

00321
78



**UNIVERSIDAD NACIONAL AUTONOMA
DE MÉXICO**

I-A

FACULTAD DE CIENCIAS

**"PROGRAMACIÓN EXTREMA Y UML PARA EL
DESARROLLO DE UN SISTEMA EN EL PROCESO
DE SURTIDO EN UN ALMACEN EDITORIAL"**

TESIS

QUE PARA OBTENER EL TITULO DE

ACTUARIO

PRESENTA

CARLOS ALBERTO RAMÍREZ LUNA

DIRECTOR DE TESIS

M. EN C. GUADALUPE IBARRUENGÓN GONZALEZ



México, D.F. 2003
FACULTAD DE CIENCIAS
SECRETARÍA ESCOLAR

**TESIS CON
FALLA DE ORIGEN**



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

PAGINACION DISCONTINUA

H



DRA. MARÍA DE LOURDES ESTEVA PERALTA
Jefa de la División de Estudios Profesionales de la
Facultad de Ciencias
Presente

Comunicamos a usted que hemos revisado el trabajo escrito: "Programación Extrema y UML para el desarrollo de un sistema en el proceso de surtido en un almacén editorial."

realizado por Carlos Alberto Ramirez Luna

con número de cuenta 9200045-6 , quien cubrió los créditos de la carrera de:

Actuaria

Dicho trabajo cuenta con nuestro voto aprobatorio.

A t e n t a m e n t e

Director de Tesis
Propietario

M en C Guadalupe E Ibarquengoitia González

Gu E Ibarquengoitia

Propietario

M A Sergio I Escamilla Sanchez

[Signature]

Propietario

Dra. Amparo Lopez Gaona

[Signature]

Suplente

M en C. Jorge Luis Ortega Arjona

[Signature]

Suplente

Mat. Victor Hugo Dorantes González

[Signature]

Consejo Departamental de Matemáticas



M. en C José Antonio Flores

ESCUELA DE CIENCIAS
CONSEJO DEPARTAMENTAL
DE
MATEMÁTICAS

Solamente quienes toman sosegadamente aquello por lo cual se atarea la gente del mundo pueden atarearse por aquello que la gente del mundo toma sosegadamente.

Chang Ch'ao.

Agradecimientos

© A mi hermana Ximena por las diferencias que nos hacen tan iguales, a mi mamá por lo que hemos vivido juntos y lo que nos espera, a mi abuelita por su paciencia y su mirada delante y detrás de mí, a mi Tillén por el tiempo y emociones en común, al gordo por su forma de ser y su cariño incondicional, a mi abuelo porque alguna vez anduvo por acá y nos caíamos bien, a mis lunas chiquitas: Andrés, Alexis Daniela y María, a mi tía Marta por sus palabras y su emoción por lo vivo, a mi tío Manolo por sus alegres y frecuentes visitas y al resto de mi familia que no aparece en estas líneas pero que siempre llevo conmigo.

∞ A Galo por las filosofías y mundos contruidos, a Iván por las reuniones e ilusiones compartidas, a Oliver y Arais por los tiempos lúdicos y tanta cosa que hemos vivido, a Jorge por las canciones y las confesiones, al trasatlántico Mameel por los hechos insólitos de que hemos sido testigos, a Anabel por las latitudes siguiéndonos los pasos, a Yarely por los días cercanos y lejanos.

§ A todos en Consucorp: Themis, Sergio, Andrés, Richard, Samara, Silvia y José Luis, es un placer convivir con ustedes. Al Club DBA: Araceli, Feli, Richard, Rose, Rubén y Ricardo por aparecer súbitamente y compartir imágenes y sueños. A Susy y Silvia en Santillana por el tiempo laboral y extra laboral. A la familia Sánchez Sandoval y Zepeda Pozos por tanto tiempo de caernos bien.

⊕ A todos aquellos con los que alguna vez compartí camino y que por razones diversas ya no se encuentran tan cerca: Elena Aguilera, que pensó y apoyó este proyecto antes de existir, Martín Cadena, Alejandra Bonabel, Juan Carlos Mora, Abigail Vega, a la familia Ortíz Arana, al buen Felipe, Federico Sánchez Bringas, Eduardo y Bárbara, Octavio Zamudio, Miguel Angel Flores, Elizabeth García, Adonay Marquina, Sergio Garay, María Elena Centeno, Christian González, Claudia Barrera, Hugo Rosas, Uriel Salinas, al chimo-escalador-de-montañas, Christian Hernández, Javier León y Frederic Mann.

↪ A Guadalupe Ibargiengoitia y Sergio Escamilla por haberme permitido ser su tesista, este trabajo no hubiera sido posible sin su ayuda y sus palabras. Muchas gracias.

✓

TESIS CON
FALLA DE ORIGEN

Índice general

Agradecimientos	III
Introducción	XI
I Marco Teórico	1
1. Programación Extrema (<i>XP</i>)	3
1.1. Valores de la Programación Extrema	3
1.2. Prácticas de <i>XP</i>	4
1.2.1. El juego de planeación (the planning game)	4
1.2.2. Pequeñas entregas (small releases)	5
1.2.3. Metáfora (metaphor)	6
1.2.4. Diseño simple (simple design)	6
1.2.5. Pruebas (testing)	6
1.2.6. Refabricación (refactoring)	7
1.2.7. Programación por pares (pair programming)	7
1.2.8. Propiedad colectiva (collective code ownership)	8
1.2.9. Integración continua (continuous integration)	8
1.2.10. 40 horas semanales (40-hour week)	8
1.2.11. Cliente en el sitio de desarrollo (on-site customer)	9
1.2.12. Estándares de codificación (coding standards)	9
1.3. Secuencia de desarrollo extremo	9
1.3.1. Planeación del proyecto	10
1.3.2. Iteraciones	11
1.3.3. Reinicio del ciclo	14
2. Lenguaje de Modelado Unificado (<i>UML</i>)	15
2.1. El enfoque orientado a objetos (O-O)	15
2.1.1. Objetos	16

2.1.2.	Mensajes	17
2.1.3.	Clases	17
2.1.4.	Herencia	17
2.1.5.	Polimorfismo	18
2.2.	Introducción al UML (Unified Modeling Language)	18
2.3.	Diagramas UML	19
2.3.1.	Diagramas de clase	20
2.3.2.	Diagramas de casos de uso	22
2.3.3.	Diagramas de objeto	24
2.3.4.	Diagramas de colaboración	25
2.3.5.	Diagramas de secuencia	27
2.3.6.	Diagramas de estado	28
2.3.7.	Diagramas de actividades	30
2.3.8.	Diagramas de componentes	30
2.3.9.	Diagramas de instalación	31
3.	Características del almacén moderno	33
3.1.	Almacenamiento de Mercancías	33
3.1.1.	Almacenamiento por unidades	34
3.1.2.	Almacenamiento especializado	34
3.1.3.	Equipo de almacenamiento	35
3.1.4.	Cuantificación del espacio de almacenamiento	36
3.2.	Manejo de carga	36
3.2.1.	Equipo de transporte	36
3.2.2.	Recepción y Embarque	37
3.2.3.	Surtido o Recolección de Órdenes	38
3.3.	Sistemas de Información	39
3.3.1.	Uso de la Información en el Almacén	39
3.3.2.	Sistemas de Localización	39
3.3.3.	Identificación Electrónica	40
II	Aplicaciones	43
4.	Características del almacén editorial	45
4.1.	Títulos y ejemplares	45
4.2.	Datos Generales	46
4.3.	Tipos de libros	46
4.4.	Paquetes	47
4.5.	Posiciones	47

4.6. Equipo de almacenamiento	47
4.7. Equipo de transporte	48
4.8. Empleados del almacén	48
4.9. Pedidos	50
4.10. Control de posiciones e inventarios	50
4.11. El proceso de surtido	51
5. El Surtido bajo la perspectiva XP/UML	53
5.1. Etapa inicial del proyecto	53
5.2. Primera iteración	54
5.2.1. Desarrollo de la iteración	55
5.2.2. Fin de la iteración	64
5.2.3. Acerca de la planeación en XP	66
5.3. Segunda iteración	67
5.3.1. Desarrollo de la iteración	69
5.3.2. Fin de la iteración	78
5.3.3. Acerca del desarrollo en XP	79
5.4. Tercera iteración	81
5.4.1. Desarrollo de la iteración	82
5.4.2. Fin de la iteración	91
5.4.3. Acerca de la implementación y pruebas en XP	92
5.5. Cuarta iteración	93
5.5.1. Desarrollo de la iteración	94
5.5.2. Fin de la iteración	99
5.5.3. Acerca del diseño	100
5.6. Etapa final	102
5.6.1. Perspectiva del proyecto para el almacén editorial	104
Conclusiones	107

Índice de figuras

2.1. Clase en UML	20
2.2. Asociación entre clases	21
2.3. Multiplicidad de las asociaciones	22
2.4. Agregación de clases	22
2.5. Composición de clases	23
2.6. Generalización de clases	23
2.7. Diagrama de caso de uso	24
2.8. Objeto en UML	25
2.9. Ligas entre objetos	25
2.10. Objeto compuesto	26
2.11. Diagrama de colaboración	26
2.12. Ejemplo de un diagrama de secuencia	27
2.13. Diagrama de estado	29
2.14. Diagrama de actividad	30
2.15. Diagrama de actividades con sincronización	31
2.16. Componente en UML	31
2.17. Diagrama de instalación	32
3.1. Tarima estándar	34
4.1. Distribución de zonas en el almacén editorial	49
4.2. Lector óptico de códigos de barras	51
5.1. Diagrama UML de los estados de un pedido	56
5.2. Caso de uso de la liberación de pedidos	57
5.3. Caso de uso de la pantalla de administración de pedidos	57
5.4. Clases involucradas en un pedido	58
5.5. Relación entre pedido y lista de surtido	58
5.6. Pantalla de generación de pedidos	59
5.7. Clase posición	60

5.8. Pantalla de administración de posiciones	61
5.9. Asociación de los paquetes con títulos y posiciones	62
5.10. Caso de uso para el alta o ajuste de títulos y cantidades	62
5.11. Caso de uso para la consulta de posición en el lector óptico	63
5.12. Caso de uso para la consulta de título en el lector óptico	64
5.13. Diagrama de clase para la primera iteración	65
5.14. Diagrama UML de clase de los conceptos participantes en la metáfora.	68
5.15. Diagrama UML para la clase posición	70
5.16. Pantalla de administración de posiciones	71
5.17. Orden de recorrido en un estante de almacenamiento de 3 niveles de profundidad y 4 de altura.	72
5.18. Caso de uso de asignación de pedidos	73
5.19. Algoritmo para presentar propuestas de surtido	74
5.20. Pantalla de avance de pedidos generados	75
5.21. Clase pedido detalle	76
5.22. Clase pedido	77
5.23. Caso de uso para las consultas "Títulos surtidos" y "Títulos por surtir"	78
5.24. Diagrama de clase para la segunda iteración	79
5.25. Diagrama UML para la clase zona	82
5.26. Relación entre zonas y posiciones	83
5.27. Pantalla de administración de zonas	83
5.28. Pantalla de administración de posiciones	84
5.29. Algoritmo para obtención de propuestas en la opción "Monta- cargas"	86
5.30. Clase paquete	87
5.31. Algoritmo de asignación de cantidades a pedidos	88
5.32. Algoritmo de surtido con confirmación automática	91
5.33. Diagrama UML de clase de la tercera iteración	92
5.34. Clase empleado	95
5.35. Pantalla de administración de empleados	95
5.36. Caso de uso de actividades del supervisor	96
5.37. Clase movimiento	97
5.38. Asociación entre la clase empleado y movimiento	97
5.39. Registro de movimientos en el almacén	98
5.40. Caso de uso de las operaciones de la opción "Calculadora"	100
5.41. Diagrama de clase de la cuarta iteración	101
5.42. Componentes físicos del sistema	105

Introducción

La revolución que en los últimos años han experimentado las tecnologías de la información hace que nos encontremos en una época sin precedentes. La capacidad de procesamiento de datos, así como la transmisión de los mismos, ha mejorado sustancialmente en un tiempo relativamente corto, trayendo como consecuencia equipos de cómputo cada vez más potentes y accesibles. Estos cambios han beneficiado a todas las empresas que ahora pueden sacar provecho de las ventajas que las tecnologías de la información les ofrecen a costos que antes hubieran sido imposibles.

La mayor accesibilidad y demanda de sistemas de información (sistemas contables, administrativos, de control, etc.) ha impulsado a su vez el desarrollo de nuevas áreas del conocimiento, así como nuevos métodos y modelos, entre los que sobresalen la Programación Extrema (XP) y el Lenguaje de Modelado Unificado (UML).

La Programación Extrema intenta establecer prácticas que agilicen y mejoren el desarrollo de software, las cuales surgen a partir de la experiencia real, mientras que UML surge como un intento por unificar y conciliar diversos enfoques de lenguajes de modelado para el diseño orientado a objetos.

Por otra parte, uno de los sectores empresariales que se ha visto beneficiado en gran medida con los avances tecnológicos es el sector logístico, y en particular el manejo de almacenes o centros de distribución. Después de ser vistos durante décadas tan sólo como repositorios de mercancías, ahora representan la ventaja competitiva de muchas empresas que han sabido sacar provecho de su gran potencial para mejorar sus servicios.

La posibilidad de participar en un proyecto para realizar un sistema para un almacén editorial dió la pauta para el desarrollo del presente trabajo, en el que se expone la evolución de un proyecto concreto usando ciertas prácticas de XP, así como el uso de diagramas UML para mostrar los avances en el diseño.

En la primer parte se describen de forma general los conceptos utilizados:

TESIS CON
FALLA DE ORIGEN

- El primer capítulo explica en qué consiste la Programación Extrema, cuáles son sus principios y reglas más importantes, así como la secuencia de desarrollo de software bajo sus principios y prácticas.
- UML es abordado en el segundo capítulo; en él se explican las características de este lenguaje y los diagramas más representativos con que cuenta.
- El tercer capítulo proporciona un panorama general del almacén moderno, su importancia en los negocios y sus características generales.

En la segunda parte de la tesis se aborda el desarrollo un sistema para un almacén editorial, concretamente en el módulo de surtido de pedidos, mediante la aplicación de ciertas prácticas de la Programación Extrema y diagramas UML:

- El cuarto capítulo detalla características muy concretas del almacén en cuestión, lo cual permitirá un mejor entendimiento de los conceptos del almacén y de la complejidad de un sistema de estas características.
- El quinto capítulo está dedicado a detallar el desarrollo del módulo de surtido de pedidos desde la óptica de ciertas prácticas de la Programación Extrema. Los diagramas UML son utilizados en este capítulo como medio para facilitar la comprensión de conceptos y visualizar el avance en el diseño.

Finalmente se presentan las conclusiones derivadas del desarrollo de este trabajo y la bibliografía recomendada.

Parte I

Marco Teórico

TESIS CON
FALLA DE ORIGEN

Capítulo 1

Programación Extrema (XP)

La Programación Extrema (Extreme Programming o XP)¹ es un conjunto de prácticas que buscan mejorar y agilizar el desarrollo de software en todos sus aspectos.

Las prácticas y reglas de XP buscan como fin primordial disminuir el riesgo inherente a proyectos de esta naturaleza así como la obtención de software de alta calidad, a la medida del cliente y con gran adaptación al cambio.

La innovación y fuerza de esta disciplina procede de la conjunción de reglas y prácticas (usadas a lo largo de la historia de la Ingeniería de Software) en ciclos cortos de desarrollo que involucren todas las etapas del ciclo "tradicional" de desarrollo de sistemas (análisis, diseño, desarrollo, pruebas, implementación).

La Programación Extrema fue concebida a principios de los años 90 por un experimentado programador llamado Kent Beck y es uno de los llamados métodos de "peso ligero" ya que propone acciones prácticas, pone énfasis en las tareas que dan resultados concretos e intenta reducir al máximo la burocracia. XP nace propiamente en 1996 al ser aplicada en un proyecto para Daimler-Chrysler.

1.1. Valores de la Programación Extrema

En el corazón de XP residen cuatro valores fundamentales:

Comunicación (Communication): La comunicación como medio de apren-

¹ "extrenia" o "extrenie" se usa en este contexto para referirse a que sus reglas y prácticas son llevadas al límite.

TESIS CON
FALLA DE ORIGEN

dizaje y transmisión de ideas es el primer valor de la Programación Extrema, siendo fomentada constantemente a través de sus prácticas.

Simplicidad (Simplicity): Consiste en el mantenimiento del proyecto a lo largo de su desarrollo en un estado que combine funcionalidad y la mayor sencillez posible.

Retroalimentación (Feedback): Este valor de XP consiste en la evaluación constante del estado del proyecto por todos las personas involucradas en él mediante la realización de pruebas, la constante comunicación y los ciclos cortos de desarrollo.

Valentía (Courage): Representa el valor para aplicar los otros tres valores, es decir, la valentía para mantener proyectos sencillos, someter el producto a continuas evaluaciones o realizar constantes modificaciones.

1.2: Prácticas de XP

La Programación Extrema consta de 12 prácticas que forman su parte modular y que son aplicadas a lo largo del desarrollo de un proyecto.

A continuación se describirá cada una de ellas:

1.2.1. El juego de planeación (the planning game)

El juego de planeación consiste en determinar de manera ágil y eficaz los alcances de cada ciclo de desarrollo (también llamado entrega o iteración). Esta práctica se concibe como la interacción de dos equipos independientes y en cierto sentido ajenos: equipo de negocios y equipo técnico o de desarrollo.

El equipo de negocios está conformado por aquellos miembros de la organización que requiere el software que estarán involucrados en el sistema directa o indirectamente. Este equipo puede estar formado por miembros del área de sistemas de la organización, miembros del personal que vaya a utilizar el software y por cualquier otro elemento de la organización que pueda aportar su conocimiento del negocio o la organización para desarrollar el software.

El equipo técnico consiste de aquellas personas encargadas de construir el software para satisfacer las necesidades de la organización. A propósito de este equipo, en XP las prácticas se exponen bajo la suposición de que el equipo de desarrollo es ajeno a la organización que requiere el software. En caso de que el equipo de desarrollo forme parte de la organización ciertas prácticas deben ser adecuadas a este esquema.

El equipo de negocios considerará que características debe poseer el software para resultar de utilidad para la empresa, mientras que el equipo realizará consideraciones técnicas sobre los planes del equipo de negocios.

El balance que debe existir entre prioridades técnicas y de negocios hace que el juego de planeación involucre un constante diálogo entre ambos equipos, donde cada uno de ellos cuenta con una participación específica en la toma de decisiones.

El equipo de negocios tiene participación en los siguientes temas:

- Alcance del proyecto, es decir, los procesos que deben ser mejorados por el sistema que resulten de valor para la empresa.
- Composición de entregas, es decir, decidir qué funcionalidades se incluirán en la próxima iteración y ordenarlas de acuerdo a su importancia para el negocio.
- Fechas de entrega, debido a que el equipo de negocios sabe cuáles son las épocas en que la presencia del software resultaría de gran utilidad.

A su vez, el equipo de desarrollo tiene decisión sobre los siguientes puntos:

- El equipo de desarrollo será el único capacitado para realizar estimaciones acerca del tiempo de desarrollo de una funcionalidad o componente del sistema.
- Consecuencias técnicas, es decir, el equipo de desarrollo está obligado a informar de las repercusiones técnicas que las decisiones del equipo de negocios pueden acarrear.
- La organización del equipo y de los métodos de trabajo quedarán a cargo del equipo técnico para permitirles adaptarse rápidamente y de la mejor manera a la cultura de negocios en la cual van a trabajar.
- Agenda detallada, es decir, el equipo de desarrollo decidirá qué funcionalidades tienen prioridad en una iteración en base a estimaciones de tiempo o riesgo.

1.2.2. Pequeñas entregas (small releases)

Los proyectos en XP evolucionan a través de entregas o ciclos cortos de desarrollo que involucren todas las etapas del ciclo "tradicional" de desarrollo de sistemas².

²Las fases del ciclo de vida o ciclo tradicional de desarrollo de sistemas son: análisis, diseño, desarrollo o codificación, pruebas e implantación.

TESIS CON
FALLA DE ORIGEN

Cada ciclo o iteración debe formar un todo, siendo lo más pequeño posible, conteniendo las funcionalidades más valiosas y siendo entregado al cliente al momento de su conclusión por parte del equipo de desarrollo.

La realización de pequeñas entregas disminuye la incertidumbre y fomenta la comunicación con el cliente debido a que los avances concretos del proyecto le son entregados de manera frecuente.

Además, el hecho de que el cliente pueda comenzar a usar el software desde las primeras etapas de desarrollo genera familiaridad con el producto y proporciona sugerencias por parte de los usuarios para realizar cambios o modificaciones sobre la marcha.

1.2.3. Metáfora (metaphor)

En proyectos que requieren la participación de varias personas, resulta útil contar con una idea común acerca del proyecto que facilite la comunicación entre los miembros del grupo.

La metáfora en XP es la elaboración de una idea general del sistema, que prescindiera de tecnicismos innecesarios y que permita entender los elementos básicos del sistema y sus relaciones de manera simple tanto a los involucrados en el proyecto como a las personas ajenas a éste.

