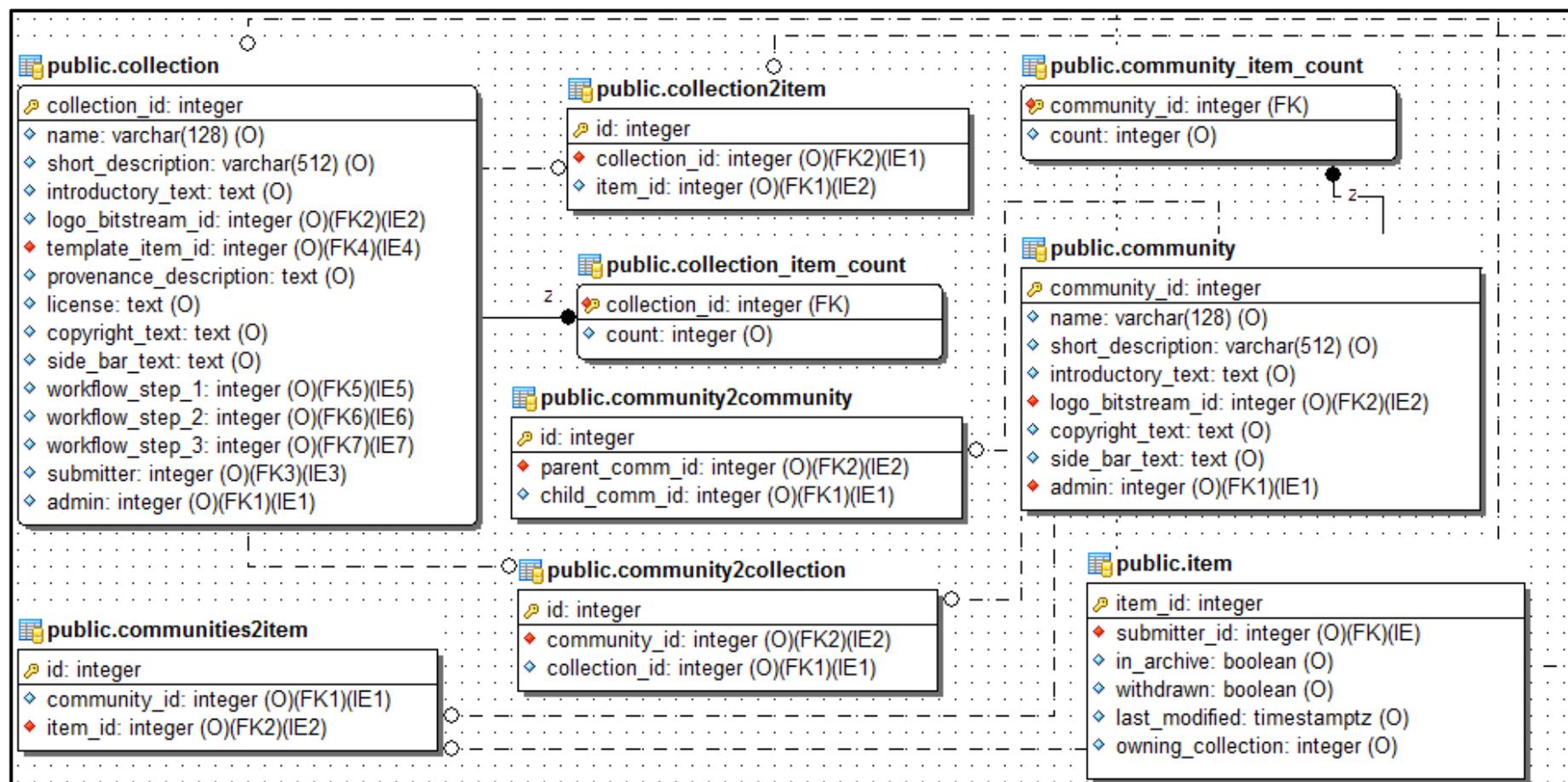


Figura 4.21: Diagrama Entidad Relación CTKREPOSITORIO (Parte I)

4.11 Diagrama Entidad – Relación CTKREPOSITORIO (Parte II)

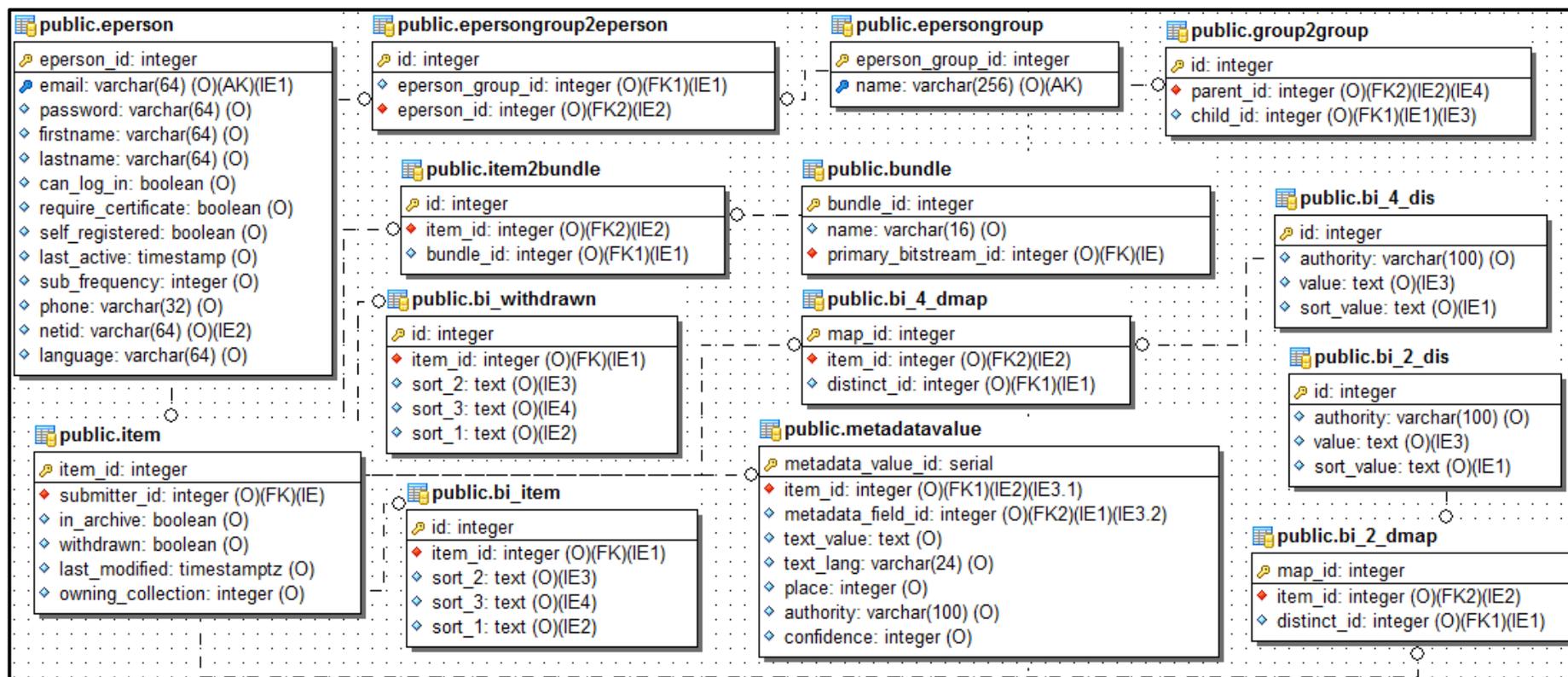


Figura 4.22: Diagrama Entidad Relación CTKREPOSITORIO (Parte II)