1.2.4. Diseño simple (simple design)

La incertidumbre respecto a las necesidades futuras del software y a las condiciones del negocio en el futuro fomenta el hecho de que los proyectos de software sean muy cambiantes. Esta práctica postula que un sistema debe contener en todo momento sólo aquellas funcionalidades que resultan necesarias y que han sido añadidas conforme las condiciones del entorno lo han requerido.

Esta práctica se basa en la premisa de que el costo del cambio resulta mayor en sistemas que cuentan con funcionalidad extra que en aquellos dotados de lo estrictamente necesario.

1.2.5. Pruebas (testing)

La confianza de un proyecto de software recae en su buen funcionamiento, por lo que esta práctica de XP recomienda el uso exhaustivo de pruebas para la verificación y detección de errores en el sistema.

Las pruebas son realizadas por los programadores durante el desarrollo e incluyen una diversa gama de métodos entre los que destacan las pruebas de caja negra (donde importa el aspecto funcional del sistema), de caja blanca

(aquellas que toman en cuenta el funcionamiento interno del software), automatizadas (consistentes en la codificación de pruebas en un software especial que deben ser superadas por el sistema), las contenidas en el propio sistema (como los mensajes de error escritos por los programadores) y las de integración (aquellas donde se valida que no existan contradicciones entre componentes escritos por diversos miembros del equipo de desarrollo).

A este uso extensivo de pruebas por parte de los programadores se deben añadir las pruebas que realizan los propios usuarios cuando cada entrega del sistema se encuentra a su alcance, siendo posteriormente traducidas a pruebas de los programadores e incrementando gradualmente la confianza en el proyecto.

1.2.6. Refabricación (refactoring)

La refabricación³ es una práctica que busca la simplicidad y el mejoramiento constante del código sin modificar su funcionalidad.

Estas mejoras incluyen la eliminación de funciones obsoletas, agregación de comentarios, reescritura de código redundante, enriquecimiento de funciones para uso general o cambio de nomenclatura fuera de estándar entre otras, y pueden ser realizadas por cualquier miembro del equipo en cualquier oportunidad.

Este conjunto de mejoras puede verse como un "gasto" de tiempo en el mejoramiento de código que ya resultaba funcional pero en XP se menciona que a la larga estos cambios proporcionan beneficios como fomentar la comunicación en el equipo o tener un entendimiento más claro del sistema.

1.2.7. Programación por pares (pair programming)

La programación por pares es aquella práctica en la que dos personas se enfrentan al desarrollo de software, analizando y discutiendo posibles soluciones con la asistencia de una sola computadora.

Esta práctica es altamente dinámica ya que un programador se dedica a codificar (generalmente, el código es escrito por aquella persona que posea la idea más clara sobre cómo resolver el problema en ese momento) y el otro supervisa la codificación, propone mejoras, analiza alternativas o diseña pruebas. A lo largo de un día el intercambio de papeles en una pareja se da

³Debido a la falta de estándares en cuanto a la traducción de conceptos de XP al español, en el presente trabajo se utilizó la palabra refabricación ya que proporciona cierta idea de la práctica.

TESIS CON
FALLA DE ORIGEN

constantemente, dándose inclusive el cambio de parejas varias veces en un día dependiendo de las condiciones de desarrollo.

La programación por pares permite una mayor comunicación y un traspaso de conocimiento más directo en el equipo siendo especialmente útil para integrar a nuevos miembros al equipo, reducir brechas de conocimiento entre programadores y tener la certeza de que el proyecto no se detendrá en ausencia de algún miembro.

1.2.8. Propiedad colectiva (collective code ownership)

Esta práctica establece que la fase de desarrollo del proyecto es responsabilidad y propiedad de todos los miembros del equipo técnico, por lo cual se permite que cualquier pareja manipule o modifique el código (obviamente con el objeto de mejorar o complementar la funcionalidad) en el momento que lo requiera, sin importar quién lo haya escrito originalmente.

Como consecuencias de esta práctica tenemos la posible mejora o simplificación del sistema con el paso del tiempo, así como el mayor entendimiento de su funcionamiento por parte de cada pareja de programadores que intente mejorar el código.

1.2.9. Integración continua (continuous integration)

La existencia de diversas parejas, el dinamismo del desarrollo y las mejoras continuas producto de la refabricación, la propiedad colectiva y las demás prácticas hacen que sea necesario integrar los cambios realizados al sistema como mínimo una vez al día.

La integración continua beneficia al equipo de desarrollo al permitirle contar con una versión actualizada del proyecto accesible a todos, así como disminuir el riesgo del proyecto mediante la detección oportuna de errores de integración.

1.2.10. 40 horas semanales (40-hour week)

En los proyectos de desarrollo de software resulta fundamental el factor humano, por lo cual XP hace gran énfasis en el bienestar de las personas. Una forma de lograrlo es mediante una adecuada planeación que evite la sobrecarga de trabajo, el estrés o el desánimo.

La Programación Extrema propone semanas laborales de 40 horas como estrategia para lograr un buen ritmo de desarrollo y un equipo de trabajo lúcido, motivado y contento. Se permite la ocasional ruptura de esta práctica debido a factores externos, aunque más de dos semanas consecutivas con

horas extra se consideran un signo negativo en el proyecto y un motivo para una reunión de replaneación.

1.2.11. Cliente en el sitio de desarrollo (on-site customer)

La funcionalidad del sistema y las ventajas que éste proporcione al usuario final dependen en gran medida en la calidad y cantidad de comunicación entre clientes y equipo de desarrollo.

Esta fuerte dependencia en los gustos y necesidades del cliente hacen que su participación a lo largo del proyecto se vuelva indispensable, por lo que se propone que al menos una de las personas que vayan a utilizar el producto acompañe constantemente al equipo para resolver las dudas que surgirán en el equipo de desarrollo conforme avance el proyecto, así como para aportar ideas y sugerencias para el mejoramiento del sistema.

En caso de que resulte imposible para el cliente realizar sus labores cotidianas separado físicamente de su lugar de trabajo se recomienda que al menos se faciliten medios para que los programadores puedan consultar sus dudas con él en todo momento.

1.2.12. Estándares de codificación (coding standards)

La refabricación, la propiedad colectiva, la programación por pares y el dinamismo del proyecto hacen que resulte indispensable un lenguaje común o manera estándar de escribir el código entre los miembros del equipo.

La meta de esta práctica es que el estándar sea tan aceptado y general que con el paso del tiempo sea imposible distinguir el código escrito por dos programadores, lo cual permitirá que la comunicación y el entendimiento mejoren sustancialmente.

1.3. Secuencia de desarrollo extremo

Por secuencia de desarrollo extremo nos referiremos a la forma en que los valores y las prácticas de XP se aplican durante el desarrollo de un proyecto concreto.

XP parte inicialmente de los siguientes supuestos:

- Se está desarrollando software a la medida.
- El proyecto en cuestión posee un grado de complejidad considerable.

TESIS CON
FALLA DE ORIGEN

- Los clientes no tienen una idea clara de los alcances que tendrá el sistema y por tanto muchas veces sus peticiones serán vagas o contradictorias.
- Las condiciones del negocio son fluctuantes, por lo que es necesario que el sistema a construir sea fácilmente adaptable a los cambios.
- Se cuenta con un equipo compacto de desarrollo (no más de 10 o 12 personas).

1.3.1. Planeación del proyecto

Las reuniones de planeación de entregas forman la parte medular de la planeación ya que en ellas el equipo de desarrollo y de negocios realizan un plan que abarca cada entrega o iteración individual y su desarrollo se menciona a continuación:

1. El primer paso de la planeación consiste en la identificación y escritura de requerimientos de usuario. Un requerimiento⁴ de usuario es una descripción sucinta de una característica, funcionalidad o tarea deseable en el sistema escrita por el equipo de negocios en lenguaje no técnico (se recomienda escribir los requerimientos en tarjetas que permitan su fácil catalogación).
2. Posteriormente, el equipo de desarrollo asignará una dificultad técnica y un tiempo de desarrollo estimado en "semanas de programación ideal".⁵ Para lograr estimaciones precisas, el equipo de desarrollo realizará todas las preguntas necesarias en caso de que los requerimientos no sean claros, y propondrá dividir o agrupar requerimientos si así lo requiere.
3. Una vez que los requerimientos de usuario han sido estimados, el equipo de negocios los ordenará de acuerdo a su valor de la siguiente manera:
 - Requerimientos esenciales en el sistema.

⁴ XP utiliza el término "user story", pero en este trabajo se prefirió utilizar la palabra "requerimiento" por considerarlo más apegado a la realidad.

⁵ Una semana de programación ideal equivale a cinco días de trabajo del equipo encargado del requerimiento bajo el supuesto de que no se sufren interferencias de ninguna especie y se sabe exactamente qué hacer, es decir, se tienen objetivos precisos y no hay dudas ni necesidad de reunirse con el usuario para realizar aclaraciones.

- Requerimientos que agregan valor.
 - Requerimientos que agregan funcionalidad.
4. El equipo de desarrollo, a su vez, los ordenará de acuerdo a su riesgo técnico:
 - Requerimientos que cuentan con una estimación precisa.
 - Requerimientos con cierta certeza en la estimación.
 - Requerimientos de difícil o nula estimación.
 5. El equipo de desarrollo informará al cliente la velocidad o factor de desarrollo del equipo (este concepto se explica en 1.3.2), lo cual le permitirá saber el tiempo estimado de construcción para los requerimientos que elija en la presente iteración.

El conjunto de historias elegido (el cual dependerá del ordenamiento de las historias de acuerdo a su valor y riesgo, así como de la velocidad de desarrollo del equipo) constituye una iteración, cuya duración se recomienda sea de entre 1 y 4 semanas, ya que con ciclos cortos el número de iteraciones resulta grande, favoreciendo una mayor retroalimentación con el cliente proporcionándole certidumbre sobre el avance del proyecto.

Obviamente, la primera iteración deberá contener aquellos requerimientos esenciales en la estructura del sistema dando como resultado un producto que funciona completamente, aunque sea en un estado embrionario. Además, la primera iteración es un buen momento para concebir una metáfora que sirva de guía a lo largo del proceso.

Aquellos requerimientos desarrollados inicialmente (consideradas más valiosos o riesgosos por desarrollar) podrán participar en un mayor número de iteraciones si así lo requieren, fomentando que los errores estructurales en el sistema sean eliminados rápidamente y por tanto el riesgo inherente al sistema también disminuya. De forma análoga, los requerimientos incorporados posteriormente (considerados de menor importancia) contarán con un menor tiempo de desarrollo respecto a aquellos construidos en anteriores iteraciones.

1.3.2. Iteraciones

Planeación

Una vez conformada la iteración, el equipo de desarrollo procederá a dividir los requerimientos en tareas o actividades de programación (se re-

**TESIS CON
FALLA DE ORIGEN**

comienda escribir las tareas en tarjetas para su fácil asignación y manipulación). Muchas veces una tarea será de utilidad en varios requerimientos y en otras habrá tareas que no tengan relación directa con ningún requerimiento en particular.

Una vez identificadas, cada equipo elegirá las tareas a realizar aceptando la responsabilidad sobre ellas y estimando el tiempo de programación ideal que tomará realizarlas (la construcción de una tarea promedio deberá llevar pocos días de programación ideal).

Al inicio de cada iteración los miembros del equipo de desarrollo establecerán sus factores de desarrollo de acuerdo a la experiencia previa; el factor de desarrollo es un número calculado en base a la relación entre el tiempo calendario que le lleva a un programador desarrollar una tarea y el tiempo de programación ideal estimado para la misma.⁶

El establecimiento de factores de desarrollo resulta una forma de medición práctica y sensible ya que la programación por pares, el tiempo invertido en reuniones y pláticas con el cliente o la ayuda a otros compañeros se harán visibles en los factores de desarrollo de cada programador.

El factor o velocidad de desarrollo del equipo será igual al promedio de los factores de desarrollo de sus miembros, por lo cual tomando como base este factor el equipo de desarrollo podrá elegir el número de tareas que puede llevar a cabo dentro de una iteración.

Estrategia de diseño

Como ya se había mencionado en las prácticas de la Programación Extrema, una de las metas que busca XP es la simplicidad en todos los aspectos del desarrollo de software. En el caso de la estrategia de diseño la simplicidad se logra partiendo de un diseño inicial del sistema muy sencillo al principio, pero que conforme pase el tiempo se le agreguen características por demanda hasta lograr sistemas muy elaborados con sólidos soportes.

Esta estrategia es contraria a la práctica conocida como "Gran Diseño Previo", en la que se realiza un diseño general que intenta abarcar tanto la funcionalidad actual como aquella que puede requerirse en el futuro.

XP se opone a esta manera de atacar el diseño debido a que limita la posibilidad del cliente de realizar modificaciones a lo largo del proceso

⁶Factores de desarrollo bajos significan una mayor experiencia o velocidad: si en 3 semanas de tiempo calendario se desarrollan tareas que suman 3 días de programación ideal entonces el Factor Desarrollo = $15/3 = 5$, en cambio si en 3 semanas de tiempo calendario se realizan tareas que suman 8 días de programación ideal entonces Factor Desarrollo = $15/8 = 1.875$.

y tiende a consumir tiempo valioso en detalles poco claros o indefinidos debido a que al inicio del proceso se carece de elementos para realizar un diseño general.

Pruebas

XP propone el sometimiento intensivo del sistema a pruebas por parte de todas las personas involucradas en el desarrollo del mismo entre las cuales destacan las pruebas automatizadas antes, durante y después del desarrollo, así como las pruebas por parte de los usuarios finales.

Una innovación de XP consiste en la designación de un probador de tiempo completo que evalúe las pruebas existentes, proponga nuevos métodos y traduzca las inquietudes de los usuarios en pruebas palpables.

A primera vista podría parecer que el uso de pruebas intensivas alenta el proceso, pero XP sostiene que a largo plazo se incrementa la confianza, aumenta la velocidad de desarrollo y el número de errores descendiendo con el tiempo debido a que las pruebas pueden ser reutilizadas y el cliente puede probar y manipular al producto a su gusto.

Desarrollo

Una vez estimadas y repartidas las diferentes tareas entre los miembros del equipo de desarrollo, se comienza a desarrollar mediante equipos de dos personas (programación por pares).

El desarrollo de cada tarea incluye la codificación específica para la tarea, el diseño de pruebas, la refabricación del código que así lo requiera y la integración de la tarea al proyecto en conjunto.

En XP se propone la integración del código naciente al menos una vez al día, para de esta forma evitar enormes integraciones futuras, divergencias de programación o fragmentación de código.

XP recomienda para el equipo de desarrollo la realización de reuniones diarias (de carácter informal y con una duración de pocos minutos) para discutir los problemas que vayan surgiendo, evaluar los avances del día anterior y formar equipos de programación. Este tipo de reuniones mantienen la cohesión del equipo y dan una idea clara del avance del proyecto desde el punto de vista del grupo encargado del desarrollo.

Si el equipo nota que se realizó una planeación imprecisa y que el plazo para la actual iteración resultará insuficiente para desarrollar los requerimientos de usuario asignados, existen varias opciones entre las cuales destaca la solicitud al cliente de una reducción en el alcance de algunos requere-

TESIS CON
FALLA DE ORIGEN

rimientos de usuario, la eliminación de tareas innecesarias o el diferimiento de requerimientos para una próxima iteración.

Asimismo, si el cliente desea añadir un requerimiento de mayor valor no contemplado en esta iteración, éste podrá ser agregado sustituyéndolo por requerimientos de la actual iteración con tiempos de desarrollo equivalentes al nuevo requerimiento a incluir.

Una vez que las tareas que componen una historia son terminadas e integradas al proyecto, el cliente recibe una nueva versión del sistema que será sometido a pruebas por parte de los usuarios finales para detectar errores no previstos en la fase de desarrollo.

1.3.3. Reinicio del ciclo

Concluido el plazo de la iteración y realizados todos los requerimientos de usuario planeados, el equipo de negocios y el equipo de desarrollo se encuentran de nuevo en situación de reunirse y discutir la próxima iteración, cuyo contenido y duración dependerá en gran medida de la iteración anterior, así como de las circunstancias actuales del negocio que quizá orienten el desarrollo en otra dirección.

Generalmente, una iteración típica incluirá tanto nuevos requerimientos de usuario como requerimientos ya construidos que requieran corrección (cuando no satisfacen las pruebas de los usuarios finales), mejoramiento o adición de funcionalidades.

Capítulo 2

Lenguaje de Modelado Unificado (*UML*)

El diseño constituye una parte muy importante en todo proyecto de software ya que representa el soporte estructural del mismo, además de que pone de manifiesto los elementos y relaciones que intervienen en el fenómeno. Por lo tanto factores como la calidad, sencillez y adaptabilidad del software dependen directamente del diseño del proyecto.

En el proyecto para el almacén editorial el diseño tuvo gran relevancia, por lo que en el presente trabajo se decidió mostrar aspectos importantes en el diseño mediante diagramas UML, lo cual permite tener una idea más clara sobre conceptos específicos de un almacén editorial. Es por esta razón que se incluye el presente capítulo para explicar los aspectos importantes del lenguaje UML, así como sus diagramas más representativos.

2.1. El enfoque orientado a objetos (O-O)

La creación de software para la resolución de una problemática específica es por lo general una tarea que conlleva un fuerte conocimiento de los elementos que intervienen en la situación, así como de las relaciones entre estos elementos. Este conocimiento es indispensable para construir un modelo que se apegue lo mejor posible a la realidad y que ayude a resolver la problemática planteada.

El diseño y la complejidad de las primeras computadoras obligaba a los programadores a plantear cualquier solución en términos del comportamiento de la máquina dando origen a modelos y lenguajes de programación basados en funciones y procedimientos, lo cual ligaba el diseño estructural

TESIS CON FALLA DE ORIGEN

del modelo con su funcionalidad.

El enfoque orientado a objetos surge posteriormente como una forma de visualizar las situaciones desde una visión más apegada a nuestra percepción del entorno, permitiendo crear modelos más entendibles, reutilizables y fáciles de modificar.

La orientación a objetos se basa en la idea de que cualquier fenómeno puede ser visto como objetos con características específicas interactuando entre sí. Con este método es posible generar soluciones en las cuales estructura y funcionalidad resultan independientes, existe una adaptación más fácil a los cambios que se registren en el tiempo y se posibilita la reutilización de elementos comunes debido a la estructura invariante de los objetos bajo cualquier contexto.

La orientación a objetos se basa primordialmente en los conceptos de objeto, mensaje, clase, herencia y polimorfismo.

2.1.1. Objetos

El enfoque O-O considera a los objetos como las unidades atómicas de los fenómenos, las cuales realizan o responden a tareas específicas y poseen ciertos atributos. Los objetos son un intento de representar nuestra visión cotidiana del mundo y prácticamente cualquier cosa puede ser vista como tal, desde una persona hasta una página de Internet.

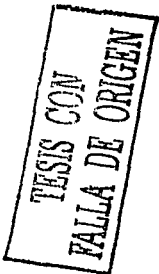
Características principales de un objeto

Un objeto posee tres características principales: estado, comportamiento e identidad.

- El estado se define como el conjunto de propiedades de un objeto, donde una propiedad representa una característica específica del mismo.

Los objetos poseen propiedades variables y constantes por lo cual el estado de un objeto será igual al valor de sus diferentes propiedades en un momento específico, lo cual hace que el estado de un objeto sea una condición que varía dinámicamente con el tiempo.

- El comportamiento de un objeto se define como el conjunto de métodos del objeto, es decir, el conjunto de tareas o acciones realizadas por el objeto o sobre el objeto. Estas acciones representan la única forma que tiene el objeto de interactuar con el mundo exterior.



- La identidad de un objeto representa una característica intrínseca del objeto que permite identificarlo sin ambigüedades, es decir, permite diferenciar dos objetos con estados idénticos.

2.1.2. Mensajes

Un mensaje es la especificación de la transmisión de información entre objetos y representa la unidad fundamental de comunicación entre ellos.

El envío de un mensaje implica una acción en el objeto receptor, el cual puede ser alguno de los tipos básicos siguientes:

- Llamado, el cual invoca un método en el objeto receptor.
- Retorno, el cual regresa un valor al emisor.
- Envío, el cual envía un valor al receptor.
- Creación, el cual crea un objeto.
- Destrucción, el cual destruye un objeto.

La flexibilidad de los modelos realizados bajo la O-O se basa en el dinamismo de los mensajes, que además refleja la independencia entre estructura y funcionalidad.

2.1.3. Clases

La complejidad de los fenómenos que se observan en la vida real hace que sea muy difícil tener un panorama completo de la situación desde el principio. El proceso de abstracción nos permite simplificar nuestro panorama buscando características similares entre diversos objetos agrupándolos bajo un criterio particular.

Las clases concentran las características generales de los objetos, mientras que las características particulares se encuentran contenidas en los objetos mismos, por tanto, un objeto es a una clase lo que una casa a un plano arquitectónico, es decir, la creación de todo objeto implica un proceso en el cual se realiza una instancia de la clase a la que pertenece.

2.1.4. Herencia

Existen diversas maneras de clasificar elementos. Dentro del enfoque orientado a objetos la forma más común de representar la clasificación es

por medio de una técnica conocida como herencia, la cual permite construir una clase a partir de otras ya existentes.

El producto de este proceso (subclase) contiene tanto atributos generales heredados de sus clases antecesoras (superclases) como características propias específicas, siendo un mecanismo sencillo para compartir atributos y métodos entre clases.

Jerarquías de clases

La existencia de clases implica una jerarquización, existiendo dos operaciones primordiales:

- La generalización consiste en agrupar características o elementos comunes de varias clases en una clase más general conocida como superclase.
- La especialización consiste en agregar características o atributos que no se encuentran en las clases ya existentes dando como resultado una nueva clase derivada de aquellas más generales.

2.1.5. Polimorfismo

En el enfoque orientado a objetos el término polimorfismo significa la capacidad que tienen los objetos de una clase de responder a un mensaje específico de manera diferente a como lo harían los objetos de su superclase. Cualquier subclase hereda las operaciones de su superclase y puede modificar el comportamiento de esas operaciones dependiendo de sus necesidades particulares.

2.2. Introducción al UML (Unified Modeling Language)

El desarrollo creciente de la tecnología, la globalización y la cada vez mayor competitividad en las empresas han fomentado la necesidad de desarrollar software cada vez más complejo y sofisticado que soporte los demandantes requerimientos de los usuarios. Esta creciente complicación ha fomentado el uso de modelos para facilitar nuestra comprensión de los fenómenos así como servir de guía para una realización más eficiente de las cosas.

Una gran cantidad de métodos y modelos basados en el enfoque orientado a objetos comenzaron a surgir a partir de su popularización en los años

80. En términos generales compartían los mismos conceptos básicos aunque la forma de nombrar y representar conceptos variaba de uno a otro. Esta diversificación resultaba contraproducente para el desarrollo de la tecnología O-O ya que desalentaba tanto a los nuevos usuarios como a las empresas que visualizaban el gran potencial por desarrollar con tecnología basada en este enfoque.

UML surge como un intento por conciliar y unificar los lenguajes de modelación desarrollados hasta ese momento. UML surge a fines de los años 90 a partir de la unión de tres personas que habían desarrollado independientemente métodos de modelación bajo el enfoque orientado a objetos para desarrollar un enfoque unificado.

Los objetivos de Grady Booch, Jim Rumbaugh e Ivar Jacobson al crear el UML fueron:

- El uso de conceptos orientados a objetos para representar sistemas complejos.
- Establecer una relación explícita entre el nivel conceptual y el nivel práctico del modelo.
- Tomar en cuenta los factores de escala inherentes a sistemas críticos complejos.
- Crear un lenguaje que pueda ser usado tanto por humanos como por máquinas.

Muchas organizaciones se dieron cuenta de la importancia del desarrollo de UML, por lo cual su desarrollo ha sido apoyado por organizaciones e importantes empresas de la industria de la información.

UML es un lenguaje de modelado con muchas ventajas como su expresividad, consistencia y aplicación general. Asimismo, la definición precisa de UML no representa una barrera para la extensibilidad del lenguaje lo cual incrementa su confiabilidad.

2.3. Diagramas UML

Los diagramas son la forma básica de representación de fenómenos en el entorno UML, existiendo varios tipos dependiendo del punto de vista o grado de complejidad requerido.

TESIS CON
FALLA DE ORIGEN

2.3.1. Diagramas de clase

En UML los diagramas de clase sirven para representar la estructura estática de un modelo, ya que representan las clases que intervienen en el fenómeno y las relaciones que hay entre ellas.

Clases

En UML las clases se representan como rectángulos con tres compartimentos. En el primero de ellos se coloca el nombre, mientras que los otros dos contienen los atributos y métodos de la clase. (Figura 2.1)

UML define tres niveles de visibilidad para los métodos y atributos:

- public (+) El elemento es visible para todos los clientes de la clase, es decir, para todas las clases que tengan interacción con ella.
- protected (#) El elemento es visible para las subclases de la clase.
- private (-) El elemento es visible sólo para la propia clase.

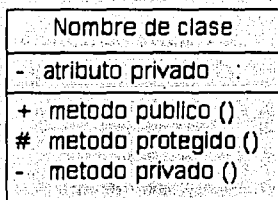


Figura 2.1: Clase en UML

Asociación

Una asociación es una relación estructural entre clases que se representa mediante una línea que une las clases asociadas y un nombre que la define. Los puntos finales de una asociación se conocen como roles, que representan la forma en que una clase ve a la otra a través de la relación y son esquematizados por medio de nombres escritos en los extremos de la línea de asociación (fig. 2.2).

La instancia de una asociación se conoce como liga y representa una relación estructural entre los objetos que son instancias de las clases involucradas en la asociación.

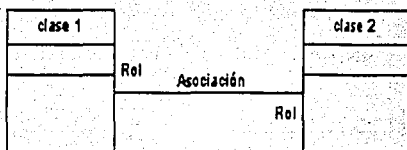


Figura 2.2: Asociación entre clases

Multiplicidad de las asociaciones

Los roles involucrados en una asociación poseen una multiplicidad que indica el número de objetos que estarán relacionados a un objeto de la otra clase en la relación. La multiplicidad es una propiedad del rol de una asociación, se escribe en los extremos de la línea de asociación y toma alguno de los siguientes valores (Figura 2.3):

1 Uno y sólo uno.

0..1 Cero o uno.

M..N De M a N.

* Cualquier entero positivo (incluyendo el cero).

0..* Cualquier entero positivo (incluyendo el cero).

1..* Cualquier entero positivo.

Agrogación

La agregación representa un tipo especial de asociación en la cual dos clases se encuentran relacionadas, pero una juega un papel más importante que la otra en el fenómeno. Este tipo de relación se representa visualmente mediante una línea de asociación con un pequeño diamante junto a la clase más importante e implica dependencia de objetos, atributos o métodos de una clase con respecto a otra. (Figura 2.4)

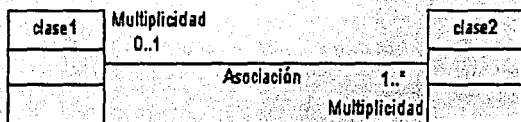


Figura 2.3: Multiplicidad de las asociaciones

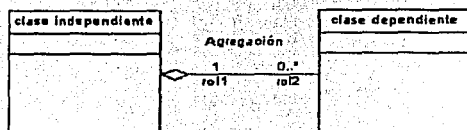


Figura 2.4: Agregación de clases

Composición

La composición es un caso especial de agregación en el cual ciertas clases están contenidas físicamente en una clase. Este caso especial de relación se representa con un diamante negro junto a la clase compuesta. (Figura 2.5)

Generalización

El término generalización es usado en UML para especificar la herencia u otra relación de jerarquía entre un elemento general y otro específico, representándose como una flecha que va de la clase especializada a la clase general. (Figura 2.6)

2.3.2. Diagramas de casos de uso

Los diagramas de casos de uso son diagramas que representan un sistema desde el punto de vista de los usuarios en un contexto dado.

El modelo de casos de uso involucra tres elementos principales: actores, casos de uso y relaciones entre ellos.

Los actores representan el papel que las personas, componentes de hardware o incluso otros sistemas juegan al interactuar con un sistema y se representan visualmente mediante un muñeco (fig. 2.7).

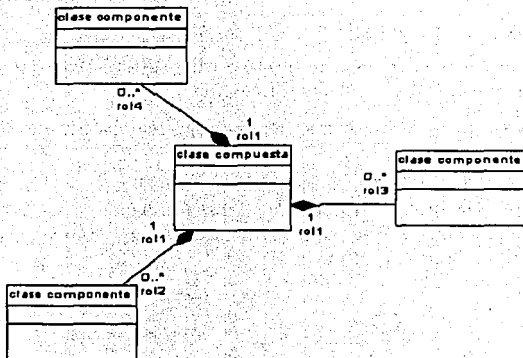


Figura 2.5: Composición de clases

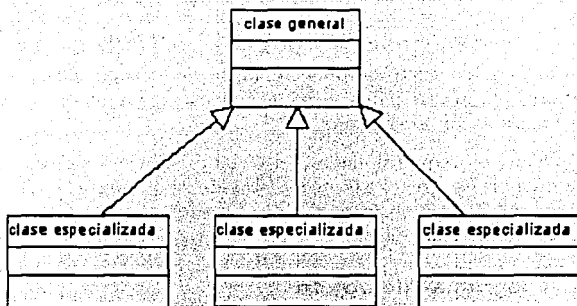


Figura 2.6: Generalización de clases

Existen cuatro tipos de actores:

- Actores principales: son quienes utilizan las principales funciones de un sistema.
- Actores secundarios: representan a quienes ejecutan labores adminis-

trativas o de mantenimiento al sistema.

- **Hardware externo:** es el conjunto de dispositivos físicos que se requieren para que el sistema pueda funcionar.
- **Otros sistemas:** son otros entornos que tienen interacción con el sistema.

Los casos de uso representan una abstracción de la funcionalidad del sistema, es decir, muestra lo que hace un sistema sin especificar la manera en la que internamente esto se realiza. Un caso de uso se representa visualmente mediante una elipse (fig. 2.7).

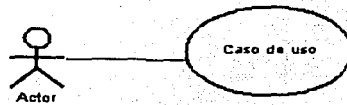


Figura 2.7: Diagrama de caso de uso

Los diagramas de casos de uso se utilizan principalmente para dos fines:

- **Modelar el contexto de un sistema,** es decir, especificar los diferentes actores que participan y el papel que cada uno de ellos desempeña al interactuar con el sistema.
- **Modelar los requerimientos del sistema,** lo cual permite concebir al sistema como una "caja negra" donde se presenta lo que el usuario desea que realice el sistema sin importar la manera en que el sistema funciona internamente.

2.3.3. Diagramas de objeto

Los diagramas de objeto representan la estructura de un sistema desde el punto de vista de los objetos y sus relaciones (liga).

Objetos

Los objetos se representan como rectángulos con dos compartimentos. En el primero de ellos se escribe el nombre y la clase del objeto separados por dos puntos, mientras que en el segundo se incluyen los atributos del objeto. En UML todo elemento que represente una instancia se debe subrayar, por lo cual el nombre y la clase del objeto se subrayan (fig. 2.8).

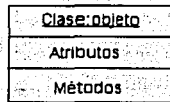


Figura 2.8: Objeto en UML

Ligas

Las ligas son instancias de asociaciones entre clases y representan conexiones entre objetos. La manera de representarlas en UML es por medio de líneas entre los objetos relacionados. (fig. 2.9)

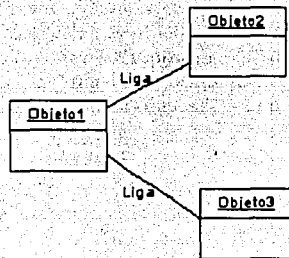


Figura 2.9: Ligas entre objetos

Objetos compuestos

Un objeto puede estar compuesto a su vez por otros objetos: la manera de representarlo en UML es mediante un objeto normal cuyos atributos son reemplazados por objetos. Un diagrama de objeto de este tipo es una instancia de un diagrama de agregación de clases. (fig. 2.10)

2.3.4. Diagramas de colaboración

Los diagramas de colaboración son una extensión de los diagramas de objetos, ya que además de representar la estructura de los objetos, modela las interacciones dinámicas que se dan entre ellos mediante mensajes.

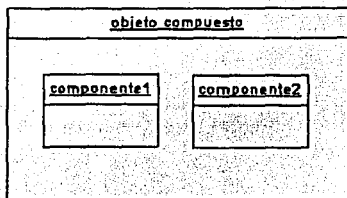


Figura 2.10: Objeto compuesto

La forma de representar los mensajes entre objetos es por medio de flechas que viajan a través de las ligas existentes entre ellos y que apuntan hacia el objeto destino del mensaje. El tiempo no está representado explícitamente en este tipo de diagramas, por lo cual se requiere numerar los diversos mensajes en caso de querer hacer referencia directa a su cronología.

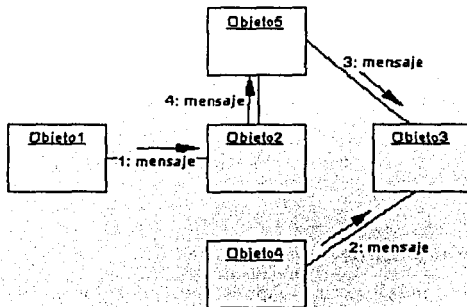


Figura 2.11: Diagrama de colaboración

Objetos activos

La activación de un objeto corresponde al lapso de tiempo en el que ejecuta una acción y por tanto toma el control del flujo. Un objeto activo

puede transmitir el control a objetos pasivos mediante el envío de mensajes. Los elementos activos se representan en UML mediante un dibujo de objeto con recuadro engrosado.

2.3.5. Diagramas de secuencia

Los diagramas de secuencia se utilizan para representar interacciones entre objetos desde el punto de vista temporal, sin dar importancia a la estructura física en la cual se desarrollan estas interacciones.

Bajo este esquema los objetos se dibujan como rectángulos con un eje vertical llamado línea vital cuyo extremo superior representa el momento cero. Los mensajes se esquematizan como flechas horizontales que viajan entre las líneas vitales de los objetos involucrados cuya altura depende de la sincronía de las interacciones.

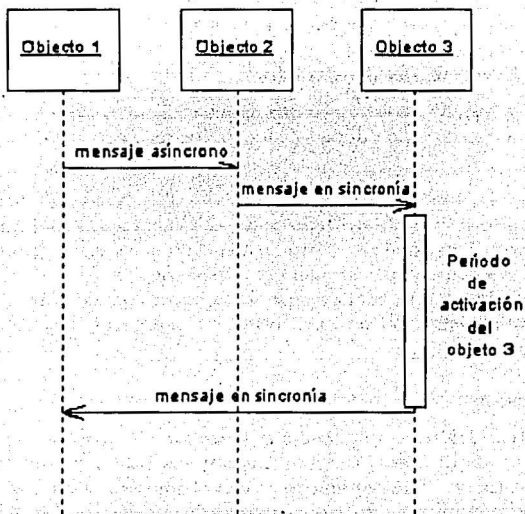


Figura 2.12: Ejemplo de un diagrama de secuencia

En los diagramas de secuencia se pueden describir dos tipos de trans-

TESIS CON
FALLA DE ORIGEN

misión de mensajes (fig. 2.12):

- **Transmisión síncrona:** en la cual el emisor del mensaje es bloqueado mientras el objeto receptor procesa el mensaje. Se representa como una flecha cuya punta está completa.
- **Transmisión asíncrona:** en este tipo de interacción el objeto emisor puede realizar otros procesos independientemente del objeto destino del mensaje. Se simboliza como una flecha con media punta.

Este tipo de diagramas permite representar la activación de objetos mediante un rectángulo que recorre la línea vital durante el tiempo que dura su activación.

Los diagramas de colaboración y secuencia resultan equivalentes debido a que ambos describen el comportamiento dinámico de un sistema.

2.3.6. Diagramas de estado

Los diagramas de estado representan el cambio de estado de un objeto durante su ciclo de vida dependiendo de los diferentes estímulos que reciba.

Estados

El estado de un objeto representa el valor de sus atributos en un momento dado. Un estado se esquematiza como un rectángulo con esquinas redondeadas que incluye un nombre que lo identifica.

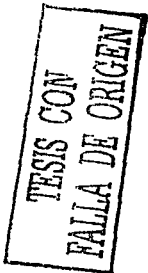
Un estado inicial y final se representa en UML mediante un círculo negro o un círculo negro rodeado por otra circunferencia, respectivamente. (fig. 2.13)

Transiciones

Una transición es una conexión entre estados (no necesariamente diferentes) dotada de una dirección. Los objetos pasan de un estado a otro al ser disparada una transición, que se considera instantánea al no haber posibilidad de que un objeto carezca de estado. Las transiciones se representan por medio de flechas entre estados.

Eventos

Un evento representa la ocurrencia de una situación específica dentro del entorno del fenómeno. Los eventos sirven como disparadores de transiciones entre estados y se ponen como etiquetas en las transiciones.



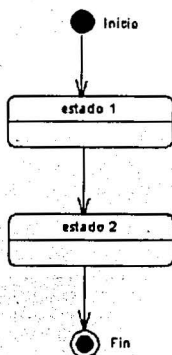


Figura 2.13: Diagrama de estado

Acciones y actividades

Una transición entre estados puede ir acompañada de otros procesos que se ejecutan con el cambio de estado de un objeto. Una acción representa la ejecución de alguno de los métodos del objeto que recibe el evento y pueden ser ejecutadas al entrar o salir de un estado o al darse un evento determinado mientras el objeto se encuentra en ese estado. En UML se representan este tipo de acciones con los siguientes comandos dentro del diagrama de estado:

entra/Acción para simbolizar una acción al entrar a un estado.

sale/Acción para representar una acción al salir de un estado.

Z/Acción para acciones que se disparan al darse el evento Z.

Una actividad es una acción que se ejecuta mientras un objeto se encuentra en un estado con un tiempo de duración no despreciable. Una actividad se representa en UML mediante el comando **haz/Operación** dentro del diagrama de estado correspondiente. Las actividades pueden ser cíclicas, secuenciales o pueden dar paso a otras transiciones al ser concluidas lo cual se conoce como transición automática.

Los diagramas de estados permiten aplicar reglas de generalización (creándose superestados y subestados) y composición de estados.

TESIS CON
FALLA DE ORIGEN

2.3.7. Diagramas de actividades

Los diagramas de actividad son una variante de los diagramas de estado y permiten representar actividades de forma más directa mostrando una vista dinámica del sistema. Las actividades se representan mediante un rectángulo de estado más achatado y se relacionan por medio de transiciones automáticas, es decir, no hay posibilidad de transiciones internas o disparadas por eventos. Las transiciones se representan como flechas entre actividades y una decisión se dibuja como un diamante que bifurca diversas transiciones. (fig. 2.14)

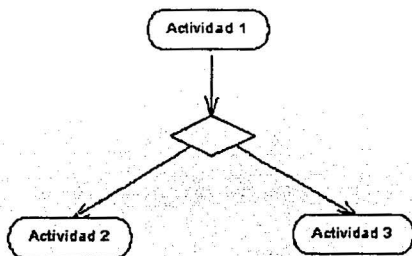


Figura 2.14: Diagrama de actividad

Este tipo de diagramas permiten ejemplificar la sincronización de actividades mediante el uso de barras horizontales para impedir o promover la ejecución de actividades bajo ciertas condiciones temporales. Otra herramienta útil en este tipo de diagramas es el uso de "carriles" para hacer visible la responsabilidad de ciertos objetos en la ejecución de actividades. (fig. 2.15)

2.3.8. Diagramas de componentes

Los diagramas de componentes sirven para representar la estructura del software involucrado en la solución o representación de un fenómeno. Un componente representa cualquier tipo de elemento de una aplicación de software; desde archivos hasta librerías ligadas dinámicamente.

La representación de un componente en UML es el presentado en el diagrama 2.16.

TESIS CON
FALLA DE ORIGEN

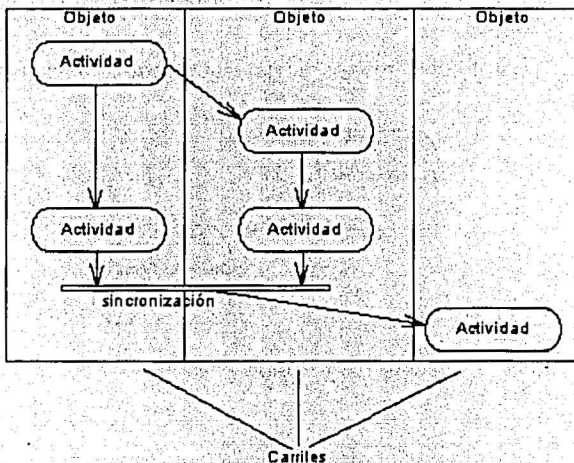


Figura 2.15: Diagrama de actividades con sincronización



Figura 2.16: Componente en UML

Las relaciones entre componentes se dibujan con flechas y representan dependencias a nivel compilación o ejecución del código de programación.

2.3.9. Diagramas de instalación

Los diagramas de instalación¹ muestran la distribución física de los dispositivos de hardware involucrados en un sistema.

¹El término original es "Deployment Diagram", pero al no existir una traducción estándar de términos en este trabajo se decidió utilizar el término "instalación".

Cada elemento de hardware es representado en UML mediante un cubo (fig. 2.17).

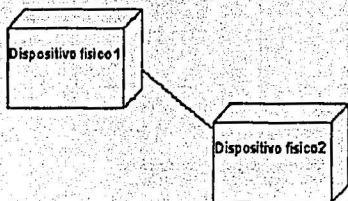


Figura 2.17: Diagrama de instalación

Capítulo 3

Características del almacén moderno

El desarrollo de las tecnologías de la información, la necesidad de hacer más eficientes los procesos y la creciente competitividad en las empresas han fomentado una drástica evolución en los almacenes de hoy. Dicha evolución se ve reflejada en la concepción del almacén no sólo como un repositorio o bodega de mercancías, sino como una rama de la empresa con alcances más amplios, capaz de agregar valor a los productos y de proveer información útil y veraz para la toma de decisiones. Esta tendencia ha fomentado la visualización del almacén no como una carga, sino como una fuente de ventajas competitivas, por lo cual el sector logístico ha adquirido gran relevancia en el enfoque actual de las empresas.

Este capítulo contiene las características más importantes presentes en la mayoría de los almacenes modernos y su intención es servir como introducción al tema del almacenamiento y el proceso de surtido de pedidos en la industria editorial visto en capítulos posteriores.