4.12 Diagrama Entidad – Relación CTKREPOSITORIO (Parte III)

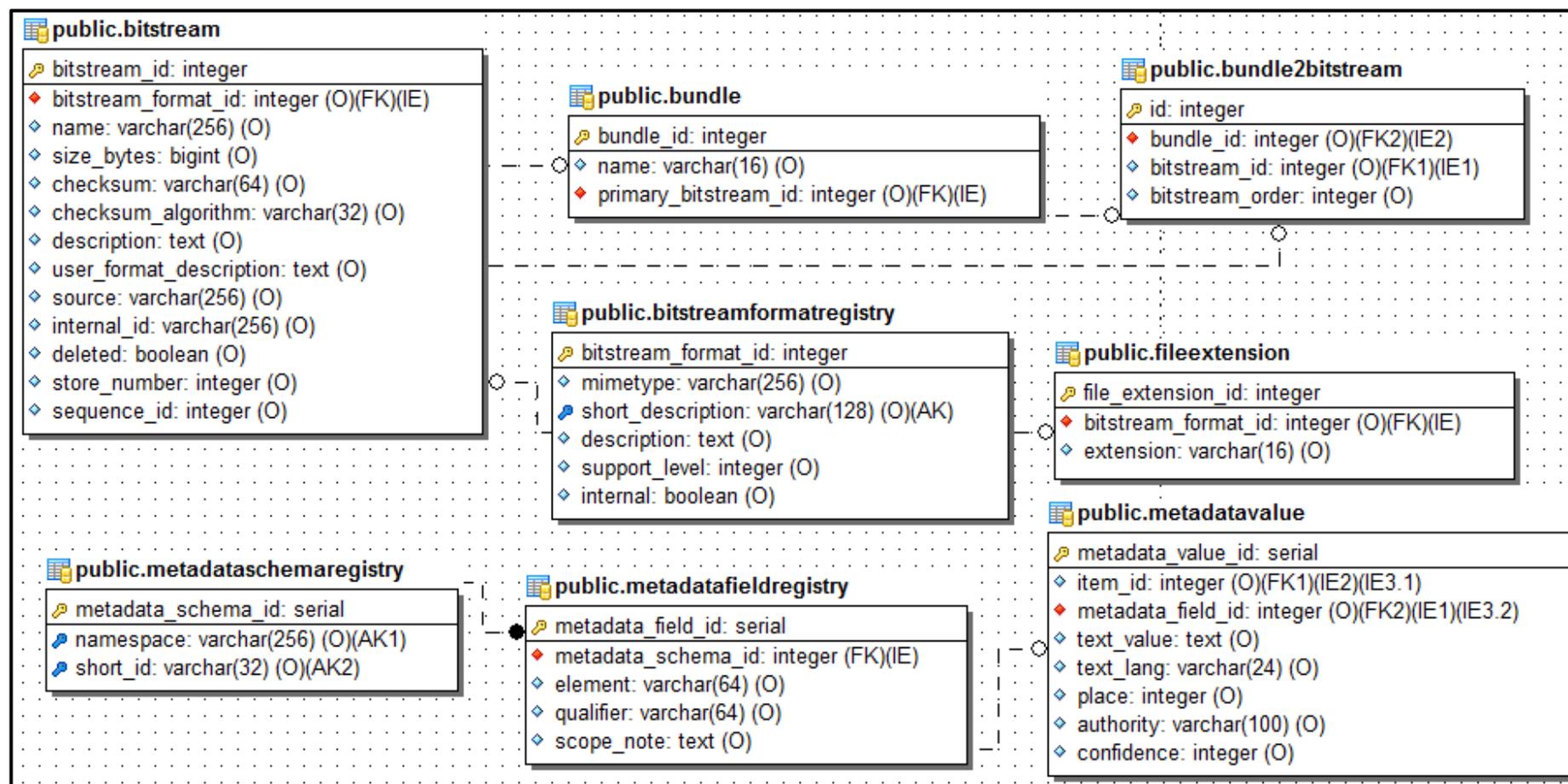


Figura 4.23: Diagrama Entidad Relación CTKREPOSITORIO (Parte III)

4.14 Diagrama de Clases CTKWEB – Módulo de Inventario (Parte II)

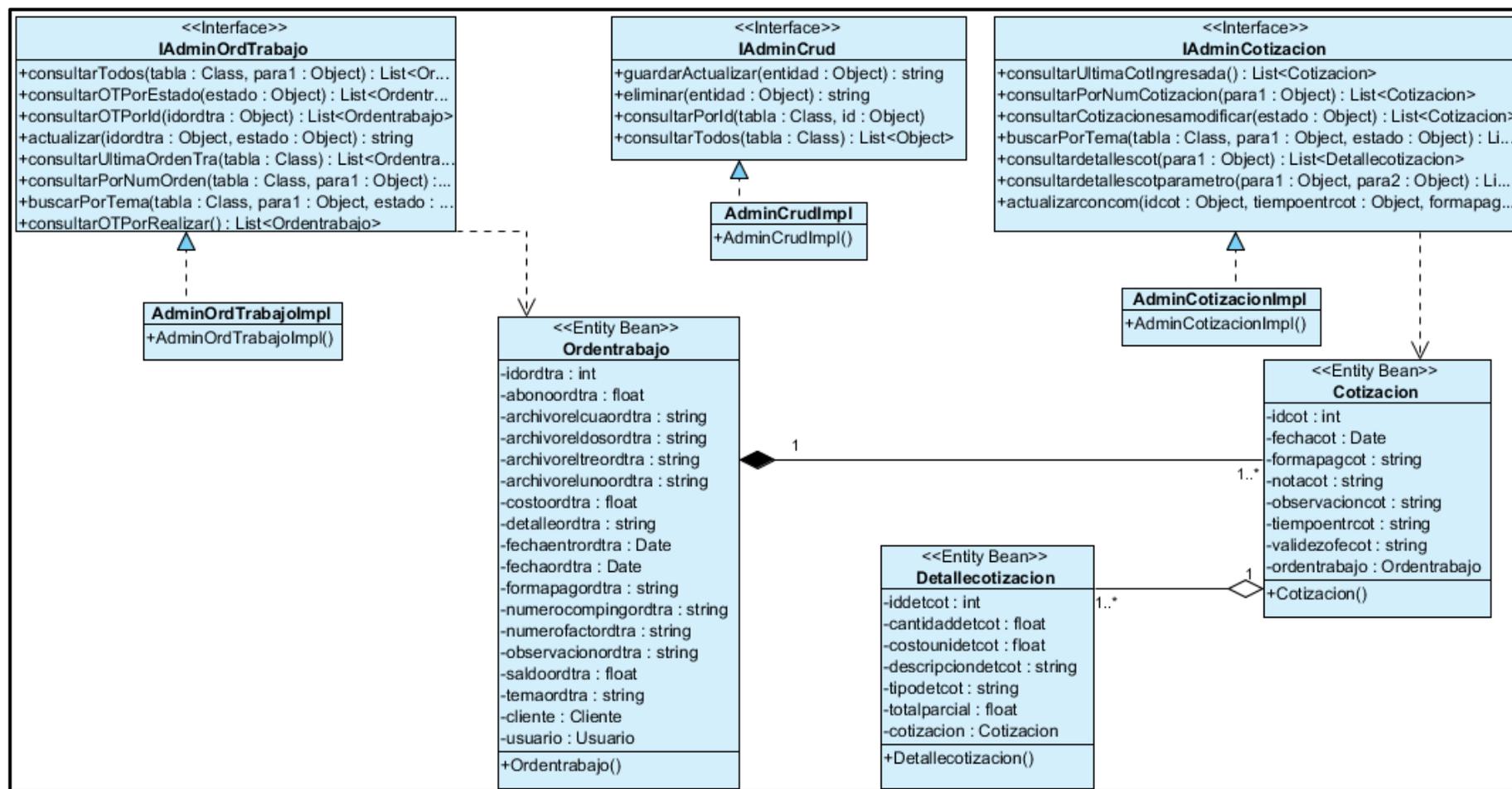


Figura 4.25: Diagrama de Clases CTKWEB – Módulo de Inventario (Parte II)