3.1. Almacenamiento de Mercancías

Las mercancías son la razón de ser de los almacenes y tienen una importancia vital en las empresas. Cada mercancía posee tanto características intrínsecas particulares (volumen, peso, fragilidad, toxicidad, etc), como cualidades asociadas a su frecuencia y magnitud de entrada y salida del recinto. Estas propiedades en conjunto determinan en gran medida los requerimientos de un almacén.

El almacenamiento de mercancías representa una de las principales acti-



vidades de cualquier almacén influyendo directamente en su forma de operar, en su tamaño, en el número de empleados con los que debe contar, en el tipo de labor que éstos deben desempeñar, y en muchas otras características que definen los requisitos de un almacén.

3.1.1. Almacenamiento por unidades

El almacenamiento y manejo de mercancías por medio de unidades de carga estandarizadas se origina en la Segunda Guerra Mundial como resultado de necesidades de optimización de espacio y tiempo. Existen diversos tipos de plataformas para manejo y almacenamiento de mercancías entre las cuales destacan las tarimas. Éstas pueden estar hechas de diversos materiales como plástico, madera o fibra corrugada.

La tarima de madera de 1 x 1.2 metros fue adoptada como estándar en la industria del almacenamiento en los años 70. La adopción de esta unidad trajo muchos beneficios entre los cuales podemos destacar la homogeneización de estructuras de almacenaje y equipo de transporte, así como una consiguiente disminución en los costos.

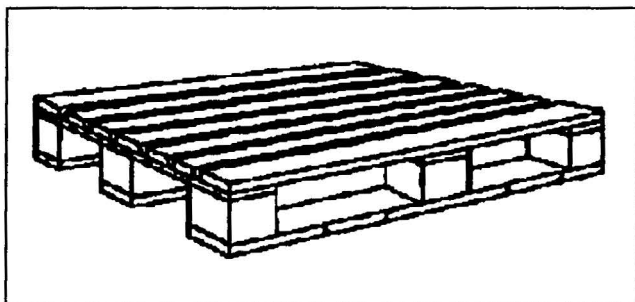


Figura 3.1: Tarima estándar

3.1.2. Almacenamiento especializado

Existen ciertos tipos de mercancía que requieren condiciones especiales para su manejo o almacenamiento, lo cual implica modificaciones en el al-

TESIS CON
FALLA DE ORIGEN

macén.

Los tipos de almacén especializado más comunes son:

El almacén con control de temperatura permite almacenamiento a bajas temperaturas para productos que así lo requieran. Sus costos son mayores debido al equipo requerido y a la dificultad en la operación del almacén.

Los almacenes para materiales peligrosos son requeridos para materiales venenosos, explosivos, radioactivos o inflamables. Requieren altos niveles de capacitación para sus empleados y fuertes medidas de seguridad en su operación.

Los almacenes con administración de pedidos son aquellos que atienden directamente la distribución y entrega de mercancías al cliente. Requieren una mayor cantidad de recursos humanos y tecnológicos por su grado de especialización.

3.1.3. Equipo de almacenamiento

La meta de todo almacén consiste en servir como un medio para movilizar y distribuir mercancías de manera ágil y efectiva. Para lograr esto se requiere el uso de equipo especializado de almacenamiento cuya selección depende de factores como el grado de flexibilidad deseado en el almacén, la forma de manejar la mercancía (por volumen, por paquetes individuales, etc), la naturaleza del edificio de almacenamiento y por supuesto los costos.

Existen diversos tipos de equipo de almacenamiento, entre los cuales destacan:

- Los racks o estantes de almacenamiento son contenedores que permiten ubicar tarimas estándar. Este tipo de equipo permite definir los niveles de altura y profundidad de almacenamiento dependiendo de las necesidades particulares del almacén.

Para realizar una elección adecuada de los niveles de altura y profundidad se debe tomar en cuenta el tipo de mercancía y la frecuencia con que será movida.

Por ejemplo, al tener estantes con muchos niveles de profundidad se obtiene una gran densidad de tarimas con baja eficiencia en su rotación y manejo, mientras que con una gran cantidad de niveles de altura se optimiza el espacio vertical, hay mayor eficiencia en la rotación pero se vuelve necesaria la utilización de equipo especializado de manejo de carga para alcanzar los niveles superiores.

TESIS CON
FALLA DE ORIGEN

- El equipo de carga ligera o picking permite el almacenamiento de pequeñas cantidades de mercancía, con lo cual el transporte y almacenaje se realizan sin la necesidad de equipo especial.
- El equipo de almacenamiento útil consiste en aquellas estructuras que combinan almacenamiento y desplazamiento de mercancía.

Ejemplos de este tipo de estructuras son los estantes que permiten el deslizamiento de tarimas de la parte trasera a la frontal (estantes de flujo de gravedad), parrillas giratorias para recolectar pequeñas cajas (sistemas de carrusel) y bandas, rieles o patines donde se desplazan los productos (sistemas de transportadores).

3.1.4. Cuantificación del espacio de almacenamiento

El espacio de almacenamiento es una medida de la capacidad real del almacén. Para obtener un cálculo aproximado se debe tomar en consideración el área del terreno en el cual se encuentra ubicado el almacén, las características de las estructuras de almacenamiento (niveles de altura y profundidad), número de pasillos y zonas para tráfico de mercancías.

La capacidad real del almacén puede ser comparada con la capacidad ideal (aquella que considera únicamente el espacio cúbico del almacén) para dar una idea de la eficiencia en la capacidad de almacenamiento.

3.2. Manejo de carga

El manejo de la carga constituye otra de las funciones de gran relevancia en el almacén moderno, debido a que las empresas requieren un manejo eficiente de sus productos para poder ser competitivos y agregar valor a sus servicios.

3.2.1. Equipo de transporte

La elección del equipo para la movilización de mercancías depende de factores internos del almacén como el tipo de producto almacenado o la velocidad de tránsito, así como factores externos como los costos, la necesidad de mantenimiento, la capacitación requerida por los empleados y la posibilidad de renovación del equipo.

Tipos de equipo de transporte

Los montacargas son vehículos que facilitan el manejo de mercancía; son el transporte más utilizado en los almacenes modernos y el diseño de la mayoría de ellos está hecho para el manejo de tarimas estándar.

A continuación se presentan los tipos más importantes de montacargas:

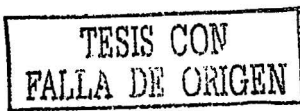
- El montacargas de caminante o "gato para tarimas" representa el tipo más económico de equipo de transporte en el almacén moderno. Es completamente manual y se utiliza para realizar recorridos cortos, ya que su utilidad depende en gran medida de la fuerza del empleado que lo utilice.
- El montacargas de jinete es el tipo más utilizado en los almacenes, ya que es un montacargas completamente mecánico, con gran capacidad de carga y desplazamiento. Los inconvenientes de este tipo de equipo son el tiempo de capacitación que requieren los empleados para su utilización y los costos derivados por compra, reparación, mantenimiento y uso de energía.
- El montacargas de tripulante es un montacargas completamente mecánico que se desplaza mediante una guía, por lo que el operador posee gran visibilidad y libertad de movimiento. Sus desventajas radican en su costo visiblemente mayor y en que este tipo de transporte resulta útil sólo para determinado tipo de mercancías y almacenes.

3.2.2. Recepción y Embarque

La recepción es el proceso de entrada de mercancía en el almacén; el embarque representa la salida de productos del almacén. Ambas actividades son cruciales ya que implican una transferencia de propiedad y responsabilidad entre transportista y almacén.

La recepción implica diversos pasos dependiendo del almacén y del material a recibir, entre ellos:

1. Llegada del transportista
2. Descarga de la mercancía
3. (Verificación) Existen varias maneras de realizar la verificación de mercancía en recepción y embarque entre las cuales destacan la utilización de códigos de barras debido a que provee una mayor exactitud que cualquier proceso de identificación manual.



4. (Colocación y almacenaje de material) La decisión sobre la ubicación de las mercancías después de la recepción puede ser tomada por la persona que ejecuta la acción (generalmente el montacarguista) o por una persona encargada de la planeación del almacén. La diferencia radica en que el montacarguista tratará de minimizar tiempos y recorridos sin importarle el espacio de almacenamiento, mientras que un planeador tomará en cuenta más elementos en su decisión.

3.2.3. Surtido o Recolectión de Órdenes

Las actividades de surtido y embarque representan las funciones más costosas y críticas en la operación del almacén. Este hecho radica en que la efectividad y utilidad de un almacén proviene directamente de estas dos actividades, es decir, su realización eficiente equivale a un buen funcionamiento del almacén.

La actividad de surtido implica la recolección de mercancías en volúmenes y zonas variables dependiendo de la orden o requerimiento particular de surtido.

Existen tres métodos básicos de recolección de mercancía:

- Recolectión de carga unitaria, en la cual la cantidad mínima a recolectar es una tarima completa .
- Recolectión por cajas, en la cual la unidad ínfima de recolección es una caja completa.
- Recolectión de caja abierta, en la cual se permite tomar artículos individuales de las cajas.

El surtido puede ser realizado de forma manual, automática, asistida por equipo especial o una combinación de éstos y se puede dividir en varias categorías:

La recolección moderada es aquella en la cual un recolector o surtidor toma una orden y la atiende de principio a fin.

La recolección por lotes implica que un surtidor toma un lote de órdenes, obtiene el total de cada artículo en el lote y recolecta estas cantidades para su posterior separación en órdenes individuales.

La recolección por zonas generalmente es llevada a cabo por más de una persona y consiste en la distribución de la orden en tantas partes como zonas del almacén en que se encuentre el material.

La recolección por oleadas consiste en la agrupación y surtido de órdenes de acuerdo a criterios específicos (transportista, volumen de orden, tipo de mercancía, fecha de entrega, etc).

3.3. Sistemas de Información

Los avances en la campo de las tecnologías de la información han sido de una importancia y velocidad impresionante en los últimos años. Los administradores de almacenes han visto cómo la mejora y disminución de costos en los equipos de cómputo han hecho posible como nunca antes la implementación de sistemas de información para la automatización o mejora de procesos en los almacenes.

3.3.1. Uso de la Información en el Almacén

La información se ha vuelto uno de los activos más importantes en los almacenes, por lo cual la calidad en los sistemas de información se vuelve un factor que proporciona ventajas competitivas respecto a otros almacenes.

El primer paso a considerar sobre el uso de sistemas de información en un almacén es el tipo de funciones que se requiere desempeñar y las actividades sobre las que se quiere obtener información y tener control. La identificación de estas necesidades servirá para decidir la compra de software o el desarrollo de sistemas propios. Además se deben considerar los costos que el sistema tendrá, la capacitación que requerirán los empleados y el soporte que tendrá el sistema si se presentan modificaciones a lo largo del tiempo.

3.3.2. Sistemas de Localización

Los sistemas de localización sirven para reducir el tiempo de surtido de órdenes, para tener información precisa y detallada sobre los productos en el almacén y para poder realizar una planeación adecuada sobre recorridos, ubicaciones y procesos. Como ejemplo de un sistema de localización se puede considerar el uso de lectores de códigos de barras para la identificación y manejo de mercancía.

La necesidad de un sistema de localización depende del número de productos manejados y de su disposición dentro del almacén.

Un sistema de localización provee una gran cantidad de ventajas a los administradores de un almacén, como:

- Dinamismo en el uso de espacio debido a la posibilidad de agrupar un producto existente en varias ubicaciones o abrir nuevos espacios para almacenar un producto nuevo.
- Ahorro de tiempo debido a que se eliminan las búsquedas visuales de mercancía al surtir o almacenar productos.
- Un sistema de localización permite un surtido más ágil y una selección de ubicaciones estratégicas de surtido para optimizar los recorridos.
- El control por primeras entradas primeras salidas (First In First Out o FIFO) se puede realizar mediante un sistema de localización indicando las ubicaciones con lotes de mercancía más antiguos.
- La cuarentena de productos por inventario, verificación u obsolescencia puede ser respaldada por un sistema de localización.

Un sistema de localización eficiente requiere de una buena comunicación, disciplina y corrección inmediata de errores. Aunque muchos almacenes operan sin un localizador formal, la mayoría de ellos incrementan su eficiencia cuando un sistema de este tipo es implementado adecuadamente.

3.3.3. Identificación Electrónica

La identificación electrónica ha mostrado un gran potencial dentro de la industria del almacenamiento, siendo el código de barras el método de identificación más utilizado.

Códigos de barras

El código de barras consiste en un grupo de barras y espacios dispuestos de forma tal que puedan ser leídos por una máquina. Los códigos de barras pueden ser agregados por el fabricante del empaque o pueden imprimirse en etiquetas y ser usados directamente en el almacén para identificar todo tipo de cosas. Una vez aplicado, el código de barras puede ser leído por una gran variedad de lectores ópticos o "scanners".

El código de barras mejora la operación del almacén de muchas maneras debido a que la lectura y copiado de datos es más rápido y exacto que cualquier proceso manual, minimizando tanto los tiempos de procesamiento de información como los errores de recepción, almacenamiento, embarque e inventario. El uso de códigos de barras permite también transmisión de instrucciones a los empleados mediante el lector óptico y la realización de

actividades "en línea", es decir, con retroalimentación en tiempo real entre el sistema y los empleados.

Cabe hacer notar que la identificación por medio de códigos de barras no eliminará el caos existente en un almacén administrado deficientemente, por lo que el éxito en la operación de un almacén depende en mayor medida de la existencia de una buena disciplina y del entendimiento de responsabilidades por parte de cada empleado que de la efectividad o velocidad de cualquier sistema de información.

TESIS CON
FALLA DE ORIGEN

Parte II
Aplicaciones

TESIS CON
FALLA DE ORIGEN

Capítulo 4

Características del almacén editorial

En la actualidad, los almacenes han cobrado mucha relevancia debido a que representan una gran oportunidad de negocios, así como una fuente poco explotada de información.

Dentro de la industria editorial este fenómeno no es la excepción, ya que es un sector muy competitivo en el cual los almacenes editoriales representan una potencial ventaja competitiva. Además, la optimización y mejora de procesos se vuelve un reto debido a las características específicas de almacenaje que el ramo requiere.

Este capítulo tiene el fin de presentar al lector las características específicas del almacén editorial para el cual se realizó el sistema y cuyo módulo de surtido de pedidos es explicado en el capítulo siguiente desde la particular óptica de la Programación Extrema y el Lenguaje de Modelado Unificado. Se presentan además, una serie de datos específicos del almacén para dar una idea más clara y cercana del problema abordado en el capítulo siguiente.

4.1. Títulos y ejemplares

En un almacén editorial el elemento preponderante son los libros. En el presente trabajo designaremos con el término ejemplar a un libro cualquiera, mientras que el término título nos servirá para hacer referencia a una agrupación de libros cuya característica consiste en poseer el mismo título.

TESIS CON
FALLA DE ORIGEN

4.2. Datos Generales

El almacén editorial utilizado como base para el presente trabajo tiene las siguientes características:¹

Superficie total del almacén*: 14600 m²

Superficie de la nave del almacén: 7000 m²

Superficie destinada a oficinas: 316 m²

Superficie para expansión del almacén: 2800 m²

Altura máxima del almacén: 8.1 m

Altura mínima del almacén: 7.12 m

Empleados en almacén*: 80

Número de montacargas: 5

Empleados administrativos: 22

Promedio de ejemplares en el almacén*: 16,000,000

Títulos en el almacén*: 7,400

Promedio de pedidos al día*: 125

Promedio de ejemplares por pedido*: 650

Promedio de títulos por pedido*: 15

4.3. Tipos de libros

En el almacén editorial utilizado como base para el presente trabajo existen dos tipos principales de libros con características bien diferenciadas, lo cual se refleja en la existencia de dos grandes zonas en el almacén destinadas al manejo de cada uno de los siguientes tipos de libros:

Libros de Texto: Son los libros que se utilizan como herramienta de apoyo didáctico en las escuelas. El número de títulos es reducido, pero su producción y manejo en el almacén se realiza en grandes volúmenes en una temporada específica del año. La zona del almacén donde este tipo de libros se almacena es conocida como zona de texto.

¹Los datos con * son aproximados.

Libros de Obra General: Dentro de esta categoría se incluyen aquellos libros cuyos temas son de interés general. A diferencia de los libros de texto, la variedad de títulos es muy amplia, con una producción y manejo en pequeños volúmenes de manera constante a lo largo del año. La zona del almacén destinada al manejo de este tipo de libros se conoce como zona de obra general.

4.4. Paquetes

Un paquete es una agrupación de libros mediante algún contenedor físico como por ejemplo una tarima o un empaque de cartón. Los libros son agrupados en paquetes para facilitar su transporte, contabilizar rápidamente los libros, y en el futuro, estandarizar el número de ejemplares por paquete para poder redondear los pedidos a paquetes cerrados.

4.5. Posiciones

Una posición en el almacén es un espacio físico bien definido dentro de las instalaciones, el cual funciona como repositorio de libros. Como ejemplos de posiciones en el almacén podemos considerar a las mesas de trabajo, a una zona delimitada en el piso del almacén mediante marcas o a posiciones contenidas en equipo de almacenamiento como racks o estantes.

4.6. Equipo de almacenamiento

A continuación presentaremos una descripción de las diversas zonas existentes en el almacén editorial desde el punto de vista del tipo de equipo de almacenamiento utilizado.

El almacén cuenta con una zona de Stock mayor o grandes volúmenes equipada con 3600 posiciones de almacenamiento agrupadas en 6 pasillos, cada uno de los cuales consta de 50 racks de 4 niveles de altura y 3 niveles de profundidad. (fig.4.1)

Un pasillo de esta zona se encuentra destinado a libros de obra general mientras que el resto es usado para almacenar libros de texto. Debido al manejo de grandes volúmenes de libros, en esta sección del almacén es necesario el uso de montacargas, por lo cual el acceso de personas a pie está restringido para evitar riesgos.

La zona de Picking o pequeños volúmenes consta de 2688 posiciones agrupadas en 4 pasillos, cada uno de los cuales consta de 96 estantes de

TESIS CON
FALLA DE ORIGEN

7 niveles de altura y 1 de profundidad para almacenaje de libros de obra general. Además existen 672 posiciones agrupadas en 8 pasillos, cada uno de los cuales contiene 28 anaqueles de 3 niveles de altura y 1 de profundidad para el almacenaje de libros de texto.

Existe a su vez una zona para la realización de varios procesos entre los que destacan limpieza, empaçado, sellado y plastificado de libros. Las primeras tres actividades se llevan a cabo en mesas que son usadas conforme a demanda, mientras que el plastificado es llevado a cabo por maquinaria especializada. (fig.4.1)

4.7. Equipo de transporte

El almacén editorial cuenta con una base de 5 montacargas de jinete (la cual puede ser ampliada en temporadas de mayor actividad mediante la renta de equipo), los cuales son utilizados para la realización de las labores más pesadas y difíciles en el almacén como son:

- Manejo de cargas pesadas como tarimas de libros o bobinas de papel.
- Alcance de ubicaciones en el almacén de gran altura.
- Transporte de mercancías a gran velocidad en el almacén.

La capacitación y especialización requerida para el manejo de los montacargas de jinete hace que el número de empleados encargados de su manejo sea limitado y se encuentre dedicado completamente a esa actividad.

Para las labores de transporte y manejo de carga menos difíciles el almacén editorial cuenta con una base de aproximadamente 10 montacargas de caminante, los cuales pueden ser utilizados conforme a demanda por cualquier empleado del almacén.

4.8. Empleados del almacén

El almacén cuenta con dos tipos principales de empleados:

Empleados administrativos: son todos aquellos empleados que forman parte del área administrativa del almacén. Entre ellos se encuentran gerentes, supervisores, secretarías, etc.

Empleados no administrativos: son todos aquellos empleados que laboran dentro de la nave del almacén. Dentro de estos empleados existe una subdivisión:

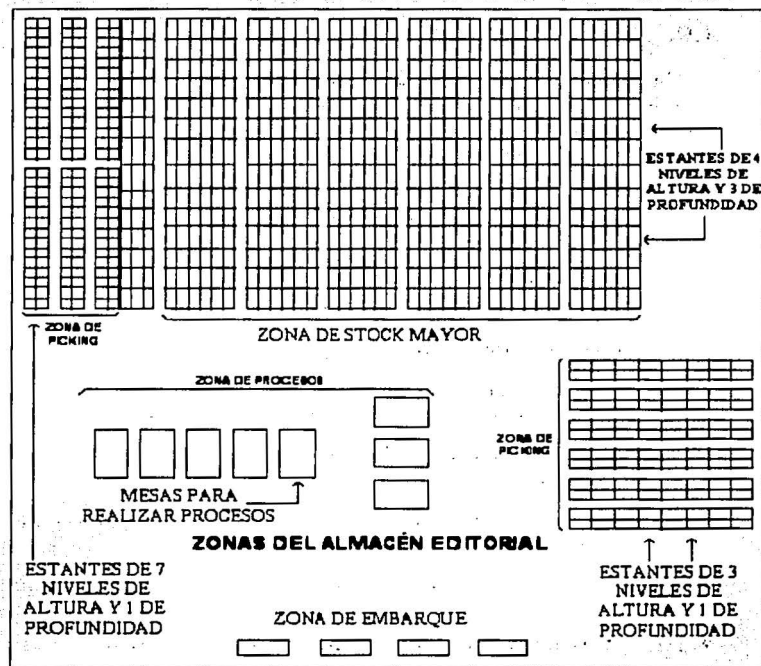


Figura 4.1: Distribución de zonas en el almacén editorial.