4.15 Diagrama de Clases CTKWEB – Módulo Contable

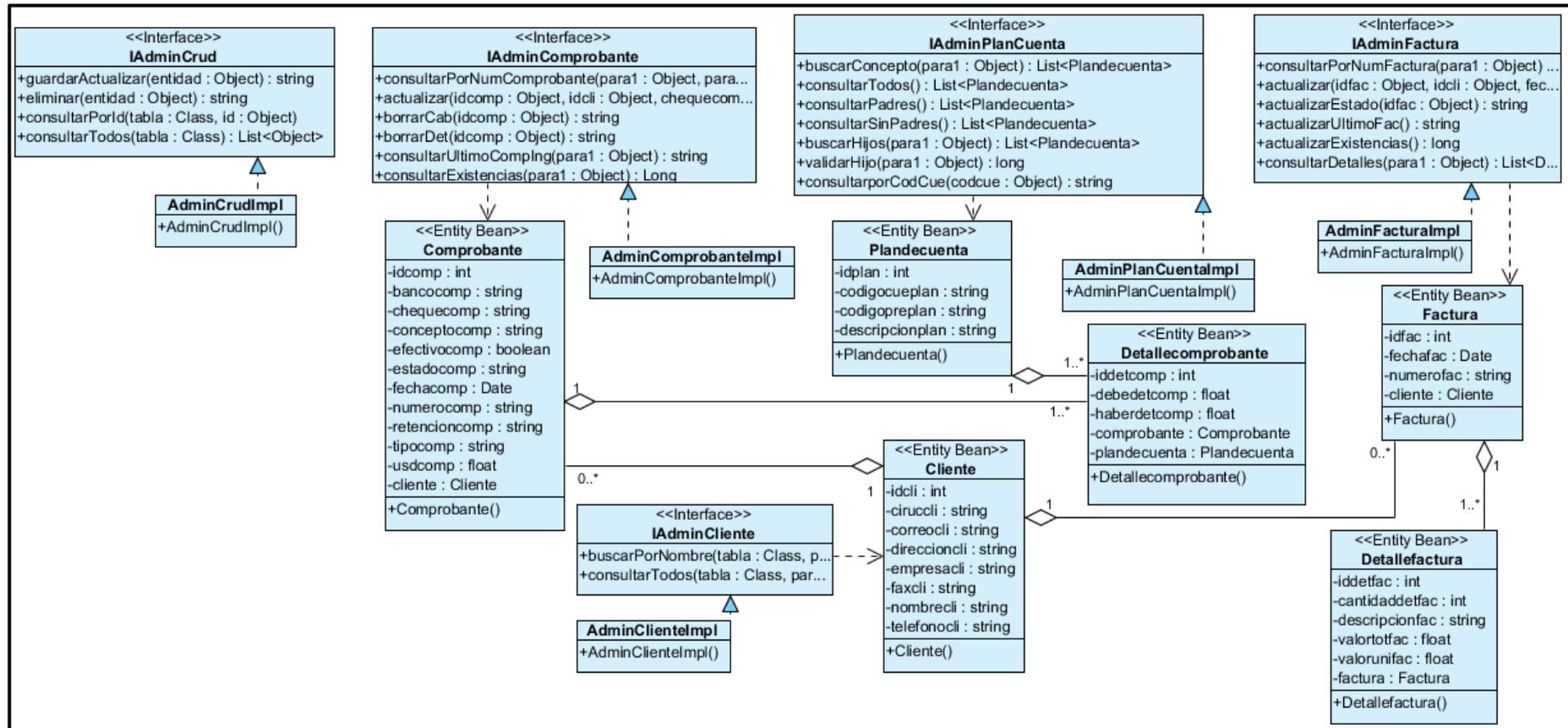


Figura 4.26: Diagrama de Clases CTKWEB – Módulo Contable

4.16 Diagrama de Clases CTKREPOSITORIO

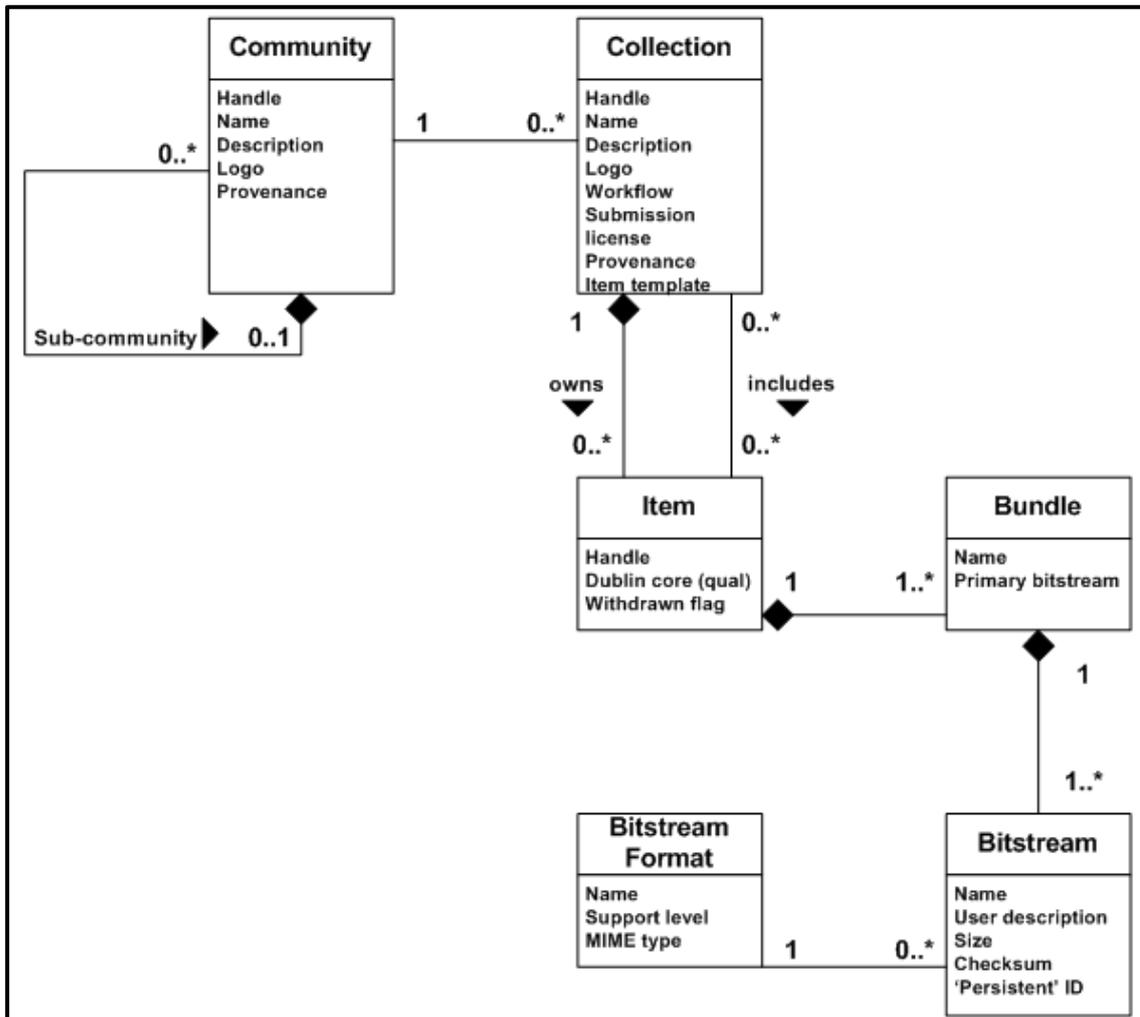


Figura 4.27: Diagrama de Clases CTKREPOSITORIO

ESPACIO EN BLANCO INTENCIONAL

4.17 Diagrama de flujo de trabajo – Inventario Administrar Proyectos

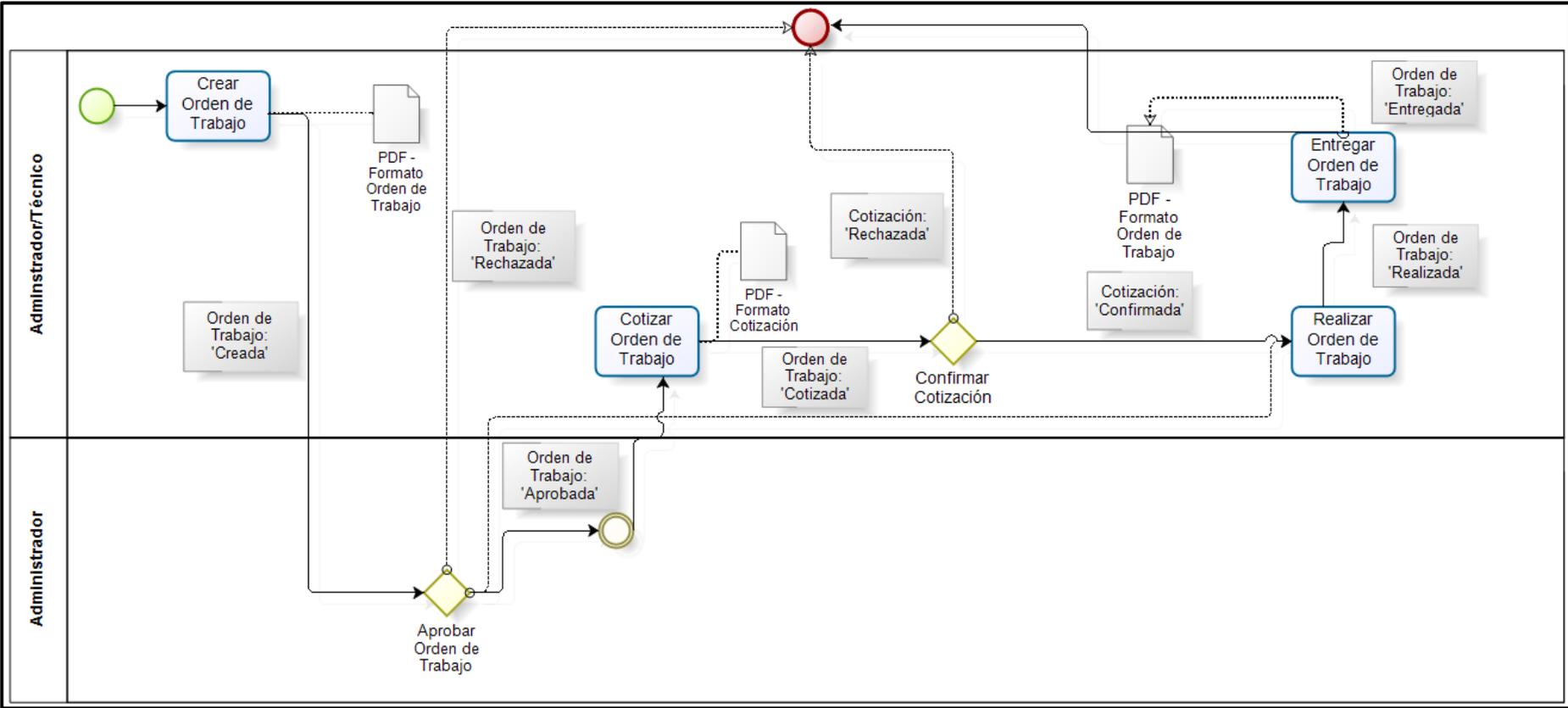


Figura 4.28: Diagrama de Flujo de Trabajo – Inventario Administrar Proyectos

4.18 Diagrama de Componentes

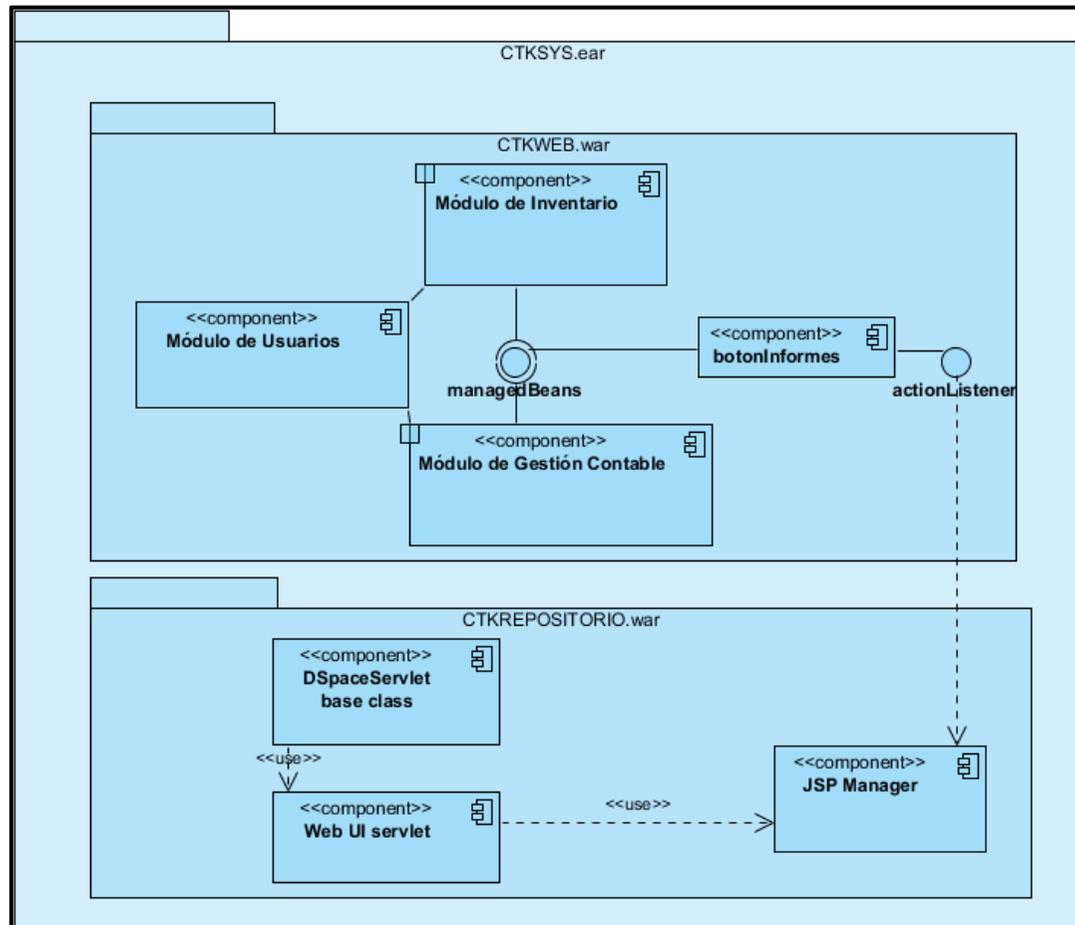


Figura 4.29: Diagrama de Componentes CTKSYS

4.19 Diagrama de Despliegue

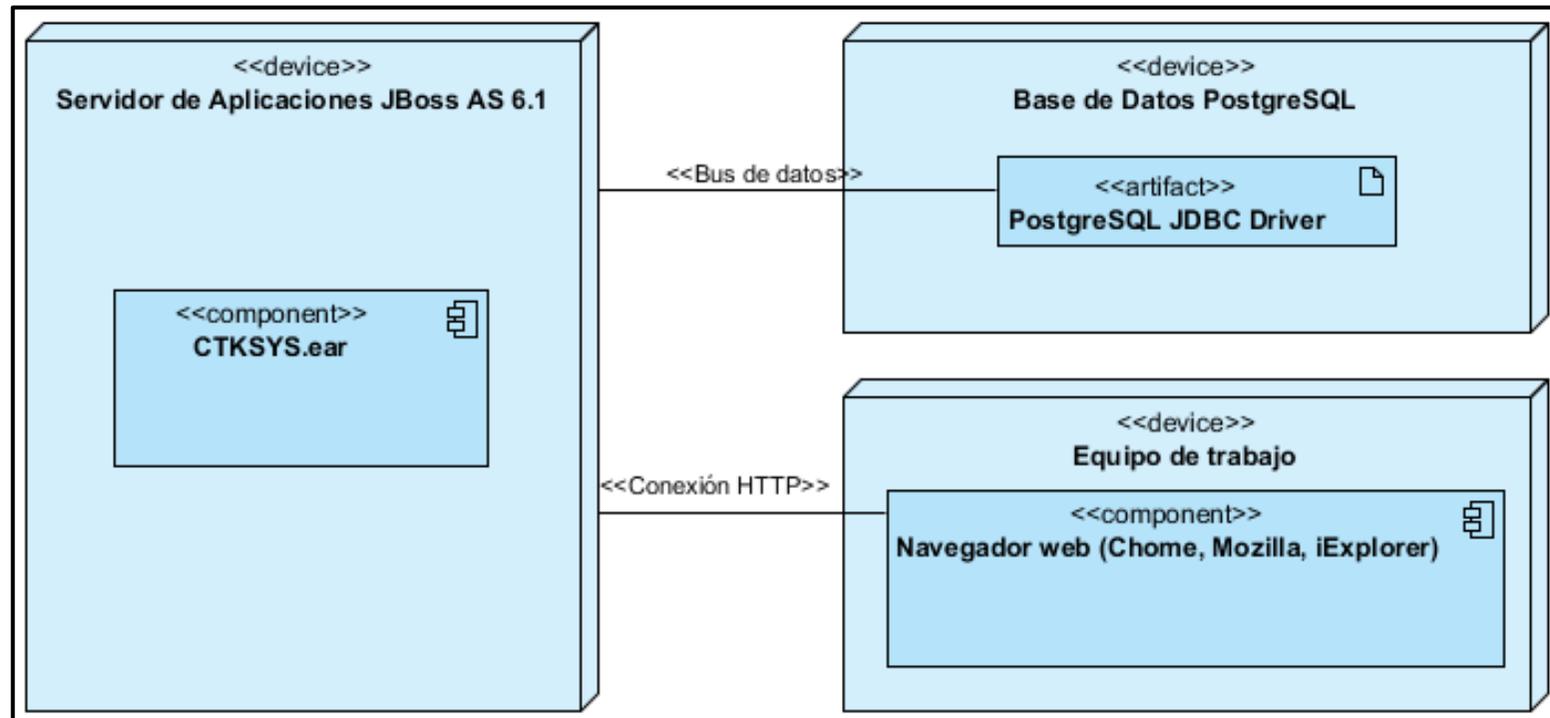


Figura 4.30: Diagrama de Despliegue CTKSYS

CAPÍTULO 5: DESARROLLO DEL SISTEMA

Tomando en cuenta lo definido en el capítulo 4, de forma general, el sistema desarrollado maneja el patrón de arquitectura MVC donde sus tres componentes son:

- El modelo, constituido por las entidades/datos que son persistentes en el sistema y se almacenan dentro de la base de datos.
- El controlador, encargado de obtener, procesar y enviar los datos al cliente.
- La vista, que muestra la información en el formato adecuado al cliente.

De manera más específica, la aplicación CTKSYS (EAR) está formada por dos módulos Web (WAR): CTKREPOSITORIO y CTKWEB que usa el módulo CTKEJB (JAR). De forma conjunta, estos módulos cuentan con tres niveles:

El nivel de Negocio, está constituido por dos capas: Lógica de Negocio y Acceso a Datos. En el módulo CTKWEB, estas capas han sido implementadas con componentes EJB (Enterprise Java Beans v3.0) que hacen uso de JPA (Java Persistence API v2.0) – Hibernate como mecanismos de mapeo Objeto Relacional para proveer de una capa de acceso a datos.