- Empleados a pie. Son aquellos que realizan labores de surtido a pie o mediante el uso de patines, así como diferentes procesos como empaque, plastificado de libros, etc.
- Montacarguistas. Es el grupo de empleados no administrativos que se encuentra a cargo del manejo de los montacargas en el almacén de manera exclusiva.

TESIS CON
FALLA DE ORIGEN

4.9. Pedidos

Un pedido es una solicitud de libros por parte de un cliente a la empresa editorial. Un pedido pasa por varias etapas o "estados" desde su solicitud por parte del cliente hasta el momento en que es entregado. Las diferentes etapas por las que pasa un pedido son las siguientes:

1. Capturado: el pedido es capturado por el Departamento de Ventas de la empresa editorial y es susceptible de sufrir modificaciones en títulos y cantidades requeridas.
2. Liberado: el pedido es remitido al almacén por el área de ventas y se encuentra en espera de ser atendido.
3. Generado: el pedido es asignado a personal del almacén para su atención.
4. Surtido: un pedido se encuentra en este estado una vez que ha sido concluida la recolección de los libros correspondientes.
5. Empacado: se ha concluido completamente el empaque de los libros anteriormente surtidos.
6. Embarcado: han sido adjuntados los documentos correspondientes y el pedido ha sido llevado a un transporte.
7. Entregado: el pedido se encuentra en manos del cliente.

4.10. Control de posiciones e inventarios

La existencia de un sistema de localización en el almacén se vuelve clave cuando se requiere optimizar los procesos en el almacén y minimizar los errores.

Una de las metas del sistema realizado para el almacén editorial consistió en la utilización de lectores ópticos de códigos de barras para identificación electrónica². (fig. 4.2)

Estos lectores ópticos se utilizan conjuntamente con códigos de barras impresos en etiquetas, las cuales pueden ser adheridas a prácticamente cualquier cosa para su identificación.

El tipo de actividad desarrollada en el almacén editorial hace que la construcción de software para la administración, monitoreo y control de las

²A partir de este momento se usará el término "lector óptico" para hacer referencia a los lectores ópticos de códigos de barras.

actividades en el almacén esté orientada a dispositivos de tipo portátil como es el caso de estos lectores ópticos. En un gran esfuerzo por impulsar el proyecto de sistematización del almacén la compañía editorial adquirió una cantidad aproximada de 40 lectores ópticos, a los cuales se les debe construir funciones propias para su operación en el almacén.



Figura 4.2: Lector óptico de códigos de barras

4.11. El proceso de surtido

En el siguiente capítulo se explicará desde una perspectiva muy particular una sección de un sistema para el manejo de un almacén editorial. La sección del sistema que se expone en el siguiente capítulo es la relacionada al surtido de pedidos en el almacén. Es por esto que aquí se da una visión general de este proceso para que el lector tenga una idea clara del tema del siguiente capítulo.

La secuencia de surtido de pedidos en el almacén editorial consta de los siguientes pasos :

1. Un cliente realiza contacto con el departamento de ventas de la empresa editorial para realizar una solicitud de libros.
2. El pedido de libros solicitado por el cliente es capturado mediante un sistema comercial ya existente en la empresa editorial.
3. Después de verificar que los títulos, cantidades requeridas, precios, descuentos y demás datos concernientes al pedido sean correctos, el Departamento de Ventas de la empresa editorial gira el pedido al almacén editorial.
4. Una vez que el pedido se encuentra en el almacén, un empleado del almacén (generalmente un supervisor) selecciona aquellos pedidos que se

van a surtir de acuerdo a criterios como volúmen del pedido, prioridad del pedido o importancia del cliente, generando así una lista de surtido individual (orden con un sólo pedido para realizar una recolección moderada) o una lista de surtido global (orden con varios pedidos para realizar una recolección por lotes, ver 3.2.3)

5. Una vez creada la lista de surtido, se asignan empleados a pie (ver 4.8) del almacén para la recolección y traslado de los libros.
6. Los empleados a pie recolectan libros de aquellas posiciones cuya altura y profundidad les resulten accesibles, solicitando verbalmente a los montacarguistas aquellos libros que no les resultan accesibles.
7. Los libros recolectados se agrupan en una sección del almacén diseñada para tal fin en donde se les realizan procesos como empaque, flejado, etc.
8. Una vez concluidos los procesos posteriores a su recolección, los libros se encuentran listos para ser embarcados y transportados a su destinatario.

El surtido de pedidos ocupa un lugar preponderante en las actividades cotidianas de un almacén debido a que existe una relación directa entre la calidad y rapidez en el servicio del almacén y la eficiencia con que se realiza esta actividad, ya que mientras más rápido y con la menor cantidad de errores sea surtido y entregado un pedido de libros al cliente, éste se sentirá más satisfecho y a gusto con la empresa.

Capítulo 5

El Surtido bajo la perspectiva XP/UML

La Programación Extrema (XP) y el Lenguaje de Modelado Unificado (UML) son aproximaciones al desarrollo de software aparecidas recientemente con el objetivo de mejorar esta actividad desde perspectivas diferentes.

Este capítulo muestra el avance en el desarrollo del módulo de surtido de pedidos de un sistema realizado para un almacén editorial. Dicha evolución es mostrada desde la perspectiva de ciertas prácticas de XP, así como mediante diagramas UML en el diseño para mostrar su uso en un caso concreto y hacer notar la viabilidad o sencillez de su aplicación.

El módulo de surtido de pedidos fue elegido como ejemplo en el presente trabajo debido a que es un proceso de gran importancia en el almacén con una gran cantidad de elementos que permiten ejemplificar las prácticas de XP, utilizar diagramas UML y dar una buena idea de la operación del almacén editorial.

5.1. Etapa inicial del proyecto

Inicialmente, todo proyecto de software requiere una etapa de definición y especificación de objetivos, a partir de los cuales es posible comenzar a trabajar. En esta etapa XP propone la aplicación de la práctica denominada "juego de planeación" (ver-1.2.1).

Para el módulo de surtido de pedidos se llevaron a cabo diversas reuniones con el cliente para establecer y detallar los objetivos que el software debe cumplir: por lo general las reuniones eran largas y resultaba difícil

TESIS CON
FALLA DE ORIGEN

obtener ideas claras acerca de los objetivos y alcances del software.

En base a la serie de reuniones iniciales con el cliente se presenta la siguiente lista de objetivos generales que el software para el almacén editorial debe cubrir:

- La actividad de surtido debe llevarse a cabo por medio del sistema con información en tiempo real.
- La información referente al contenido de las ubicaciones de almacenamiento y de los títulos existentes en el almacén debe encontrarse disponible en el sistema para su consulta en cualquier momento.
- La actividad de surtido de pedidos se debe llevar a cabo lo más rápidamente posible minimizando el número de errores.
- La información relacionada al surtido de pedidos debe ser almacenada para su posterior utilización para toma de decisiones, archivo histórico o fuente de estadísticas.
- El módulo de surtido de pedidos debe funcionar como herramienta de monitoreo y control de la actividad en el almacén.

5.2. Primera iteración

A partir de las reuniones entre el equipo de desarrollo y el equipo de negocios a cargo del almacén editorial se realizó la elección de los primeros requerimientos de usuario a desarrollar.

Acorde con el valor de simplicidad de XP (ver 1.1), la elección de los primeros requerimientos para el módulo de surtido de pedidos fue elegida de forma tal que el sistema resultara sencillo y que al cabo de la iteración pudiera ser utilizado como herramienta de apoyo y consulta.

En esta iteración o ciclo corto de desarrollo se pretende llevar a cabo todas las etapas "clásicas" del desarrollo de sistemas, permitiendo así que la actividad de surtido se realice de la misma manera en que se venía llevando a cabo con la paulatina integración del software a las actividades del almacén.

A continuación se detallan los requerimientos de usuario considerados para esta iteración:

1. Se requiere un componente que permita administrar los pedidos que el Área de Ventas remite al almacén editorial.
2. Un componente para administrar las posiciones en el almacén.

3. Como primeras opciones para desarrollar en los lectores ópticos se eligieron "Inventario", "Ajuste de posición" y "Traslado" ya que son aquellas opciones consideradas más útiles bajo el punto de vista del equipo de negocios de acuerdo a lo establecido en el juego de planeación de XP (ver 1.2.1).

Con la opción "Inventario" se va a lograr dar de alta nuevas posiciones en el almacén.

"Ajuste de Posición" va a permitir realizar modificaciones a la información en el sistema referente al contenido de una posición (número de ejemplares de cada título existentes en la posición).

"Traslado" permitirá realizar movimientos de libros entre posiciones.

4. Para que el sistema comience a ser utilizado y sirva de apoyo al entregar la primera etapa de desarrollo (práctica de pequeñas entregas, ver 1.2.2) se tomó la decisión de realizar un par de consultas que ayudarán a cumplir uno de los objetivos iniciales del sistema consistentes en saber dónde se encuentran los libros y en qué cantidad: "Consulta de título" y "Consulta de posición"

En la consulta de título el usuario preguntará sobre un título específico obteniendo información sobre aquellas posiciones del almacén en que éste se encuentra.

La consulta de posición permitirá al usuario conocer la información en el sistema relativa al contenido de una posición dada.

5.2.1. Desarrollo de la iteración

Administración de pedidos

La práctica de realización de pequeñas entregas de XP propone construir primero aquellos componentes que resulten más útiles para el usuario. Es por esto que el primer requerimiento de usuario a realizar, sea un componente que permita seleccionar los pedidos a surtir, ya que este elemento resulta fundamental para los componentes que se pretenden desarrollar posteriormente. Esta importancia se deriva de que todo el proceso de surtido depende de los pedidos remitidos al almacén por el Departamento de Ventas. El componente a construir para seleccionar pedidos será una pantalla que permita monitorear y seleccionar los pedidos a surtir, a la cual llamaremos Pantalla de administración de pedidos.

TESIS CON
FALLA DE ORIGEN

El diagrama UML de estado 5.1 muestra los diferentes estados de un pedido explicados en la sección 4.9. El objetivo de la pantalla de administración de pedidos consiste en cambiar el estado de un pedido "liberado" a "generado", es decir, el pedido deja de estar en espera de ser atendido y es asignado a personal del almacén para ser surtido.

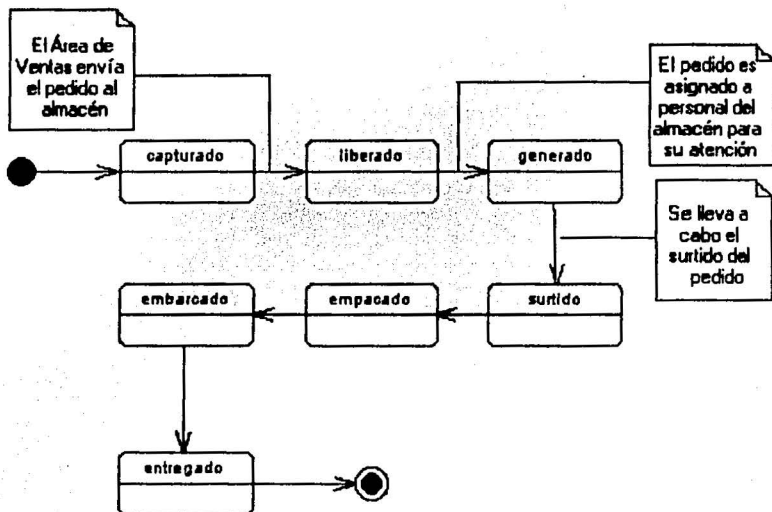


Figura 5.1: Diagrama UML de los estados de un pedido

Para que un pedido sea visualizado en la pantalla debe ser capturado por el Departamento de Ventas de la compañía editorial y los títulos, cantidades y precios de los libros deben ser validados antes de enviar el pedido al almacén editorial como se observa en el diagrama UML de caso de uso 5.2.

En primer término un pedido es capturado por el Área de Ventas la cual realiza además validaciones de títulos, cantidades y precios de los libros antes de liberar el pedido al almacén editorial como se observa en el siguiente diagrama de caso de uso. (fig. 5.2)

La pantalla de administración de pedidos permitirá a los empleados del

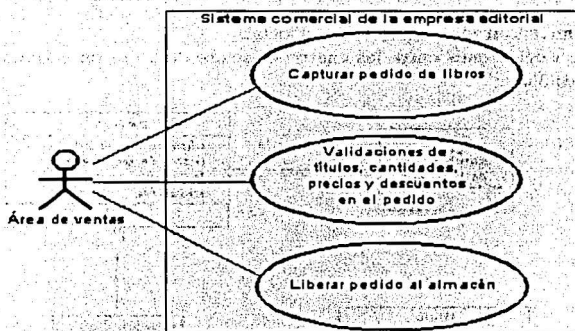


Figura 5.2: Caso de uso de la liberación de pedidos

almacén realizar las actividades descritas en el diagrama de caso de uso 5.3.

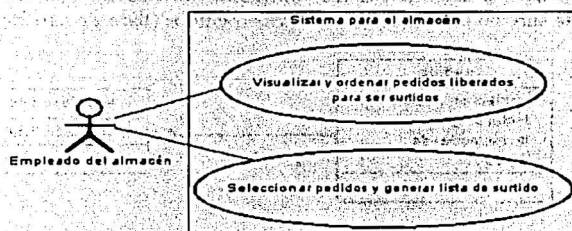


Figura 5.3: Caso de uso de la pantalla de administración de pedidos

Para la realización de esta pantalla requerimos una clase llamada pedido, así como otras tres clases llamadas cliente, título y pedido_detalle.

La clase pedido representará el encabezado y los datos generales de un pedido, mientras que la clase pedido_detalle representará el detalle o desglose de los artículos requeridos en el pedido. La clase cliente representará a todo aquel que solicite libros a la empresa editorial y la clase título representará un

título particular existente en la editorial.

Las asociaciones entre las cuatro clases se detallan en el diagrama UML de clases 5.4.

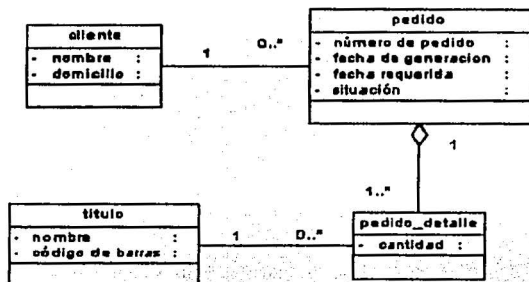


Figura 5.4: Clases involucradas en un pedido

El concepto lista de surtido representa un concentrador de un pedido (lista de surtido individual) o de varios pedidos (lista de surtido global).

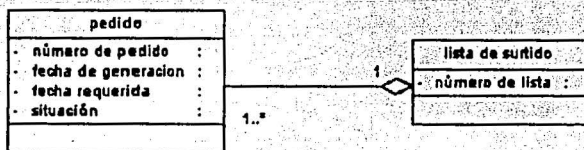


Figura 5.5: Relación entre pedido y lista de surtido

La pantalla de administración de pedidos (fig. 5.6) crea una lista de surtido cuando uno o varios pedidos son seleccionados en el lado izquierdo y arrastrados al lado derecho de la pantalla provocando su transición de la situación liberado a la situación generado.

Esta pantalla permite ordenar los pedidos de acuerdo a ciertos criterios de importancia en el almacén como cantidad de ejemplares requeridos (cantidad), fecha en que se requiere que el pedido sea entregado (fecha requerida) y cliente que solicita los libros (cliente).

Sistema General de Órdenes, Logística y Distribución

Archivo Edición Inventarios Facturación Cuentas por cobrar Contabilidad Pack Administración Ventana 2

Proceso de generación

Generación

Lista global

Pedidos liberados				Pedidos generados				
Pedido	Cantidad	Cliente	Fecha recu	Lista global	Lista	Cantidad	Cliente	Fecha
60545	577	RIVAS DE SANTIAGO MARIA	2003/04/28	51263	50181	40	CALDERON JUAN DE DIOS	2003/4
60547	43	GALLEGOS MARTINEZ RODO	2003/05/01	51513	60487	58	LIBRERIAS GONVILL S.A. DE C.V	2003/4
60548	100	ROUZAUD FISCHER ARTURO	2003/05/04	51574	60544	3	PROMOCIONES INFANTIL	2003/4

Listo

Figura 5.6: Pantalla de generación de pedidos

Administración de posiciones

Una vez que existe una forma de administrar los pedidos que el Área de Ventas remite al almacén editorial es hora de comenzar a dar forma a la representación física del almacén dentro del sistema. Siguiendo el valor de simplicidad de XP, comenzaremos con la unidad básica de almacenamiento en el almacén editorial: las posiciones.

Para administrar las posiciones en el almacén editorial se realizará una pantalla que permita visualizar las posiciones dadas de alta en el sistema, así como insertar, modificar o eliminar posiciones, lo cual requiere la creación de una clase llamada posición.

En este ciclo corto de desarrollo se llegó a la conclusión de que estas son las características más importantes de una posición (atributos de la clase posición, fig. 5.7):

- Clave: clave o identificador exclusivo de la posición.
- Altura: nivel de altura de la posición dependiendo del tipo de equipo

posición	
• clave	:
• altura	:
• profundidad	:

Figura 5.7: Clase posición

de almacenamiento que se trate(ver 4.6).

- **Profundidad:** nivel de profundidad de la posición de acuerdo al equipo de almacenamiento utilizado (ver 4.6).

La pantalla de administración de posiciones resultante(fig. 5.8) es sencilla, pero permite dar de alta todas las posiciones existentes en el sistema y conocer la altura y profundidad de cualquier posición, lo cual resulta útil para saber si se requiere equipo especializado de transporte para introducir o extraer libros de la posición.

El nexo entre las posiciones físicas del almacén editorial y las posiciones dadas de alta en el sistema será una etiqueta con el código de barras de la clave de la posición que será adherida en un lugar visible de la posición. Las etiquetas tendrán impresa la clave de la posición tanto en letras como en código de barras.

Ajuste de posiciones

Una vez creada la clase posición y la pantalla de administración de posiciones el siguiente paso consiste en poder dar de alta los títulos y cantidades existentes en cualquier posición del almacén.

En la reunión de planeación para la presente iteración se acordó realizar las opciones "Inventario" y "Ajuste de posición" debido a que se requiere comenzar a catalogar y dar de alta el contenido de cada una de las posiciones en el almacén editorial.

Para el caso de las posiciones se le realizaron consultas al cliente (práctica de cliente en el sitio de desarrollo de XP (ver 1.2.11) para saber si existía alguna limitante en cuanto al número de títulos en una posición, a lo cual el equipo de negocios respondió que una posición no debe tener limitaciones en cuanto al número de títulos existentes.

Para poder construir las opciones en el lector óptico y reflejar a la vez lo que observamos en el almacén (posiciones que pueden contener al mismo

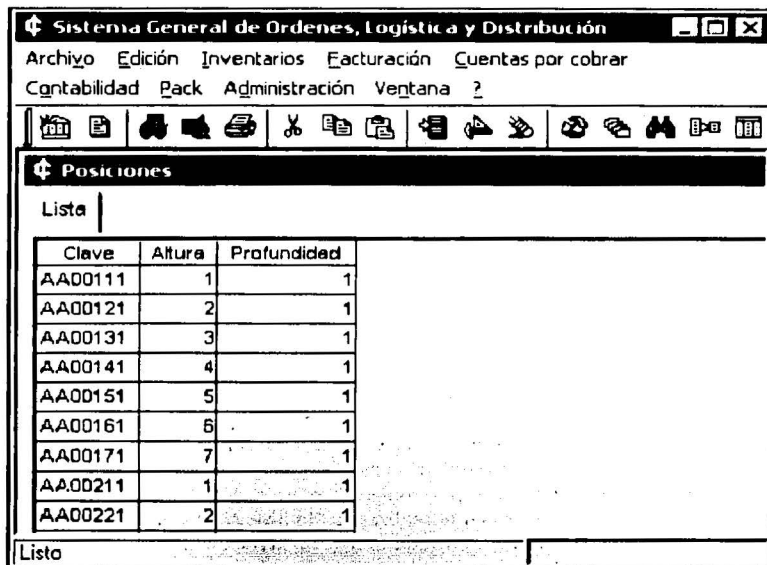


Figura 5.8: Pantalla de administración de posiciones

tiempo cajas, tarimas o libros sueltos) se concibió el concepto paquete (ver 4.4) como medio para agrupar libros. De esta manera, las posiciones contienen elementos más pequeños llamados paquetes, donde cada paquete es un contenedor de libros con un único título.

La relación existente entre títulos, paquetes y posiciones se detalla en el diagrama UML de asociación 5.9.

El diseño reflejado en el diagrama 5.9 tiene las siguientes consecuencias:

- Todo paquete debe hallarse en una única posición.
- Cada posición posee un paquete por cada título que se encuentre en ella.
- Una vez que un paquete tiene cantidad cero, carece de utilidad como

TESIS CON
FALLA DE ORIGEN

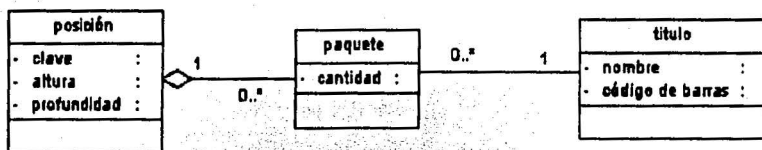


Figura 5.9: Asociación de los paquetes con títulos y posiciones

agrupador y es eliminado.