En cuanto al módulo web CTKREPOSITORIO, se hace uso de las APIs: DSpace Public API y Storage API propias de DSpace para las negociaciones entre la vista y el modelo, además de implementar el API JDBC (Java Database Connectivity API) para el acceso a datos.

Para el nivel de Presentación, en el módulo CTKWEB, se implementó la vista

basada en JSF (Java Server Faces v2.0) a través de la utilización del framework Primefaces.

En el módulo CTKREPOSITORIO se hace uso de la interface clásica de DSpace (JSPUI) implementada con JSP.

En el nivel de Datos para este sistema se utilizó PostgreSQL versión 8.4.

El modelo de arquitectura utilizado permitió implementar en su totalidad los requerimientos que exige el sistema y además, provee de una infraestructura escalable con un bajo nivel de dependencia entre cada uno de sus componentes.

Para el sistema se definió como patrón de diseño al Patrón de diseño Fachada el cual posee un diseño estructural, y resulta muy útil para sistemas en los que se quiere obtener un diseño amigable, pues lo que propone este patrón es una interfaz unificada; esto ayuda a simplificar la complejidad de un sistema, en especial cuando este posee subsistemas, como es el caso del presente sistema.

Este patrón resulta muy útil para el presente sistema principalmente porque es utilizado en el framework Primefaces, que es una de las herramientas fundamentales para la construcción de esta aplicación.

5.1 Estandarización de los elementos de desarrollo

Los estándares de programación ayudan a normalizar la estructura de codificación de variables, constantes, clases, tablas, vistas, etc.; lo que se consigue cuando una aplicación esta estandariza, es promover una escritura

clara y una mejor organización del código fuente, haciendo comprensible para cualquier programador cualquiera de sus clases.