La opción "Inventario" permite dar de alta una nueva posición en el almacén, así como su contenido (títulos y cantidades de cada título dentro de la posición, ver fig. 5.10).

La opción "Ajuste de posición" permite informar al sistema sobre cambios en el contenido de una posición registrados a raíz del ingreso o salida de libros o registro erróneo del contenido de la posición (fig. 5.10).

La realización de dos opciones independientes para el registro del contenido de una posición fue por solicitud expresa del cliente, el cual deseaba tener un mayor control en la realización de inventarios por parte de los empleados y consideraba de utilidad en esta iteración la creación de las dos opciones (ver 1.2.2).

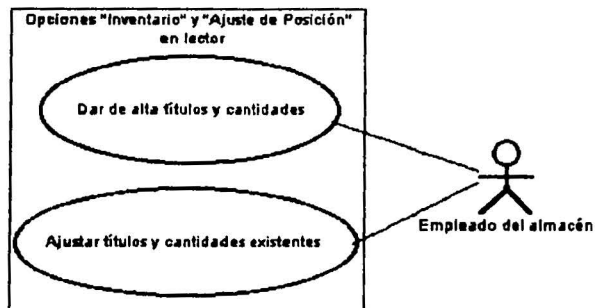


Figura 5.10: Caso de uso para el alta o ajuste de títulos y cantidades

Consulta de posición

Para que el sistema pueda apoyar como herramienta de consulta la labor de surtido en el almacén editorial, en el juego de planeación se eligió como requerimiento de usuario que el sistema pueda informar a los empleados del almacén sobre el contenido de las posiciones.

Es por esto que se ideó una consulta en el lector óptico en la cual se solicite una posición y se muestre su contenido (títulos y cantidad de ejemplares, fig. 5.11).

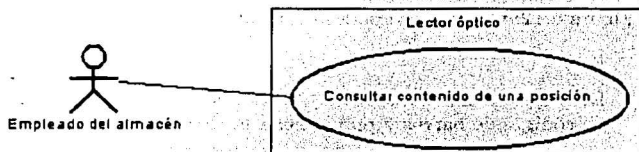


Figura 5.11: Caso de uso para la consulta de posición en el lector óptico

La consulta de posición en el lector óptico se basa en el siguiente algoritmo:

1. El usuario ingresa a la opción "Consulta de posición" en el lector óptico de códigos de barras y proporciona la clave de la posición a consultar.
2. Se realiza una búsqueda de todos aquellos paquetes asociados a la posición en cuestión.
3. Una vez encontrados los paquetes, se realiza una lista de las cantidades y títulos de todos los paquetes pertenecientes a la posición.
4. La lista es mostrada en la pantalla del lector óptico.

Consulta de título

En el juego de planeación se consideró como requerimiento de usuario el hecho de que los empleados del almacén puedan saber en qué posiciones del almacén se encuentra algún título específico. Para cubrir este requerimiento de usuario se realizó una consulta en el lector óptico que solicita la clave de un título específico y entrega la(s) clave(s) de aquellas posiciones que posean el título.

La consulta de título en el lector óptico sigue el siguiente algoritmo:

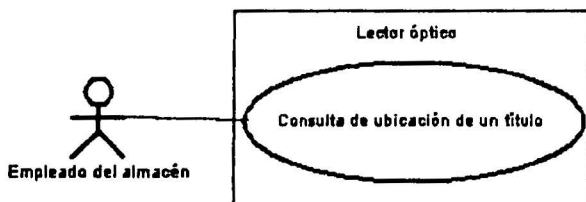


Figura 5.12: Caso de uso para la consulta de título en el lector óptico

1. El usuario ingresa a la opción "Consulta de título" en el lector óptico de códigos de barras y proporciona la clave del título a buscar.
2. Se realiza una búsqueda de todos aquellos paquetes que contengan el título especificado.
3. Una vez encontrados los paquetes, se realiza una lista de aquellas posiciones que los contienen, junto con la cantidad existente del título en cada posición.
4. La lista de posiciones y existencias es desplegada en el lector óptico al usuario.

Traslado

Al comienzo del juego de planeación se consideró la construcción de una opción para realizar traslados de libros entre posiciones del almacén, pero por el momento es posible simular esta operación mediante la opción ajuste de posición (disminuyendo los títulos y cantidades de una posición y aumentando esos mismos títulos y cantidades en otra). Así, siguiendo la práctica de pequeñas entregas de XP (ver. 1.2.2) en la cual cada entrega debe contener aquel conjunto de funcionalidades que resulte más útil, la opción "Traslado" se desechó de la presente iteración y podrá ser incluida en otro momento si es que se considera necesario.

5.2.2. Fin de la iteración

En el diagrama UML de clase 5.13 se puede observar un panorama de lo realizado en la primera iteración.

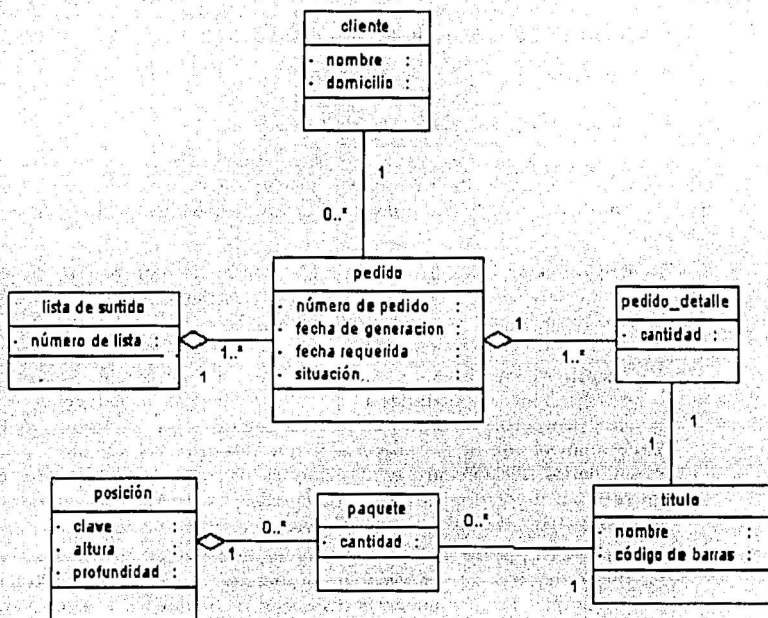


Figura 5.13: Diagrama de clase para la primera iteración

Acorde con la práctica de pequeñas entregas de XP, al concluir el desarrollo de los requerimientos de usuario relativos a esta iteración se entregó al cliente el software para que comience a ser utilizado y probado, lo cual da confianza al cliente respecto al avance del proyecto y fomenta la comunicación y retroalimentación entre equipo de negocios y equipo de desarrollo.

En este momento, el sistema cuenta con las siguientes opciones:

- Pantalla de administración de pedidos
- Pantalla de administración de posiciones
- Opción Inventario y Ajuste de Posición en el lector óptico

- Consulta de Posición en el lector óptico
- Consulta de Título en el lector óptico

5.2.3. Acerca de la planeación en XP

En este apartado hablaremos acerca de algunos puntos concretos de XP respecto a la planeación del proyecto y su comparación con la experiencia vivida en el desarrollo del módulo de surtido de pedidos para el almacén editorial.

- Dentro de la práctica de XP llamada "juego de planeación", la definición de objetivos y de requerimientos por parte de los usuarios no resultó para nada sencilla debido principalmente a que el cliente no sabía exactamente cuáles eran sus necesidades respecto al software.

XP asume que en las reuniones de planeación el equipo de negocios conoce bien sus necesidades respecto al software, lo cual en el caso de este proyecto no resultó ser cierto debido a que las personas que integraban el equipo de negocios y que discutían los alcances y funcionalidades del software eran personas con puestos gerenciales que no estaban involucrados en la problemática cotidiana del almacén y tampoco iban a ser los usuarios finales del software.

Esta práctica podría resultar tal como la describe XP en aquellos casos en que el equipo de negocios se componga (al menos en parte) por personas que sean los usuarios del software una vez que se encuentre en uso, lo cual podría no resultar sencillo ya que las organizaciones no siempre se encuentran dispuestas a cambiar su manera de trabajar.

- XP asigna objetivos específicos a cada una de las partes involucradas en las reuniones de planeación, sólo que en este proyecto fue muy difícil para el equipo de negocios establecer el alcance del proyecto o el contenido de las entregas debido a las razones expresadas en el punto anterior. En este proyecto la definición de muchos requerimientos de usuario fue labor conjunta de ambos equipos y en algunos casos labor exclusiva del equipo de desarrollo (como en el caso de ciertas opciones a desarrollar en los lectores ópticos).

Para que esta diferencia no resulte tan marcada respecto a lo que se menciona en XP resulta necesario (como en el caso anterior) que aquellas personas que efectivamente serán usuarios del software tengan una participación activa en las discusiones de planeación de objetivos y especificación de requerimientos.

- XP propone la práctica de cliente en el sitio de desarrollo, en la cual uno o varios miembros del equipo de negocios permanecen constantemente junto con el equipo de desarrollo para resolver dudas, hacer comentarios y observar el avance en la construcción del software. Desgraciadamente, esta práctica resulta casi imposible de llevar a la práctica debido a que difícilmente una empresa permitirá a uno o varios de sus empleados fungir como asesores de tiempo completo para el equipo de desarrollo. Esta limitación se debe principalmente a que la frecuencia de las dudas sobre el sistema no resultará justificante para requerir asesores de tiempo completo, además de que el alejamiento del lugar de trabajo implicará en alguna u otra forma un descuido de las labores del empleado.

En el proyecto para el almacén editorial se prefirió mantener una estrecha comunicación telefónica con el cliente para resolver aquellas dudas o conflictos menores. Para discurrir dudas esenciales se convocaba a reuniones con el equipo de negocios (donde generalmente no todos los miembros estaban disponibles) que podían concluir en unas horas con resultados satisfactorios o podían alargarse por días o semanas sin resultados claros.

En otros proyectos se ha optado por improvisar un espacio dentro de la organización para que el equipo de desarrollo trabaje cerca del cliente, lo cual resulta lo más cercano a la práctica de cliente en el sitio de desarrollo con la ventaja de que el equipo de negocios no es separado físicamente de su lugar de trabajo y sólo es consultado cuando se le requiere. Desgraciadamente, llevar a cabo esta práctica depende de muchos factores como la disposición a cooperar por parte del cliente, las dimensiones físicas de las instalaciones del cliente o la distancia entre el lugar de trabajo del cliente y del equipo de desarrollo.

5.3. Segunda iteración

Una vez concluida la primera iteración del sistema y habiendo conocido los elementos esenciales que están involucrados en el módulo de surtido de pedidos nos encontramos en un buen momento para esbozar una metáfora del sistema (ver 1.2.3) que ayude a los involucrados en el proyecto a entender mejor el sistema y que permita explicar de forma clara el funcionamiento del software a aquellas personas no involucradas.

Como base de la metáfora del sistema ubicaremos el concepto de paquete. Dentro del sistema un paquete representará una caja, bolsa, saco, o cual-

TESIS CON
FALLA DE ORIGEN

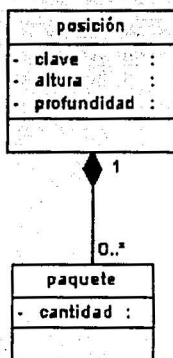


Figura 5.14: Diagrama UML de clase de los conceptos participantes en la metáfora.

quier otro elemento físico que permita agrupar libros. La única restricción de este agrupador será el hecho de que permite guardar sólo libros del mismo título. En el siguiente nivel encontramos a las posiciones del almacén editorial, las cuales pueden ir desde una posición contenida en un estante de almacenamiento hasta una sección marcada en el piso del almacén. Las posiciones funcionan como repositorios de paquetes, donde la cantidad de paquetes posibles en una posición se encuentra limitada únicamente por las capacidades físicas de la propia posición. (ver fig. 5.14)

Utilizando paquetes y posiciones podemos tener una idea general del funcionamiento del software ya que es posible concebir al sistema para el almacén editorial como un “mover” de paquetes entre posiciones y “modificador” de los atributos de los mismos, debido a que cualquier actividad realizada en el almacén se puede descomponer en traslados y modificaciones de paquetes.

En el caso del surtido de pedidos, éste se puede visualizar como el traslado de paquetes con libros de posiciones de almacenamiento en el almacén (estantes de almacenamiento, racks, etc) a posiciones para la realización de procesos (empaquetado, embalaje, plastificado) o a posiciones para embarque de material.

En el caso de que un surtido sólo tome parte de los libros contenidos

en un paquete, el surtido se puede visualizar como una modificación del paquete, creación de nuevos paquetes (que es a su vez un caso especial de modificación) y traslado de los paquetes creados.

De acuerdo con los ciclos cortos de desarrollo promovidos por XP, una vez concluido el desarrollo de los requerimientos iniciales de usuario es necesario volver a aplicar la práctica de juego de planeación para detallar el conjunto de requerimientos que conformarán la segunda iteración.

En este ciclo se pretende que el sistema no sea utilizado únicamente como auxiliar en las actividades del almacén editorial sino que las operaciones cotidianas se realicen mediante el sistema con un uso extensivo de los lectores ópticos de códigos de barras.

Los requerimientos de usuario incluidos en la presente iteración son los siguientes:

1. Para la realización de la opción surtido se requiere dotar a las posiciones del almacén de un atributo para saber en qué orden será realizado el recorrido de surtido. Este atributo se llamará "orden de recorrido".
2. El requerimiento más importante en esta iteración será una opción desarrollada en el lector óptico de códigos de barras que permita realizar la labor de surtido proponiendo posibles ubicaciones de surtido y respetando el orden de recorrido de las posiciones.
3. Se requiere un componente que permita a los supervisores monitorear el avance en el surtido de los pedidos.
4. Se requiere la construcción de opciones en el lector óptico que permitan a un empleado monitorear el avance del surtido en cantidades surtidas y faltantes por surtir.

5.3.1. Desarrollo de la iteración

Orden de recorrido

La noción de orden de recorrido surge de la necesidad de reflejar en el sistema la "accesibilidad" de cada posición, la cual depende de su ubicación física. Este ordenamiento de las posiciones permite formar un recorrido de surtido, es decir, una ruta o camino que los surtidores deben seguir para llevar a cabo la labor de surtido.

El "orden" de las posiciones es agregado como un atributo llamado "Orden de recorrido" en la clase posición (fig. 5.15).

TESIS CON
FALLA DE ORIGEN

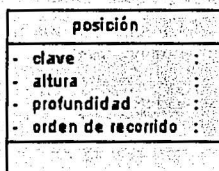


Figura 5.15: Diagrama UML para la clase posición

Asimismo, el desarrollo iterativo hace que la agregación de un atributo a la clase posición se vea reflejada también en un cambio a la pantalla de administración de posiciones, a la cual se le realizaron adaptaciones para que mostrara el atributo "orden de recorrido" de las posiciones. La realización de estos pequeños cambios conforme avanza el proyecto va acorde con el valor de simplicidad de XP, ya que cada agregación de funcionalidad o atributos al software obedece a una necesidad y no a una previsión de funcionalidades futuras.

Para que el surtido funcione correctamente es necesario que el "orden de recorrido" sea asignado de manera consistente, es decir, el valor del atributo dependerá de la accesibilidad de la posición en el estante de almacenamiento, así como de la zona y el pasillo en que se encuentre el estante.

En base a consultas con el cliente (cliente en el sitio de desarrollo) se llegó a la conclusión de que una posición resulta más accesible que otra si:

1. Se encuentra ubicada en un estante más cercano de acuerdo al recorrido natural del surtidor en el almacén¹.
2. Se encuentra en un nivel de altura más cercano al piso.
3. Se encuentra en un nivel de profundidad más cercano al pasillo.

Para ejemplificar la asignación de orden de recorrido a las 12 posiciones de que consta un estante de almacenamiento de 4 niveles de altura y 3 niveles de profundidad se muestra la figura 5.17.

¹El recorrido natural que siguen los empleados comienza en los estantes de obra general siguiendo por los pasillos en el sentido de las manecillas del reloj, ver fig. 4.1

☛ Sistema General de Ordenes, Logística y Distribución

Archivo Edición Inventarios Facturación Cuentas por cobrar Contabilidad Back
Administración Ventana 2

☛ Posiciones

Lista

Clave	Altura	Profundidad	Orden
AA00111	1	1	0
AA00121	2	1	0
AA00131	3	1	0
AA00141	4	1	0
AA00151	5	1	0
AA00161	6	1	0
AA00171	7	1	0
AA00211	1	1	0
AA00221	2	1	0
AA00231	3	1	0
AA00241	4	1	0
AA00251	5	1	0
AA00261	6	1	0
AA00271	7	1	0
AA00311	1	1	0
AA00321	2	1	0

Lista

Figura 5.16: Pantalla de administración de posiciones

Surtido de pedidos

El objetivo de este requerimiento es construir una opción inicialmente sencilla (de acuerdo al valor de simplicidad de XP) que permita realizar la actividad de surtido utilizando los lectores ópticos de códigos de barras.

Para que un surtidor en el almacén comience a recolectar libros para un pedido es necesario que un supervisor le asigne una lista de surtido para atender (fig. 5.18).

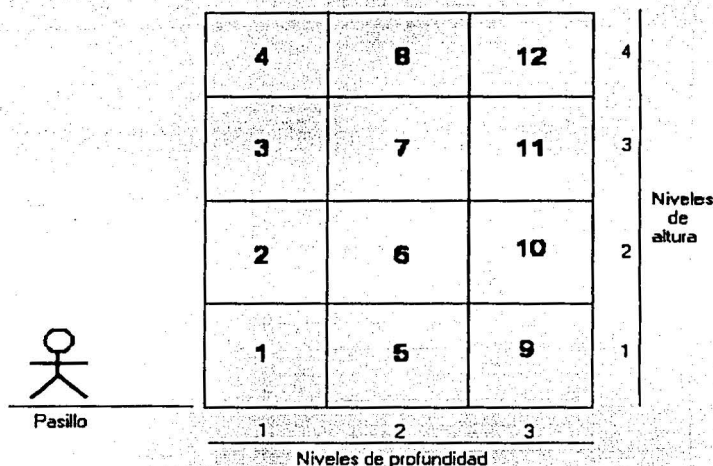


Figura 5.17: Orden de recorrido en un estante de almacenamiento de 3 niveles de profundidad y 4 de altura.

La asignación de la lista de surtido al empleado se realizaba antes de la llegada del sistema mediante un documento que contenía los títulos y cantidades requeridas de cada título en la cual el surtidor iba marcando los libros que recolectaba. En el sistema consideraremos asignada una lista de surtido cuando el supervisor comunique verbalmente a un empleado el número de la lista de surtido requerida para que el empleado ingrese a la opción de surtido e ingrese el número.

La opción desarrollada en el lector óptico llamada "surtido" consiste en la presentación de propuestas o sugerencias de surtido en la pantalla del lector óptico que deben ser confirmadas por los usuarios.

Definir los parámetros importantes dentro del algoritmo de selección de propuestas no fue tarea fácil, ya que se invirtió mucho tiempo en reuniones con el equipo de negocios para definir el algoritmo en términos generales. Además, una vez detallado el esquema general surgieron muchas dudas en el equipo de desarrollo al momento de programar respecto a casos no consi-

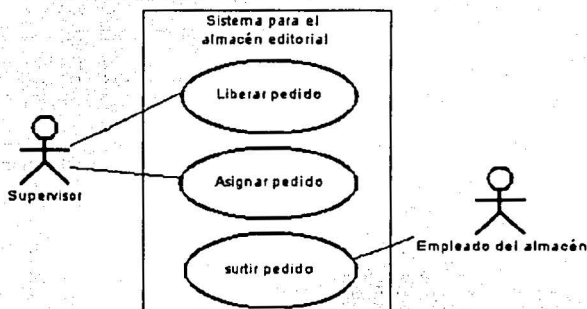


Figura 5.18: Caso de uso de asignación de pedidos

derados originalmente, lo cual también produjo retrasos.

El algoritmo para elegir una posición y mostrarla al empleado como propuesta de surtido es el siguiente (supondremos que el supervisor ha informado al empleado el número de la lista de surtido correspondiente):

- El usuario ingresa a la opción "Surtido" y digita en el lector óptico el número de la lista de surtido asignada, el cual es verificado (que sea una lista existente y que no haya sido surtida completamente).
- Las propuestas de surtido son mostradas en base a las siguientes consideraciones(fig. 5.19):
 1. El sistema resume las cantidades pendientes por surtir de cada título en los pedidos pertenecientes a la lista de surtido.
 2. Se realiza una búsqueda de todas aquellas posiciones que contengan los títulos obtenidos en el paso anterior.
 3. Las posiciones son ordenadas de acuerdo al atributo orden de recorrido, siendo presentada al usuario aquella posición con el atributo de orden menor.
 4. El usuario confirma la propuesta o ajusta la posición en caso de que existan inconsistencias entre el sistema y las cantidades y títulos existentes en la posición.