5.1.1 Código de Aplicaciones

En la aplicación Web, existirán los siguientes tipos de objetos: Paquetes, Clases, Interfaces, Métodos, Variables.

5.1.1.1 Paquetes

El prefijo del nombre de un paquete será escrito con letras minúsculas, y será el nombre de dominio de alto nivel *com*.

Los subyacentes componentes del nombre del paquete variarán de acuerdo a las convenciones de nombres internas de la organización.

5.1.1.2 Clases

Los nombres de las clases deben ser sustantivos, cuando son compuestos tendrán la primera letra de cada palabra que lo forma en mayúscula. Si se tratase de una clase implementadora de una interface, se le asignará la palabra *Impl* al final del nombre de la clase. Si se trata de un managed bean, escribir la palabra *Bean* al final del nombre de la clase. Intentar mantener los nombres de las clases simples y descriptivas. Usar palabras completas, evitar acrónimos y abreviaturas. Ejemplos:

- class Cliente;
- class clienteBean;

```
1 package com.ctk.modelo;
2
3 import java.io.Serializable;
4
5
6
7
8 /**
9  * The persistent class for the cliente database table.
10 *
11 */
12 @Entity
13 @Table(name="cliente")
14 public class Cliente implements Serializable {
15     private static final long serialVersionUID = 1L;
16
17     @Id
18     @GeneratedValue(strategy=GenerationType.IDENTITY)
19     @Column(unique=true, nullable=false)
20     private Integer idcli;
21
22     @Column(length=15)
23     private String ciruccli;
24
25     @Column(length=40)
26     private String correcli;
27
28     @Column(length=50)
29     private String direccioncli;
30
```

Figura 5.1: Clase Cliente, captura desde IDE Eclipse

ESPACIO EN BLANCO INTENCIONAL

```
AdminCrudImpl.java
1 package com.ctk.negocio;
2 import java.util.List;
9
10 /**
11  * Esta clase representa la clase de negocio para las operaciones crud
12  */
13 @Stateless
14 public class AdminCrudImpl implements IAdminCrud {
15     // private EntityManagerFactory fabrica;
16     @PersistenceContext(unitName = "CTKSYSEJB")
17     private EntityManager controlador;
18
19     public AdminCrudImpl() {}
20
21     @Override
22     public String guardarActualizar(Object entidad) throws Exception {
23         controlador.merge(entidad);
24         return entidad.getClass().getSimpleName()
25             + " guardado correctamente";
26     }
27     @Override
28     public String eliminar(Object entidad) throws Exception {
29         controlador.remove(controlador.merge(entidad));
30         return "Eliminado Correctamente";
31     }
32     @SuppressWarnings("unchecked")
33     @Override
34     public Object consultarPorId(Class tabla, Object id) throws Exception {
35         return this.controlador.find(tabla, id);
36     }
37
38     @Override
39     public List<Object> consultarTodos(Class tabla) throws Exception {
40         Query consultaObjeto = controlador.createQuery("Select t from "
41             + tabla.getSimpleName() + " t");
42         return consultaObjeto.getResultList();
43     }
44 }
```

Figura 5.2: Clase AdminCrudImpl implementa la interface genérica IAdminCrud

ESPACIO EN BLANCO INTENCIONAL

```
1 package com.ctk.backingbean;
2
3 import javax.faces.application.FacesMessage;
15
16 /**
17  * clase bean para manejar entidad Cliente
18  *
19  * @author Diana Cruz, Christian Z.
20  *
21  */
22
23 @ManagedBean
24 @ViewScoped
25 public class ClienteBean {
26     // variable tipo Objeto
27     private Cliente cliente;
28     private Cliente clienteSel;
29     // listas
30     private List<Cliente> listaCliente;
31     private List<Object> listaCli;
32     // variables
33     int numCli;
34
35     // CAPA EJB Negocio
36     @EJB
37     private IAdminCrud controladorGeneral;
38
39     public ClienteBean() {
40         this.cliente = new Cliente();
41         this.clienteSel = new Cliente();
42         this.listaCliente = new ArrayList<Cliente>();
43         this.listaCli = new ArrayList<Object>();
44     }
45     //GETTER & SETTER
46     public Cliente getCliente() {}
49
50     public void setCliente(Cliente cliente) {}
53
```

Figura 5.3: Clase ClienteBean, captura desde IDE Eclipse

5.1.1.3 Interfaces

La nomenclatura es parecida a la de las clases pero antecedida por una I en mayúscula. Ejemplo:

- interface IAdminCrud;

```
IAdminCrud.java
1 package com.ctk.negocio;
2 import java.util.List;
3
4 /**
5  * Esta clase representa el contrato de negocio para las operaciones crud
6  */
7
8 @Local
9
10 public interface IAdminCrud {
11     /**
12      * Definición de método para guardar
13      * @param entidad
14      * @return
15      * @throws Exception
16      */
17     public String guardarActualizar(Object entidad) throws Exception;
18     /**
19      * Definición de método para eliminar
20      * @param tabla
21      * @param id
22      * @return
23      * @throws Exception
24      */
25     public String eliminar(Object entidad) throws Exception;
26     /**
27      * Definición de método para consultar por id
28      * @param tabla
29      * @param id
30      * @return
31      * @throws Exception
32      */
33     public Object consultarPorId(Class tabla, Object id) throws Exception;
34     /**
35      * Definición de método para consultar todos los campos de una tabla
36      * @param tabla
37      * @return
38      * @throws Exception
39      */
40     public List<Object> consultarTodos(Class tabla) throws Exception;
41 }
```

Figura 5.4: Interface genérica IAdminCrud, captura desde IDE Eclipse

5.1.1.4 Archivos de presentación

Estos archivos de extensión .xhtml generalmente constituyen la parte de exposición del sistema hacia el usuario; aquí se ve el uso del framework Primefaces a nivel de capa de presentación. En el presente proyecto se manejaron plantillas para mantener un estándar; se crearon 3 carpetas para cada módulo donde se encuentran las plantillas de: cabecera, lado izquierdo y plantilla, para cada uno de éstos. Ejemplo:

```

plantillaInventario.xhtml
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
2 <html
3   xmlns:h="http://java.sun.com/jsf/html"
4   xmlns:f="http://java.sun.com/jsf/core"
5   xmlns:ui="http://java.sun.com/jsf/facelets"
6   xmlns:p="http://primefaces.org/ui">
7 <f:view contentType="text/html">
8   <h:head>
9   <h:body>
10    <p:layout fullPage="true">
11      <p:layoutUnit id="top" position="north" size="50">
12        <ui:include src="cabecera.xhtml"></ui:include>
13      </p:layoutUnit>
14      <p:layoutUnit id="bottom" position="south" size="30"
15        collapsible="false" collapsed="false">
16        <p
17          style="font-size: small; margin-top: 0px; margin-bottom: -12px; text-align: center;">
18          • @rosstronik 2012 •</p>
19        </p:layoutUnit>
20      <p:layoutUnit id="left" position="west" size="100" resizable="true"
21        closable="false" collapsible="true" header="M E N Ú"
22        style="text-align:center" minSize="250">
23        <ui:include src="ladoizq.xhtml"></ui:include>
24      </p:layoutUnit>
25      <p:layoutUnit id="center" position="center">
26        <ui:insert name="content">
27          <span>Página en construcción</span>
28        </ui:insert>
29      </p:layoutUnit>
30      <ui:include src="../../sessionTimeOut.xhtml"></ui:include>
31    </p:layout>
32  </h:body>
33 </f:view>
34 </html>

```

Figura 5.5: plantillaInventario.xhtml, captura desde IDE Eclipse

Luego dichas plantillas son consumidas por un archivo.xhtml principal que es el que pertenece a cada una de las interfaces que se muestra en el sistema.

Ejemplo:

```

1 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
2 <html xmlns="http://www.w3.org/1999/xhtml"
3   xmlns:h="http://java.sun.com/jsf/html"
4   xmlns:f="http://java.sun.com/jsf/core"
5   xmlns:ui="http://java.sun.com/jsf/facelets"
6   xmlns:p="http://primefaces.org/ui">
7 <ui:composition
8   template="/templates/minventario/plantillaInventario.xhtml">
9   <ui:define name="head">: Crosstronik - Administracion Cliente ::</ui:define>
10  <ui:define name="content">
11    <h:form id="form">
12      <script type="text/javascript">
13      <script type="text/javascript">
14      <script type="text/javascript">
15    </div id="page">
16    <p:panel id="panelbor" rendered="true"
17      style="border:solid 10px #003db9;">
18    <p:growl id="growl" style="font-size: 24px;" showDetail="true"
19      sticky="false" />
20    <p:panel id="panel" header="Cliente"
21      style="margin-bottom:10px;position:relative;">
22      <p:focus context="panel" />
23      <p:fieldset
24        style="width:800px;height:170px;margin-left:40px;margin-top:10px;font-size: 13px;
25        text-align: center; font-family:Arial Narrow;">
26      <h:panelGrid columns="4" id="display2" cellpadding="2" style="margin-top:0px;">
27        <h:outputLabel value="Nombre/Contacto*:" style="font-size: 14px; text-align: justify;
28        color: #1555CB; margin-top: 5px;margin-left: 0px;font-family:Arial

```

Figura 5.6: admincliente.xhtml, captura desde IDE Eclipse

Para estos últimos archivos, el estándar que se maneja fue el siguiente:

- Para interfaces que alimentan propiamente al sistema de los datos básicos el nombre se forma del sufijo “admin” y el nombre de la clase que administran. Ejemplo: adminbodega.xhtml
- Para interfaces que controlan los procesos delicados del sistema, el nombre se forma de los sufijos “do”, más las tres primeras letras de la acción que realiza y las tres primeras letras del documento a nivel de la empresa en el que se las realiza. Ejemplo: doaprcot.xhtml se forma de los sufijos “do”, el sufijo “apr” de aprobación y “cot” de cotización.

5.1.1.5 Métodos

Los métodos deben ser verbos. Si son compuestos, tendrán la primera letra en minúscula; y la primera letra de las siguientes palabras que lo forman en mayúscula. Ejemplo:

- consultarItem();

5.1.1.6 Variables

Todas las instancias y variables de clase o método empezarán con letra minúscula. Si son palabras compuestas, tendrán la primera letra en mayúscula; y las siguientes palabras que lo forman en minúscula. Los nombres de variables no deben empezar con (“_”, ó, “\$”) aunque si sean permitidos por el lenguaje.

Se evita poner nombres de variables de un solo caracter. Ejemplo:

- `private String nombreProducto;`

5.1.2 Modelado de Datos

El modelado de datos se usa para escribir una base de datos de manera correcta. El modelo que se va a utilizar es el de Entidad – Relación.

Los nombres de las tablas en la base de datos deben estar escritos en minúsculas, si tienen dos palabras deben estar seguidas y en minúsculas.

Ejemplo:

- `producto`
- `detallemovimiento`

Los nombres reflejan a lo que hace alusión el contenido de la tabla, sólo se usan caracteres alfabéticos; no se utilizan dígitos numéricos, caracteres de puntuación o símbolos.

5.2 Módulos del Sistema

5.2.1 Gestión de perfiles de usuario

Este módulo permite crear, modificar, habilitar o deshabilitar usuarios. Además se asignarán permisos de manera que se de acceso a la funcionalidad del sistema únicamente a los usuarios que sean debidamente autorizados.

Repositorio digital de informes técnicos

Para la construcción de este módulo, se hace una adaptación del Repositorio Digital DSpace a los requerimientos de la empresa. Esta aplicación permite gestionar documentos (libros, informes, etc.) a través de la web. Por lo tanto, mediante este módulo, se gestiona la publicación/subida, búsqueda, y descarga de documentos –informes técnicos- que genera la empresa para alguno de sus clientes.

Mediante el uso del repositorio digital, un cliente que tenga acceso a este módulo podrá buscar o descargar los informes técnicos que han sido publicados por el administrador del sistema. El cliente hará uso de la información (semanal o mensual) que se publique en el sitio.

El administrador realizará la creación de usuarios, creación de grupos, asignación de privilegios además de la publicación/subida, búsqueda, descarga, y eliminación de archivos del repositorio.

5.2.2 Gestión de Inventario

Módulo que cubre tanto el control de las entradas y salidas de los productos pertenecientes a cualquier bodega de la empresa, como la automatización de los procesos para la generación de Órdenes de Trabajo y Cotizaciones (de materiales, y mano de obra).

Con la información ingresada a lo largo de éste módulo se generará: el formato de Orden de Trabajo (permitirá llevar un control de las órdenes de trabajo que están en trámite, desde su creación hasta la entrega del trabajo); y el formato de Cotización de Materiales y Mano de Obra (permitirá calcular el

costo por materiales, y mano de obra, de una Orden de Trabajo).

De manera integrada, la sección de Administración de Proyectos de este módulo de Inventario, trabaja con el Módulo de Gestión Contable, en el momento en el cual se registra uno o más anticipos de pagos por la ejecución de una Orden de Trabajo. Para esto, se realiza en la sección Administración de Comprobantes del módulo contable, el registro del o los comprobantes de ingreso que sean necesarios.

De igual manera, se utiliza la sección de Facturación del módulo Contable para registrar la factura que se genera por el pago total del trabajo realizado

5.2.3 Gestión Contable

En este módulo se gestionan los procesos de apoyo contable, siendo principalmente: la generación de comprobantes de ingreso, comprobantes de egreso, facturas, y reportes contables (Libro Diario, Libro Mayor, Balance de Comprobación, Estado de Resultados).

CAPÍTULO 6: PRUEBAS

En este capítulo se presenta todo lo referente a pruebas, según la metodología Extreme Programming.

Las pruebas en X.P., constituyen una de sus prácticas fundamentales. Esta actividad debe realizarse en forma continua a lo largo del proyecto. La metodología define la utilización de dos tipos de pruebas: Pruebas Unitarias y Pruebas de Aceptación. Sin embargo, para probar este sistema, también se hizo uso de Pruebas de Integración, como se muestra en la siguiente figura:

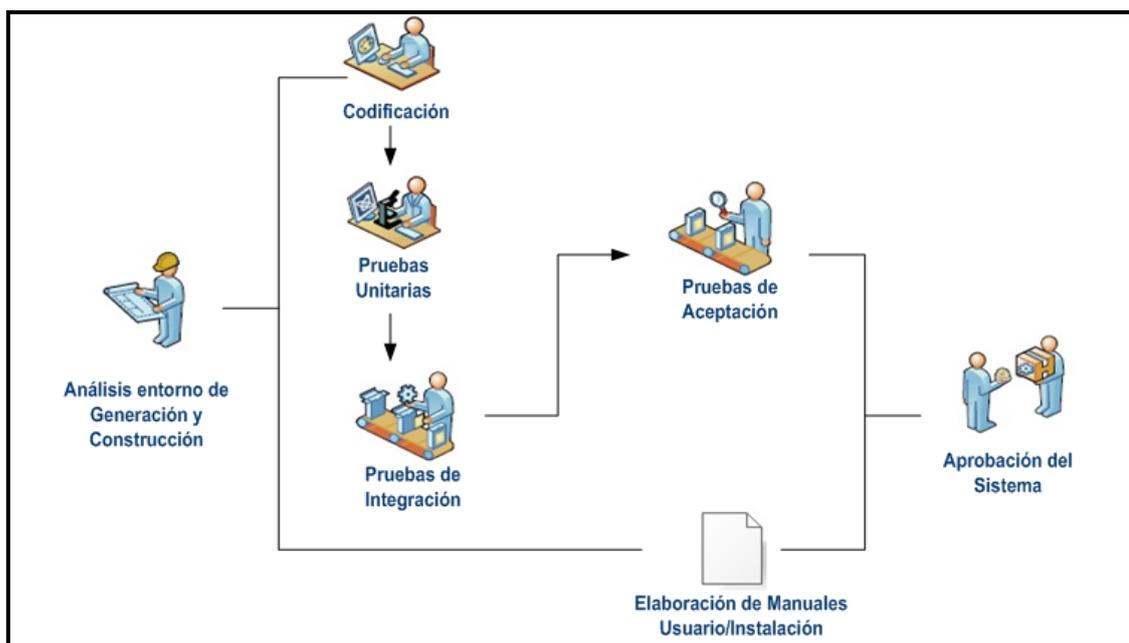


Figura 6.1: Ciclo de Pruebas²¹

²¹ Figura 1. Ciclo de vida de Prueba. Adaptado de "Máster en Gestión y Diseños de Servicios y Sistemas de Información", por Jiménez A. 2010 p. 5

6.1 Pruebas de Unidad

Las pruebas unitarias o de unidad son aquellas que verifican un módulo del sistema. El objetivo principal de la realización de estas pruebas es asegurar que un módulo determinado del sistema cumpla con un comportamiento esperado en forma aislada antes de ser integrado al sistema.

Este tipo de pruebas deben ser definidas por el desarrollador antes de comenzar a escribir el código.

Luego de escribir el código, los programadores ejecutan las pruebas, las cuales deben resultar 100% efectivas, caso contrario, el código no puede integrarse al resto del sistema. Por lo que, hay que solucionar los errores de manera completa y volver a ejecutar los casos de prueba²² hasta conseguir que no se produzcan más errores.

El no realizar pruebas unitarias conlleva a tener que invertir gran cantidad de horas en sesiones de debugging al momento de integrar el código con el sistema existente.

Por otro lado, la realización de pruebas unitarias, implican retroalimentación en la realización del desarrollo, porque permiten al programador saber si una determinada funcionalidad se puede agregar al sistema existente sin alterar su funcionamiento.

²² Los casos de Prueba son la descripción de las pruebas a realizar previamente a realizarlas.

Durante el desarrollo de este sistema, se realizaron estas pruebas de forma continua cuando: la interfaz de un método no era clara, la implementación era complicada, para testear entradas y condiciones inusuales, etc. Se contempló cada módulo del sistema que podría tener fallas. Para estas pruebas, no fue necesario utilizar ningún artefacto Test Case.

6.2 Pruebas de Integración

Este tipo de pruebas se realizan una vez que se finalizan las pruebas Unitarias, con éstas se busca asegurar que el sistema y los subsistemas que componen las piezas individuales del software funcionen correctamente al interoperar en conjunto.