TESIS CON
FALLA DE ORIGEN

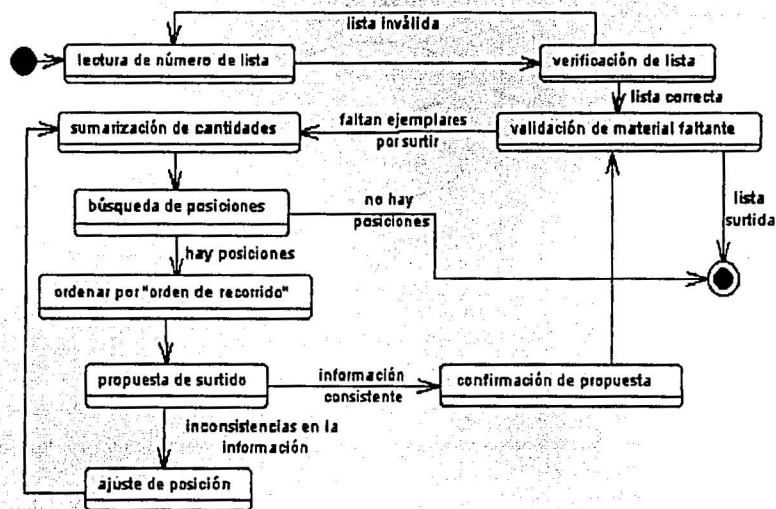


Figura 5.19: Algoritmo para presentar propuestas de surtido

En caso de que la posición resultante sea una posición accesible únicamente mediante montacargas, el surtidor tendrá que informar verbalmente al montacarguista la posición y el título deseado. Antes de implantar el sistema la comunicación entre surtidor y montacarguista era de esta manera, sólo que se carecía de información precisa respecto a la ubicación de los títulos y se dependía en gran medida de la memoria del montacarguista en turno.

Avance de pedidos

La práctica pequeñas entregas de XP hace hincapié en desarrollar aquellos requerimientos que se consideren primordiales en el momento actual. La creación de un componente de software que permite la realización de la labor de surtido a través del lector óptico necesita de una herramienta para el monitoreo de esta labor.

☛ Sistema General de Órdenes, Logística y Distribución

Archivo Edición Inventarios Eacturación Cuentas por cobrar Contabilidad Pack
Administración Ventana 2

☛ Avance de pedidos generados

Pedidos

Pedido	Lista	Original	Surtido	Falta surtir	Surt. conf.	Fec. generación	Fec. requerida
84080	11568	77	7	70	NO	24/04/2003 10:50	27/04/2003
84083	11565	1	1	0	SI	24/04/2003 19:16	27/04/2003
1338	480	5,200	5,200	0	SI	22/04/2003 15:34	26/04/2003
1368	485	3	3	0	SI	24/04/2003 19:36	28/04/2003
1370	487	2	2	0	SI	24/04/2003 19:36	28/04/2003
1371	488	5	5	0	SI	24/04/2003 19:36	28/04/2003
1373	490	1	1	0	SI	24/04/2003 19:36	28/04/2003
1374	491	3	3	0	SI	24/04/2003 19:36	28/04/2003
1375	492	101	101	0	SI	24/04/2003 19:36	28/04/2003
1379	496	59	59	0	SI	24/04/2003 19:36	28/04/2003
60465	51493	30	30	0	SI	22/04/2003 10:20	26/04/2003
60468	51554	92	40	0	SI	24/04/2003 10:50	26/04/2003
60479	51516	2,500	2,500	0	SI	23/04/2003 13:36	27/04/2003
60510	51534	63	63	0	SI	23/04/2003 20:14	27/04/2003
60513	51537	63	63	0	SI	23/04/2003 20:14	27/04/2003

Listo

Figura 5.20: Pantalla de avance de pedidos generados

La necesidad de este componente ha dado lugar a la realización de una pantalla que permita monitorear el surtido de los pedidos que han sido asignados a personal del almacén llamada "Avance de pedidos generados" (fig. 5.20).

De acuerdo a la solicitud del cliente y la sencillez inicial requerida en la pantalla, esta es la información que aparecerá en forma de columnas:

- Número de pedido.
- Número de lista de surtido a la que pertenece el pedido.
- Original: cantidad de ejemplares requerida por el pedido, la cual se obtiene a partir del atributo "cantidad" en la clase pedido_detalle.

TESIS CON
FALLA DE ORIGEN

- **Surtido:** cantidad de ejemplares surtidos², la cual se obtendrá a partir de un nuevo atributo que debemos agregar a la clase pedido_detalle llamado "surtido" (fig. 5.21)

pedido_detalle	
- cantidad :	
- surtido :	

Figura 5.21: Clase pedido detalle

- **Falta surtir:** es la resta entre la columna "Original" y "Surtido" y representa la cantidad de libros que hace falta recolectar para completar el surtido del pedido.
- **Surtido confirmado:** es un campo que permitirá al encargado de monitorear la operación confirmar y dar por terminado un surtido (sin importar si las cantidades requeridas han sido surtidas completamente o no).
- **Fecha de liberación:** es la fecha en la que el pedido fué remitido al almacén por parte del Departamento de Ventas de la compañía.
- **Fecha de generación:** es la fecha en la cual el pedido fue asignado a una lista de surtido para ser atendido por un empleado del almacén.
- **Fecha requerida:** es la fecha en la que se debería tener listo el pedido.

Para poder realizar la confirmación del surtido de los pedidos se tuvo que modificar la clase pedido agregándole un atributo nuevo llamado surtido confirmado (fig. 5.22), con el cual sabremos si el pedido ya ha sido confirmado (la labor de confirmación de surtido será exclusiva de los supervisores en el almacén a través de esta pantalla).

La pantalla "Avance de pedidos generados" va a incluir más columnas conforme se vayan desarrollando los módulos del sistema relativos a la actividades de empaque, embarque, entrega, etc y se requiera monitorear estas

²En todo pedido la cantidad surtida siempre será menor o igual que la cantidad original.

pedido	
• número de pedido	:
• fecha de generación	:
• fecha requerida	:
• situación	:
• pedido confirmado	:

TESIS CON
FALLA DE ORIGEN

Figura 5.22: Clase pedido

actividades. Sin embargo, se ha decidido no incluir componentes que prevengan esas modificaciones hasta el momento en que se requieran, ya que el desarrollo iterativo y la práctica de XP diseño simple mencionan que en un futuro las necesidades podrían cambiar y esa funcionalidad ya no sería requerida.

Consulta de títulos surtidos y por surtir

Acorde con la práctica de pequeñas entregas y debido al desarrollo de la opción "Surtido" surge la necesidad de desarrollar opciones complementarias. En este caso, se decidió realizar un par de consultas que se realizarán mediante el lector óptico llamadas "Consulta de títulos surtidos" y "Consulta de títulos por surtir" que permitirán al surtidor conocer los títulos y cantidades surtidos o faltantes por surtir de una lista de surtido determinada (estas consultas representan el equivalente a las marcas anteriormente realizaban en su papel los surtidores).

La mecánica de esta opción consistirá en el ingreso por parte del usuario de un número de lista de surtido recibiendo a cambio los títulos y cantidades surtidos o faltantes por surtir de esa lista. (fig. 5.23)

En ambas consultas se realiza una sumarización de los atributos "cantidad" y "surtido" de la clase pedido línea de acuerdo al siguiente algoritmo.

1. Se realiza una búsqueda de todos aquellos pedidos pertenecientes a la lista de surtido.
2. Una vez hallados los pedidos, se realiza la sumarización en pedido detalle para el atributo "cantidad" o "surtido"
3. Las cantidades son organizadas y presentadas en la pantalla del lector óptico.

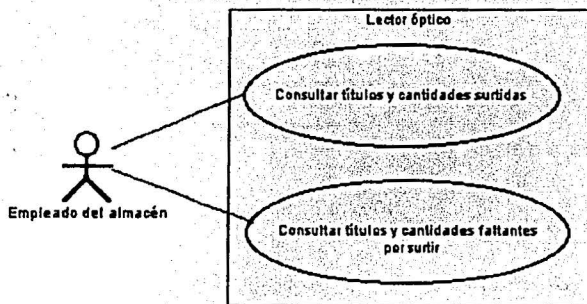


Figura 5.23: Caso de uso para las consultas "Títulos surtidos" y "Títulos por surtir"

5.3.2. Fin de la iteración

Siguiendo la práctica de pequeñas entregas de XP, al concluir el desarrollo de los requerimientos de la presente iteración se realizó una segunda entrega del sistema en el que aparece toda la funcionalidad desarrollada en la primera iteración más las siguientes características:

- Agregación a la pantalla de administración de posiciones de la columna "orden de recorrido"
- Opción "Surtido" en el lector óptico
- Pantalla "Avance de pedidos generados"
- Consulta de títulos y cantidades surtidas
- Consulta de títulos y cantidades por surtir

Con esta entrega se espera impulsar de manera importante la utilización de los lectores ópticos de códigos de barras en la operación diaria del almacén, así como permitir un monitoreo adecuado tanto de la labor de los empleados como de la velocidad y eficacia en la recolección de libros.

En el diagrama de clase 5.24 se muestra el estado del sistema al término de la segunda iteración, resaltando los atributos o clases agregados.

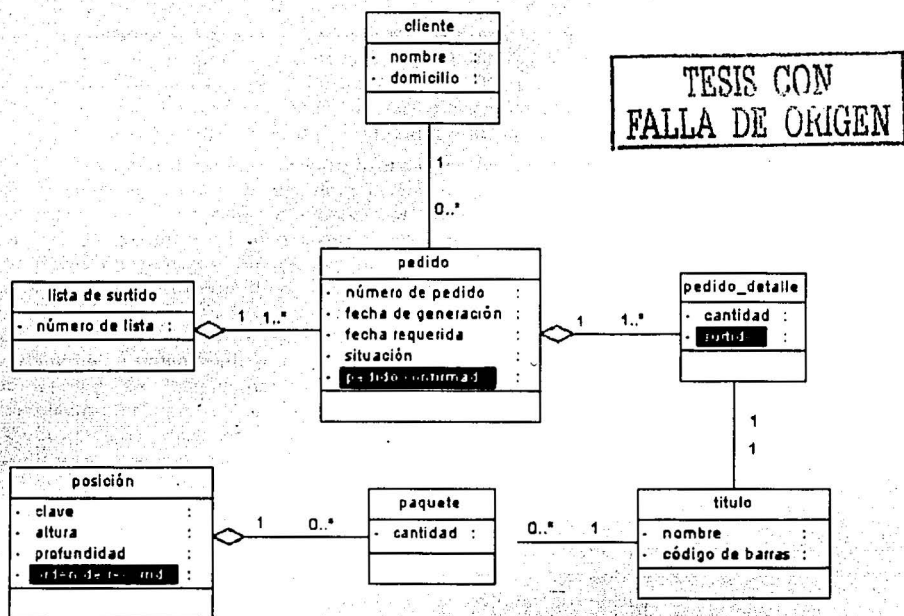


Figura 5.24: Diagrama de clase para la segunda iteración

5.3.3. Acerca del desarrollo en XP

En este apartado se hablará acerca de las prácticas de desarrollo de XP utilizadas en el transcurso del proyecto.

- La programación por pares representa una práctica novedosa dentro del conjunto de acciones propuestas en XP. En el proyecto para el almacén editorial esta práctica se utilizó en ciertos momentos, a diferencia de la propuesta de XP de llevar a cabo el desarrollo con esta práctica en todo momento. El no haber aplicado esta práctica en todo momento se debió a que la mayoría de las veces no se contaba con

el suficiente tiempo para desarrollar en parejas y resultaba preferible que cada programador se encargara de un componente específico para desarrollar. En XP se menciona que a la larga la programación por pares resulta más productiva que el desarrollo por separado, pero ante situaciones de gran apremio preferimos no experimentar.

En la práctica se observó que la programación por pares resultaba útil en aquellas secciones en las que había gran cantidad de dudas respecto a la forma en que el software debía comportarse. Para incorporar a nuevos miembros al equipo de desarrollo vimos que esta práctica resulta útil, ya que después de un período de programación por pares la adaptación de los nuevos elementos resulta más rápida y apegada a los estándares de desarrollo del equipo (práctica de estándares de codificación, ver 1.2.12) que con los elementos con los que no se practicaba programación por pares. Sin embargo, el dinamismo en el cambio de roles en una pareja y el cambio entre parejas no se realizó de manera tan dinámica como propone XP, ya que en aquellas situaciones en que se utilizó la Programación por pares resultaba contraproducente la realización tan dinámica de cambios (falta de consenso respecto al diseño o integración deficiente de los miembros al equipo).

- La refabricación unida a la practica de propiedad colectiva forman una actividad altamente dinámica en la que todo miembro del equipo de desarrollo puede modificar el código existente para hacerlo más sencillo o entendible sin modificar su funcionalidad. Estas prácticas se utilizaron pocas veces debido a que no resulta tan sencillo realizar modificaciones y quedar completamente seguro de que no se ha afectado en alguna forma la funcionalidad. Las ocasiones en que se llevaron a cabo refabricaciones se debió a que los cambios eran pequeños e identificables o el código era propio y por tanto, se conocían las repercusiones que los cambios iban a ocasionar. Cuando se requerían cambios grandes e importantes se prefería llamar al programador que había realizado el código y contar con su participación en la refabricación.
- Estándares de codificación es una práctica que resulta intuitivamente necesaria para cumplir con dos valores de XP: simplicidad y comunicación. En el proyecto esta fue la práctica que tuvo mas consenso y menor debate respecto a su aplicación debido a el equipo de desarrollo ya contaba previamente con estándares propios respecto a la manera de codificar. (nomenclatura común para designar variables, funciones, etc).

La utilidad de esta práctica salta a la vista en el momento en que se intenta leer y entender el código escrito por otras personas, por lo cual resulta una obligación de los miembros del equipo de desarrollo hacer uso de esta práctica para favorecer el entendimiento y el intercambio de información.

- La integración continua es una práctica importante debido a que permite detectar errores que pueden surgir de la integración de varios componentes desarrollados independientemente y además permite contar con una única versión actualizada del software.

En el proyecto esta práctica se llevó a cabo de manera constante y por lo general se realizaba una vez al día (generalmente en la noche) para contar a la mañana siguiente con una versión actualizada.

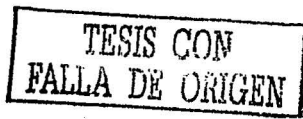
5.4. Tercera iteración

Como en anteriores iteraciones, una vez concluida una iteración se aplica la práctica de juego de planeación de XP para detallar los próximos requerimientos de usuario a realizar. La meta de esta iteración es fortalecer la opción de surtido, agregándole mayor funcionalidad, basándonos en las necesidades y sugerencias de los usuarios finales a raíz de la utilización del software en las actividades del almacén.

En esta iteración el juego de planeación resultó más ágil debido a que las ideas respecto al alcance del sistema y los conceptos importantes involucrados en el desarrollo resultan más familiares debido a la experiencia adquirida, la manipulación del software y el uso de la metáfora. Además, el hecho de que los verdaderos usuarios del sistema hayan comenzado a utilizar y probar el software proporciona mucha información respecto a las necesidades reales de los usuarios.

Los requerimientos de usuario a desarrollarse en esta iteración son las siguientes:

1. Se requiere organizar las posiciones en el almacén por zonas para facilitar su clasificación y mejorar la opción de surtido en el lector óptico.
2. Se requiere una opción en el lector óptico que sea para uso exclusivo de los montacarguistas, en la cual se ponga al alcance de los empleados a pie títulos existentes en la zona de Stock Mayor o grandes volúmenes sin la necesidad de comunicación verbal entre empleados a pie y surtidores (ver. 4.11).



3. Se requiere incluir en la opción "Surtido" elementos que permitan elegir una posición destino para los libros recolectados y facilitar la realización de un surtido entre varias personas.
4. Mejora de la consulta de título realizada en la primera iteración para que despliegue las existencias del título por zona.
5. Confirmación automática del surtido cuando se terminen de recolectar las cantidades solicitadas en el pedido.

5.4.1. Desarrollo de la iteración

Agrupación de posiciones

La existencia de diversas secciones en el almacén con características bien definidas han creado la necesidad de agrupar las posiciones del almacén en zonas. La existencia de pasillos en el almacén nos hizo pensar también en la creación de pasillos que agruparan posiciones y de zonas que agruparan pasillos para futuros requerimientos, pero siguiendo el valor de simplicidad de XP se decidió agrupar las posiciones únicamente por zonas. (ver figs. 5.25 y 5.26)

Por lo tanto para realizar este requerimiento vamos a crear una nueva clase llamada zona, la cual inicialmente contará con una clave o identificador y un orden de recorrido (el mismo concepto existente en las posiciones pero a nivel de zona).

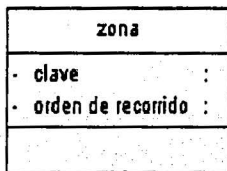


Figura 5.25: Diagrama UML para la clase zona

La agregación de zonas al sistema hace necesario el desarrollo de un componente que permita administrar las zonas. Por tanto, se decidió crear una pantalla en la cual es posible insertar, modificar o eliminar zonas del almacén. (fig. 5.27)

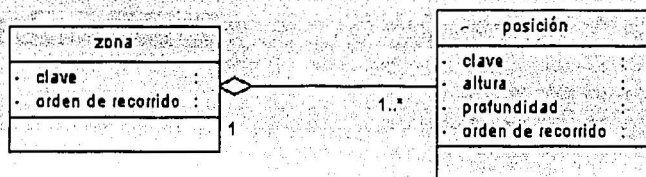


Figura 5.26: Relación entre zonas y posiciones

Sistema General de Ordenes, Logística y Distribución

Archivo Edición Inventarios Eacturación Cuentas por cobrar Cqntabilidad Back
Administración Ventana 2

Zonas del almacén

Lista

Clave	Orden
DEOG	0
INS	0
MESAS	0
MOST	0
MUESTRA	0
PISO	0
PKVO	0
RECIB	0
REOG	0
RESA	0
RUTA	0
TRASLADO	0

Lista

Figura 5.27: Pantalla de administración de zonas

Acorde con el desarrollo iterativo se realizaron modificaciones a la pantalla de administración de posiciones para que permita seleccionar la zona correspondiente a cada posición.

TESIS CON
FALLA DE ORIGEN

☒ Sistema General de Ordenes, Logística y Distribución

Archivo Edición Inventarios Facturación Cuentas por cobrar Contabilidad Pack
 Administración Ventana ?

☒ Posiciones

Lista

Cleve	Orden	Altura	Profundidad	Zona
AA01111	0		1	ZN-A - PICKING OBRA GENERAL
AB01111	0	1	1	ZN-A - PICKING OBRA GENERAL
AC01111	0	1	1	ZN-A - PICKING OBRA GENERAL
AD01111	0	1	1	ZN-A - PICKING OBRA GENERAL
BA01111	0	1	1	ZN-B - STOCK MAYOR
BB01111	0	1	1	ZN-B - STOCK MAYOR
BC01111	0	1	1	ZN-B - STOCK MAYOR
BD01111	0	1	1	ZN-B - STOCK MAYOR
BE01111	0	1	1	ZN-B - STOCK MAYOR
BF01111	0	1	1	ZN-B - STOCK MAYOR
DA01111	0	1	1	ZN-D - JAULA SANTILLANA
DB01111	0	1	1	ZN-D - JAULA SANTILLANA

Listo

Figura 5.28: Pantalla de administración de posiciones

La inclusión del concepto de zona en el esquema de diseño del sistema nos permitirá reflejar en el sistema las diversas zonas existentes en el almacén como zona de stock mayor, picking, etc. (ver 4.6)

Surtido para montacargas

A partir de la puesta en marcha del sistema en el almacén se hizo evidente la necesidad de una opción en el lector óptico específica para los montacarguistas debido a la existencia de problemas de comunicación entre empleados a pie y montacarguistas.

Estos problemas se deben a que en la operación tradicional del almacén los montacarguistas no tenían responsabilidad directa sobre ningún pedido en particular y cada surtidor a pie se acercaba a ellos cuando requería un título de la zona de Stock Mayor (posiciones inaccesibles para los surtidores a pie) lo que generaba aglomeración de personas en torno a los montacargas, incomunicación y favoritismos por parte de los montacarguistas.

Con esta nueva opción en el lector óptico llamada "Montacargas" se pretende que los montacarguistas continúen sin responsabilidad directa en el surtido de pedidos, cumpliendo la función de auxiliares en la operación pero evitando los problemas de comunicación.

Esta opción servirá como apoyo al proceso de surtido en forma general, no requerirá como entrada ninguna lista de surtido y mediante un algoritmo se propondrán posiciones en las cuales se encuentren títulos que sean requeridos por surtidores a pie, los cuales deberán ser trasladados por montacarguistas a posiciones accesible para los surtidores a pie, las cuales se encontrarán en una zona llamada "Picking Volumen".

El algoritmo que se utilizará para mostrar propuestas de traslado a los montacarguistas en la opción "Montacargas" (ver diagrama UML 5.29) es el siguiente:

1. El sistema sumaria las cantidades pendientes por surtir de cada título en todos aquellos pedidos cuya situación sea "Liberado" o "Generado" y no se encuentren confirmados.
2. Los títulos pendientes por surtir son ordenados descendientemente respecto a la cantidad faltante por surtir.
3. Se toma el primer título en la lista y se sumarian las cantidades existentes de ese título en posiciones pertenecientes a la zona Picking y a la zona Picking volumen.
4. Si la existencia en esas zonas es mayor o igual a la cantidad faltante por surtir, entonces se pasa al siguiente título.
5. Si la existencia es menor a la cantidad faltante por surtir, ese título será solicitado de la zona de stock mayor por algún surtidor a pie, por lo cual es un título elegible para ser trasladado.
6. Se realiza una búsqueda de aquellas posiciones de la zona de Stock mayor que contienen ese título.