Para este sistema, y debido a que tiene como base una metodología ágil se hace uso de las pruebas de integración evolutiva, cuya principal ideología se fundamenta en lo siguiente:

“Siempre hay cambios de última hora. El objetivo es poderlos tratar, son necesarios. Si como probador no los puedes manejar, tienes un problema, y la solución no está ni en quejarse ni en tratar de pararlos”²³ Malfará (2011)

Lo que se logra con la integración evolutiva es la entrega de una pieza pequeña de software (prototipo) como si fuera el sistema. Después se añaden otras piezas que aportan más funcionalidad y se vuelve a probar todo como un conjunto.

²³ Adaptado de “*Testing en eXtreme Programming*”, Disertación Gestión de Software, por Malfará, D., Cukerman, D., Cócara, F., Cassinelli J.P., Séttimo, R.

Con la utilización de este tipo de pruebas se hace uso, también, del método de verificación por revisión, Prototipaje. Es un método para revisar que el análisis y diseño responde a las expectativas del cliente; el cual consiste en construir versiones incompletas del sistema que solo cumplen con un cierto número de funcionalidades, o solo con las principales pantallas.

La utilización de prototipos fue ventajosa ya que permite:

- Obtener la validación del análisis por parte del usuario.
- Aclarar expectativas
- Limitar riesgos
- Estimar mejor el proyecto
- Comprobar la viabilidad de ciertas funcionalidades.

Para este tipo de pruebas, no fue demandante la utilización de un artefacto Test Case, como en el caso de las pruebas de aceptación.

6.3 Pruebas de Aceptación

En X.P. las pruebas de aceptación deben consistir en un conjunto mínimo (no pobre ni insuficiente) de casos de prueba que cubran los requerimientos de negocios fundamentales planteados por el cliente.

Acorde a lo que plantea la metodología, las pruebas de aceptación son responsabilidad del cliente; pues deben reflejar los requerimientos y funcionalidades que ellos quieren obtener.

Sin embargo, para este proyecto, resultó imposible que sea el cliente quien escriba las pruebas de aceptación, porque como cliente no tiene los conocimientos técnicos ni la experiencia necesarios para escribirlos correctamente. Por esta razón, los dos integrantes de este equipo de desarrollo, asumieron los roles de Analista – Desarrollador y Verificador/Tester – Desarrollador.

Siendo el Tester quien sostuvo reuniones con el cliente, de manera que pueda interpretar sus ideas y luego poder escribir los casos de prueba correspondientes, utilizando el formato de Casos de Prueba de aceptación que se muestra en la siguiente figura:

Fecha Realización	<i>[ingresar fecha ejecución del test case]</i>				
Título del Caso de Prueba	<i>[ingresar el título del test case]</i>				
Núm. Caso de Prueba	<i>[ingresar el número del testcase, el mismo que ayudará para llevar un mejor control]</i>				
Objetivo	<i>[ingresar el objetivo que se pretende alcanzar con la prueba]</i>				
Resultado Esperado	<i>[ingresar a manejar general los resultados que se esperan de la prueba]</i>				
Paso	Procedimiento	Resultado Observado	Resultado de la Prueba		
			Pasa	Falla	Comentarios
1	<i>[se detalla los pasos a seguir]</i>	<i>[se detalla el resultado que se espera obtener de la configuración o acción realizada, es decir, como debería funcionar]</i>	<i>[se marca si el resultado fue igual al esperado]</i>	<i>[se marca si el resultado NO es igual al esperado]</i>	<i>[Generalmente se ocupa este campo para describir el error que se ha obtenido, o algún comentario válido]</i>
Start Testing					
2					
3					
** End of Test Case **					

Figura 6.2: Formato Test Case

Se pueden apreciar los Casos de Prueba realizados en la empresa, en el anexo F

El Caso de Prueba a formular debe indicar:

- Qué se prueba
- Cómo se prueba
- Quién lo prueba
- Qué hace con el sistema
- Cómo debería responder el sistema

Puntos relevantes que se tomaron en cuenta para la definición de los casos de prueba:

- Describir anticipadamente la salida esperada.
- Inspeccionar los resultados de cada prueba. También se cometen errores al describir y ejecutar las pruebas.
- Se deben cubrir las condiciones inválidas e inesperadas.
- Los casos de prueba no son de uso y deshecho, se pueden ejecutar tantas veces como haga falta.
- No estimar el tiempo para pruebas asumiendo que no se encontrarán errores, sino al contrario, lo positivo al probar es que los errores sean encontrados.

Finalmente, y para dar validez a las pruebas realizadas; se pone en práctica la concepción que hizo Kent Beck (autor de la metodología X.P.) quien recomienda la exhibición de los resultados que se obtienen al ejecutar las pruebas de aceptación, a través de la generación de reportes y gráficas que desplieguen los porcentajes de efectividad obtenidos. Estos índices evalúan si

el equipo de desarrollo está realizando un buen trabajo o no.

Para el presente sistema, se muestran los resultados obtenidos, en la Tabla 6.1, y en las figuras: Figura 6.3, Figura 6.4, Figura 6.5.

Tabla 6.1: Resultado de Pruebas

Usuario	PARAMETROS				
	Adaptación al Sistema (%)	Gasto Papel en hojas		Tiempo ejecución en minutos	
		Antes del Sistema	Con el sistema	Antes del Sistema	Con el sistema
Módulo Usuarios		Por operación		Por actividad	
Administrador	90	1-5	0	N/A	N/A
Módulo Inventario		Por Orden de Trabajo		Por actividad	
Bodeguero 1	80	20-30	1-10	30'-45'	5'-7'
Bodeguero 2	80			20'-30'	
Técnico 1	80	30-40	1-10	20'-30'	5'-8'
Técnico 2	75			15'-25'	
Módulo G. Contable		Por Comprobante y Factura		Por actividad	
Contador 1	75	10-13	3-5	9'-15'	3'-5'
Contador 2	90			6'-10'	
Módulo Repositorio		Por Mes		Por actividad	
Personal CTK	80	4	0	5'-10'	2'-3'
Cliente Externo	90	4	0	2'-5'	1'-2'

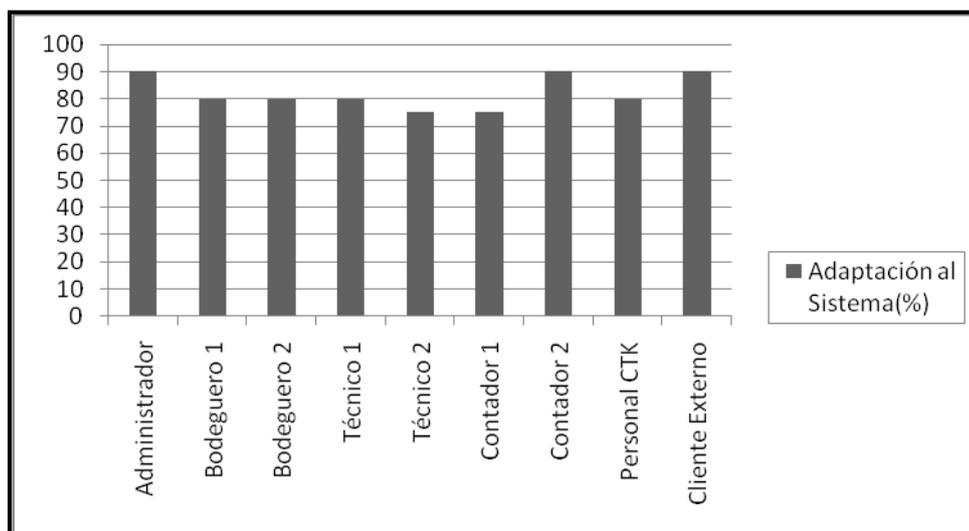


Figura 6.3: Porcentaje de adaptación al sistema CTKSYS

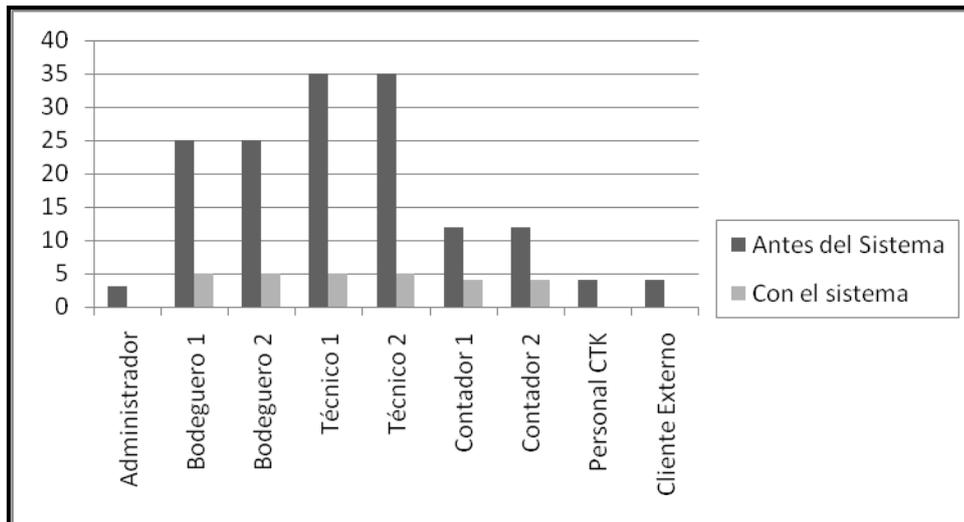


Figura 6.4: Gasto de papel en hojas

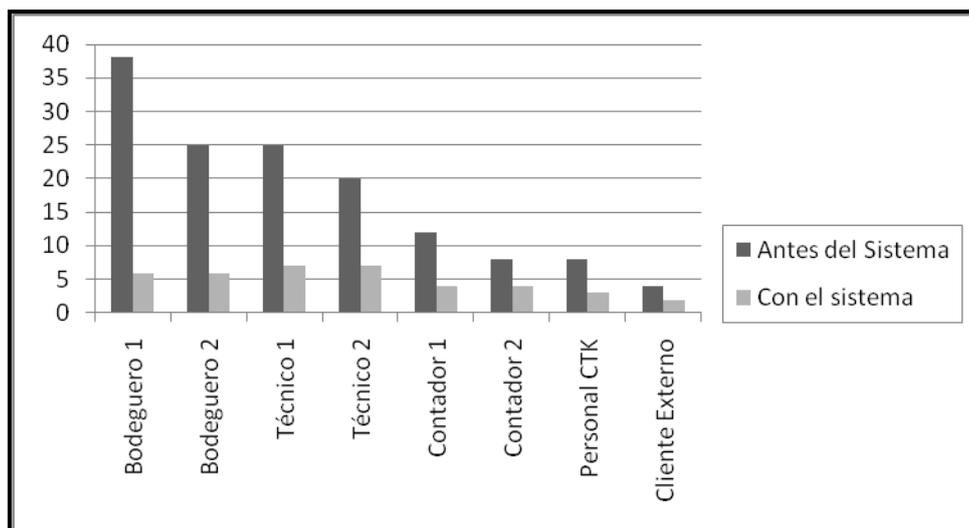


Figura 6.5: Tiempo ejecución de actividades en minutos

De la Figura 6.3 se puede determinar que el porcentaje de adaptabilidad del personal de la empresa al sistema, es bastante alto, por su amigable diseño, y fácil manejo.

Como se muestra en la Figura 6.4, se evita la pérdida de tiempo en la generación, emisión, recepción y control de archivos, documentos o informes técnicos por medio de correo electrónico, o impresos.

En la Figura 6.5 se puede observar que hay una clara disminución en el tiempo de ejecución de las actividades que antes realizaba manualmente el personal técnico y el contador de la empresa, principalmente en lo referente a la administración de Órdenes de Trabajo, Cotizaciones, Inventario, generación de comprobantes y facturas. Con el uso adecuado de la herramienta informática desarrollada se optimiza el manejo de un recurso importante como es el tiempo, permitiendo canalizarlo adecuadamente con el fin de lograr una mejor atención a sus clientes.

CAPÍTULO 7: CONCLUSIONES Y RECOMENDACIONES

7.1 Conclusiones

- El sistema desarrollado para la empresa Crosstronik Cía. Ltda., utilizando herramientas de software libre permite que dicha empresa cuente con una herramienta informática a la medida, que automatiza los procesos internos de la empresa y que no genera costos por licencias.
- La arquitectura utilizada para la construcción del sistema, es una infraestructura escalable con un bajo nivel de dependencia entre sus componentes, por lo tanto, se podría optimizar sus actuales servicios y añadir nuevos.
- El uso de metodologías ágiles ha generado un punto de inflexión donde cada día las empresas requieren software flexible, que se adapte a los cambios en el menor tiempo posible y se evidencie avances por el concepto de prototipado.
- El diseño de una interfaz clara, consistente y estandarizada, facilita el uso y capacitación del personal de la empresa, que poco o nada estaba familiarizado con la utilización de algún sistema informático.
- Al generar una aplicación con software libre, se debe evaluar el soporte técnico y la curva de aprendizaje que avala dicha herramienta, dado que al escalar a futuro en funcionalidad puede resultar más costosa que una herramienta propietaria.

7.2 Recomendaciones

- Se debe probar si lo desarrollado es lo deseado, realizando una retroalimentación cuidadosa con el cliente, que es quien va a hacer uso del sistema y quien mejor conoce los procesos de su negocio.
- Desarrollar aplicaciones escalables, y de arquitecturas robustas, que permitan una fácil integración con las nuevas tecnologías, y que dicha integración no requiera de una gran cantidad de recursos.
- Utilizar la metodología ágil XP dependerá del tipo de sistema que se tiene que desarrollar. Un manejo correcto de esta metodología, permitirá tanto al cliente como al desarrollador mejorar el producto en cada prototipo y adaptarse de mejor manera a los cambios que ocurren en el desarrollo del sistema.
- Al definir las interfaces gráficas es necesario seguir un estándar que sirva como guía para facilitar la elaboración de estas y que resulten de fácil manejo para el usuario.
- Analizar minuciosamente todas las herramientas a usar en el desarrollo de un sistema, sobre todo si éstas son de software libre, dado que a la hora de personalizarlas, puede representar una elevada curva de aprendizaje, por lo que se aconseja tener conocimientos básicos las herramientas a utilizar.

BIBLIOGRAFÍA

- Beck, K. (1999). *Extreme Programming Explained: Embrace Change*. Boston.
- Beck, K. (2001, Febrero 15). *Principios del manifiesto ágil*. Retrieved from <http://agilemanifesto.org/iso/es/principles.html>
- Calserón, C. (2012, Mayo 20). *Desarrollo rápido de aplicaciones empresariales con Java EE*. Retrieved from Ubuntu One: <http://ubuntuone.com/7OjbuiEbcwLuXbfLNDH6OK>
- Canós, J. P. (2012, Marzo 2012). *Metodologías Ágiles en el Desarrollo de Software*. Retrieved from Willy Dev: <http://willydev.net/descargas/prev/TodoAgil.pdf>
- Chazarra J., R. V. (2012, Marzo 10). *Desarrollo de un repositorio de objetos de aprendizaje usando DSpace*. Retrieved from E-Prints Complutense: <http://eprints.ucm.es/11078/1/MemoriaSI.pdf>
- Daum, B. (2005, Junio 19). *Professional Eclipse for Java Developers*. Indianápolis, Indiana, Estados Unidos.
- Garzón, A. (2011, Noviembre 23). *Un acercamiento a hibernate spatial*. Retrieved from ICDE: http://www.icde.org.co/web/ide_gig/blogs/-/blogs/un-acercamiento-a-hibernate-spatial
- Group, T. P. (2009, Mayo 14). *PostgreSQL 8.4 Documentation*. Estados Unidos.
- Malfará, D. C. (2010). *Testing en extreme programming*. *Extreme Programming*, (p. 75). Montevideo.
- Marchioni, F. (2009, Julio 23). *JBoss AS 5 Development*. Toronto, Ontario, Canadá.

Peralta, I. (2011, Agosto 20). *Historias de Usuario*. Retrieved from User Agile Development: <http://useragiledevelopment.blogspot.com/2011/05/user-stories.html>

Richardson, C. A. (2007, Septiembre 22). Professional Java JDK. Indianápolis, Indiana, Estados Unidos.

Sommerville, I. (2006, Enero 23). Ingeniería del Software. Madrid, Madrid, España.

Villena, A. (2009, Abril 16). *Introducción a los métodos ágiles*. Retrieved from Slideshare Present Yourself: <http://www.slideshare.net/chileagil/introduccion-gil-a-extreme-programming-webpreendedor08-350127>

HOJA DE LEGALIZACIÓN DE FIRMAS

ELABORADO POR

Diana Catalina Cruz Caiza

Christian Xavier Zhamungui Oviedo

DIRECTOR DE LA CARRERA

Ing. Mauricio Campaña

Sangolquí, Junio del 2013

BIOGRAFÍA

Christian Xavier Zhamungui Oviedo

Nació en la ciudad de Quito, el 13 de junio de 1988; sus padres son el Sr. José Zhamungui y la Sra. Carmen Oviedo. Es el primero de tres hermanos, Realizó sus estudios primarios en la Unidad Educativa “Abdón Calderón” donde inculcaron en él valores como el respeto, la camaradería y responsabilidad; luego continuó sus estudios secundarios en el COMIL N° 10 “Abdón Calderón” donde logró de Cuarto a Sexto Curso la distinción de Comandante de Curso, por su buen rendimiento académico; además en Sexto Curso formó parte del Pelotón Comando de su Colegio, Luego se incorporó como bachiller en la especialidad de Informática; su amor por la tecnología y la evolución de los sistemas de información le hizo inclinarse por la carrera de Ingeniería en Sistema e Informática en la Escuela Politécnica del Ejército, complementando su formación con la aprobación de los 4 módulos de la certificación internacional CCNA, además de la obtención de una suficiencia en el idioma Inglés.

BIOGRAFÍA

Diana, nació el día Jueves 28 de julio de 1988. Aprendió sus primeras letras en la Escuela Fiscal Abelardo Flores. Sus estudios secundarios los realizó en la Unidad Educativa San Francisco de Sales, mientras que los Universitarios los finalizó en la Escuela Politécnica del Ejército -ESPE-.

Además ha realizado cursos que han aportado a su formación profesional como: la aprobación de los 4 módulos de la certificación internacional CCNA, Java Advanced Developer, y la obtención de la suficiencia en el idioma Inglés.