TESIS CON
FALLA DE ORIGEN

7. Las posiciones son ordenadas de acuerdo a orden de recorrido y se presenta la primer posición como propuesta de traslado al montacarguista.

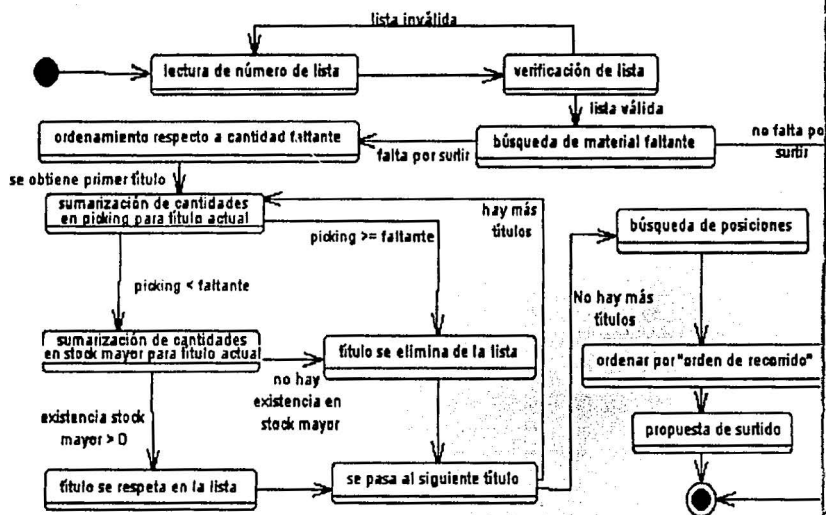


Figura 5.29: Algoritmo para obtención de propuestas en la opción "Montacargas"

Posición destino del surtido

Una vez que se concluye la recolección de los libros para un pedido específico se lleva a cabo el proceso de empaque en el cual los libros son colocados en cajas para su entrega posterior. La existencia del proceso de empaque y la utilidad de contar con todos los libros de un pedido agrupados en una posición única dió lugar a un requerimiento de usuario en el que se solicitó la realización de una pregunta al lector óptico al inicio de la opción "Surtido" sobre la posición para depositar los libros recolectados. Para realizar

TESIS CON
FALLA DE ORIGEN

esta sencilla pregunta es necesario modificar la clase paquete agregándole un atributo "asociado" para saber si el paquete se encuentra asociado a un pedido y un atributo "número de pedido" para saber a qué pedido pertenece el paquete. De esta forma se podrán diferenciar paquetes no asociados a pedidos (disponibles para surtir) de paquetes asociados (paquetes bloqueados para surtir).

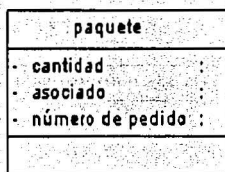


Figura 5.30: Clase paquete

La elección de una posición destino³ para depositar los libros cuenta con ciertas reglas que fueron surgiendo conforme aparecían dudas al desarrollar y se consultaban con el cliente para llegar a un acuerdo (práctica de cliente en sitio de desarrollo).

Las reglas para la elección de posición destino son las siguientes:

- La posición destino es elegida por el primer empleado a pie en atender la lista de surtido (en caso de que la lista sea atendida por múltiples empleados): aquellos empleados que posteriormente atiendan el pedido serán informados acerca de la posición destino, pero no la podrán modificar.
- Cada que se realice la confirmación de una propuesta de surtido se creará en la posición destino un paquete asociado a un pedido con el título y la cantidad confirmada. En caso de que al realizar la confirmación de surtido de un título el paquete asociado ya se encuentre creado, la cantidad surtida será agregada al paquete (fig. 5.31).
- En caso de que una lista de surtido cuente con varios pedidos (lista global) que requieren el mismo título la asignación de cantidades dará prio-

³Se ha pensado en la creación de una zona llamada "Mesas", así como de varias posiciones llamadas "Mesa 1", "Mesa 2", etc., que sirvan como posiciones destino de los surtidos.

ridad a los pedidos más urgentes, es decir, a aquellos pedidos que tengan fechas requeridas menores(ver fig. 5.31).

- Todo paquete no asociado a pedidos(disponible para surtir) tendrá un atributo "asociado" = no y "número de pedido" = 0.

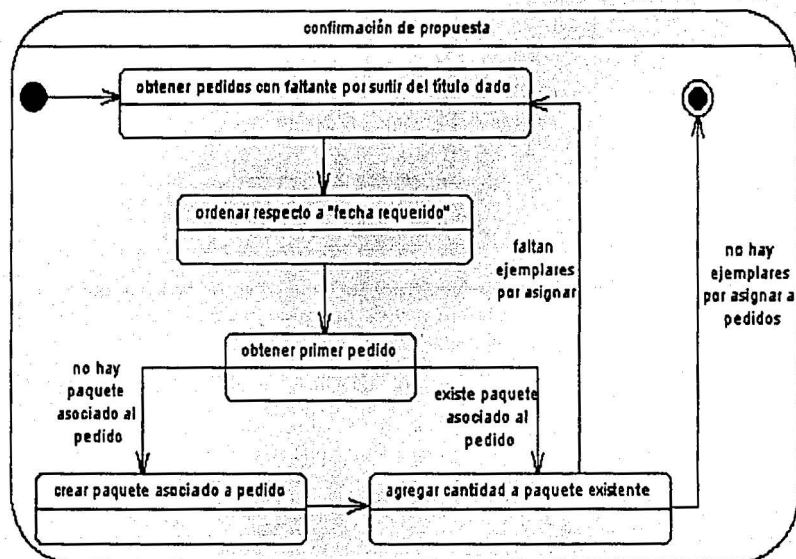


Figura 5.31: Algoritmo de asignación de cantidades a pedidos

Posición para iniciar el surtido

La realización de ciclos cortos de desarrollo y pequeñas entregas proporciona gran cantidad de información respecto a los componentes construidos en iteraciones previas, ya que al ser puestos en práctica al poco tiempo permiten detectar errores o insuficiencias. En el caso de la opción "Surtido"

TESIS CON
FALLA DE ORIGEN

construida en la segunda iteración nos dimos cuenta que el algoritmo de generación de propuestas no funcionaba bien cuando el pedido era atendido por varios empleados, ya que a todos les mostraba las mismas propuestas.

Intentar solucionar esta deficiencia condujo a conversaciones con el cliente que mostraron varias soluciones posibles. En este caso se procedió siguiendo el principio de simplicidad de XP y se acordó realizar un pequeño cambio a la opción "Surtido" en la cual se agrega una pregunta al inicio de la operación solicitando la posición en la que el empleado comenzará a surtir.

La posición elegida dependerá de la ubicación en que se encuentre cada empleado a pie y el cambio se verá reflejado únicamente en el algoritmo de surtido, el cual tendrá ahora la siguiente secuencia(ver fig. 5.32):

1. El sistema sumaliza las cantidades pendientes por surtir de cada título en aquellos pedidos pertenecientes a la lista de surtido.
2. Se realiza una búsqueda de todas aquellas posiciones que contengan los títulos obtenidos en el paso anterior.
3. Las posiciones son ordenadas respecto a "orden de recorrido".
4. Se obtiene el orden de recorrido de la posición de inicio de surtido.
5. Se muestra en el lector óptico la primer posición cuyo orden de recorrido sea mayor o igual que el orden de recorrido de la posición de inicio de surtido.

La inclusión de esta pregunta dentro de la mecánica de la opción de surtido tiene las siguientes consecuencias:

- Para realizar el surtido entre varias personas y evitar que el sistema genere propuestas similares a los empleados será necesario que cada empleado se sitúe en pasillos o zonas suficientemente alejadas respecto a sus compañeros seleccionando la primer posición de su zona o pasillo como posición de inicio.
- Si un surtidor elige como posición de inicio la primer posición de la zona de Picking y el sistema comienza a proponer posiciones de Stock mayor para el surtido, eso significará que los títulos requeridos se encuentran únicamente en posiciones de Stock Mayor y que el empleado a pie debe esperar a que los montacarguistas trasladen los libros a una posición accesible mediante la opción "Montacargas".

TESIS CON
FALLA DE ORIGEN

Consulta de título

Como parte de la retroalimentación obtenida a partir del uso del sistema por parte de los usuarios finales, se tomó en cuenta la sugerencia de agregar a la consulta por título desarrollada en la primera iteración un despliegue de existencias del título por zonas del almacén que permita al usuario elegir una zona y consultar qué posiciones de la zona contienen el título.

Para llevar a cabo este requerimiento es necesario modificar el algoritmo para la opción "Consulta de título", el cual se presenta a continuación:

1. El usuario ingresa a la opción "Consulta de título" en el lector óptico y proporciona la clave del título a buscar.
2. Se realiza una búsqueda de todos aquellos paquetes que contengan el título especificado.
3. Una vez encontrados los paquetes, se realiza una lista de aquellas posiciones que los contienen, sumalizando las cantidades en cada posición.
4. Con la lista de posiciones y cantidades del título, se realiza una sumalización de las cantidades existentes por zona.
5. La lista de zonas y cantidades existentes por zona es desplegada en la pantalla del lector óptico.
6. Si el usuario elige alguna de las zonas desplegadas en el lector óptico, se muestra en el lector óptico la lista de posiciones y sus cantidades para la zona elegida.

Confirmación automática de surtido

Otra sugerencia (derivada de los ciclos cortos de desarrollo) que recibimos por parte de los supervisores del almacén fue la modificación de la pantalla de avance de pedidos para que los pedidos se confirmen de manera automática cuando se hayan recolectado todos los libros requeridos, es decir, cuando para todos los títulos pertenecientes a un pedido el atributo "cantidad" y "surtido" de la clase pedido_detalle sean iguales. (ver fig. 5.32)

Esta petición surge porque la confirmación de cada pedido dentro de la pantalla de avance de pedidos consume demasiado tiempo a los supervisores y no les permite realizar otras actividades que tienen asignadas. Por lo tanto, esta modificación les permitirá hacerse cargo únicamente de los pedidos que no han sido completados y requieren una decisión por parte de los supervisores respecto a su confirmación.

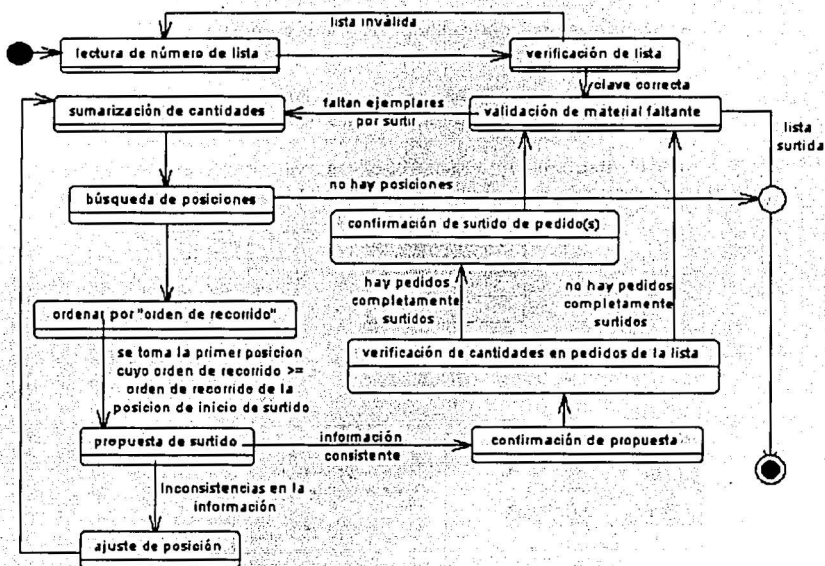


Figura 5.32: Algoritmo de surtido con confirmación automática

5.4.2. Fin de la iteración

A continuación mostraremos la lista de los componentes agregados al sistema en la presente iteración:

- Pantalla de administración de zonas
- Agregación en la pantalla de administración de posiciones de una opción para elegir la zona a la que pertenece cada posición.
- Opción "Montacargas" en el lector óptico.
- Modificación de la opción "Surtido" con la inclusión de opciones para elegir posición destino y posición para iniciar el surtido.

TESIS CON
FALLA DE ORIGEN

- Modificación de la consulta de título en el lector óptico de códigos de barras para agregar un despliegue de cantidades por zona y posición.
- Confirmación automática de surtido en la pantalla de avance de pedidos cuando se completen las cantidades requeridas.

En el diagrama de clase 5.33 se marcan los cambios realizados en la tercera iteración y se observa el estado del sistema hasta este momento.

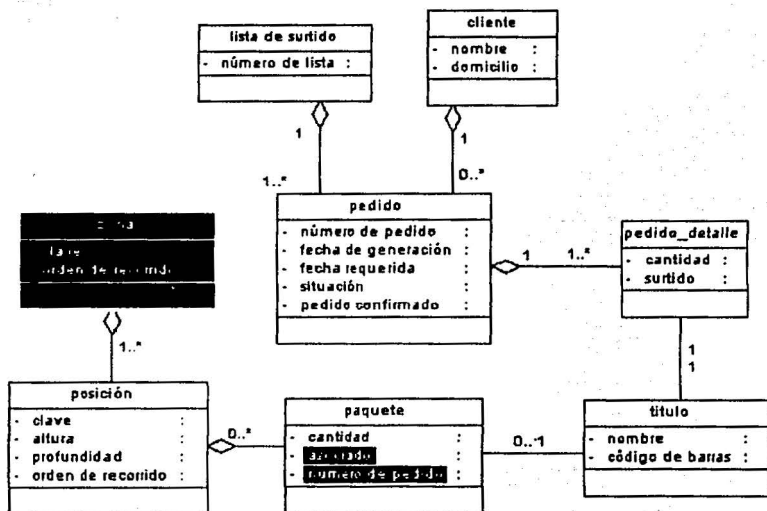


Figura 5.33: Diagrama UML de clase de la tercera iteración

5.4.3. Acerca de la implementación y pruebas en XP

- La práctica de XP denominada 40 horas semanales establece horarios "aceptables" de trabajo en aras de mantener un equipo de desarrollo activo y motivado. En el caso particular del proyecto para el almacén editorial esta práctica se llevó a cabo al principio del proyecto. Sin

TESIS CON
FALLA DE ORIGEN

embargo, debido a que el juego de planeación para la primera y segunda iteración tomó demasiado tiempo se hizo necesario alargar los horarios de trabajo para poder cumplir con lo acordado. Esta experiencia refleja el hecho de que no siempre es posible realizar una reestructuración de los tiempos de desarrollo, ya que en muchas ocasiones el cliente no resulta tan flexible o tan dispuesto a retrasar el desarrollo del software, por lo cual es necesario alargar los horarios de trabajo para poder concluir a tiempo.

Una medida que posiblemente permitiría aplicar esta práctica a lo largo de todo el proyecto es la cotización del proyecto por períodos pequeños de tiempo (justamente los ciclos cortos de desarrollo de XP) en lugar de una única cotización para todo el proyecto, ya que al ser menor la cantidad de dinero gastada por el cliente, el riesgo disminuye y por tanto es probable que se encuentre con mayor disposición al cambio.

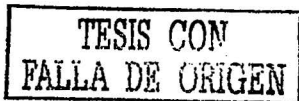
- XP propone la realización extensiva de pruebas, haciendo especial hincapié en el uso de pruebas automatizadas, es decir, utilización de software específico para realización de pruebas. En el proyecto para el almacén editorial se realizaron tanto pruebas por parte de los programadores al momento de desarrollar como pruebas con los usuarios finales en el almacén mismo. Las pruebas de los desarrolladores no utilizaron ningún software para pruebas, por lo que resulta imposible determinar qué tan benéfica resulta la utilización de estas herramientas.

En cambio, se puso gran énfasis en todos aquellos errores o detalles que surgían en la operación del almacén con el sistema en funcionamiento. Para poder corregir lo más pronto posible los errores y obtener retroalimentación de los usuarios respecto al sistema, el equipo de desarrollo se trasladó al almacén durante ciertas temporadas, lo cual resulta similar a la práctica cliente en sitio de desarrollo, sólo que en este caso podríamos llamarlo "desarrollador en sitio del cliente".

5.5. Cuarta iteración

Para esta iteración el conocimiento del software ha crecido bastante respecto a las anteriores iteraciones, por lo que el juego de planeación resulta mucho más ágil.

El objetivo de esta iteración es tener un monitoreo más completo del



surtido y obtener información que resulte de utilidad para la toma de decisiones.

Con base en estas ideas generales y con el apoyo del equipo de negocios se obtuvieron los siguientes requerimientos de usuario para esta iteración:

1. Se requiere un componente que permita administrar (dar de alta, modificar y eliminar) los empleados en el almacén.
2. Ingreso de los empleados a las opciones del lector óptico con una clave y contraseña personalizadas.
3. Se requiere llevar un registro de los movimientos de libros realizados en el almacén para control, monitoreo y obtención de estadísticas.
4. Se requiere construir una opción en el lector óptico para poder realizar movimientos de libros de una posición a otra.
5. Se requiere una opción para realizar cálculos aritméticos integrada al lector óptico de códigos de barras.

Los requerimientos para la presente iteración fueron elegidos de forma tal que complementen y extiendan lo desarrollado en anteriores ciclos y vayan acorde con las necesidades actuales, lo cual resulta consistente con el desarrollo iterativo y los ciclos cortos de XP.

5.5.1. Desarrollo de la iteración

Administración de empleados

La alta rotación de personal existente en el almacén hace necesaria la creación de un componente de software que permita tener control e información de los empleados que se encuentran trabajando actualmente en el almacén editorial.

Para lograr este objetivo resulta necesaria la creación de una clase llamada empleado con los atributos mostrados en el diagrama 5.34.

La pantalla de administración de empleados (fig. 5.35) será utilizada por los supervisores y permitirá ver los atributos de los empleados, así como dar de alta, modificar y eliminar empleados.

La clase empleado y la pantalla de administración de empleados cuentan con aquellas características consideradas necesarias y útiles en este momento (valor de simplicidad de XP) para tener control de los empleados actuales del almacén.

TESIS CON
FALLA DE ORIGEN

empleado	
- nombre	:
- clave	:
- contraseña	:

Figura 5.34: Clase empleado

Sistema General de Ordenes, Logística y Distribución

Archivo Edición Inventarios Ecturación Cuentas por cobrar Contabilidad Pack
Administración Ventana 2

Empleados

Lista

Nombre	Clave	Contraseña
JORGE RODRIGUEZ GONZALEZ	JRG	1971
JUANA VELA MORALES	JVM	1972
KEYLA ROQUE OLVERA	KRO	1984
LUIS BELTRAN VALENCIA	LBV	1945
LIZBETH GONZALEZ GONZALEZ	LGG	1977
LUCILA GOMEZ JIMENEZ	LGJ	1958
LUIS ALBERTO HERNANDEZ SORIANO	LHS	1981
LAURA OLVERA RODRIGUEZ	LOR	2088
LAURA SANCHEZ OLVERA	LSO	1980
MIRIAM ALMARAZ ARAUJO	MAA	1980
MARTHA DEL AGUILA DIAZ	MAD	1963
MIGUEL ALVARADO MONTEL	MAM	1957
MAGDALENA BERTAIN CASTILLO	MBC	1980
MARCOS BONILLA GARCIA	MBG	1973
MARGARITA BARRERA HERNANDEZ	MBH	1979
MA DE LA LUZ CASTRO GONZALEZ	MCG	1962

Lista

Figura 5.35: Pantalla de administración de empleados

TESIS CON
FALLA DE ORIGEN

Ingreso personalizado de empleados

A raíz de la puesta en marcha del software en el almacén se observó la necesidad de contar con un medio de control para el ingreso de los empleados al sistema (en este caso, a los lectores ópticos de códigos de barras), así como formas de obtener información de la actividad individual de cada empleado.

Mediante reuniones y pláticas telefónicas con el cliente llegamos a la conclusión de que el ingreso a los lectores ópticos debe ser personalizado, lo cual proporcionará un mejor control en la asignación y uso de estos dispositivos. De esta manera, el lector óptico preguntará clave y contraseña de empleado al inicio de la operación.

Registro de movimientos en el almacén

La idea de este requerimiento es dotar al sistema de un repositorio de datos para la obtención de estadísticas y reportes o realización de consultas para la toma de decisiones. (ver fig. 5.36)

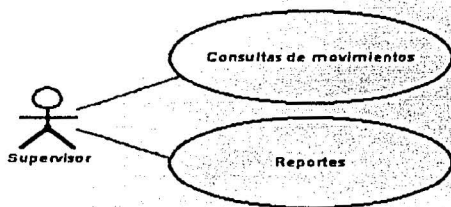


Figura 5.36: Caso de uso de actividades del supervisor

Los gerentes y supervisores del almacén requieren analizar la información en el almacén por tipo de actividad (surtido, empaque, etc), por lo cual se creó una clase llamada movimiento que contendrá información respecto a las labores realizadas en el almacén. La idea de esta clase es representar la fecha y hora en que se ejecutó una acción determinada en el almacén, por lo cual la clase movimiento contiene los atributos especificados en el diagrama 5.37.

El atributo tipo movimiento de la clase movimiento (fig. 5.37) representará los tipos principales de actividades en el almacén, entre los que se encuentran:

TESIS CON
FALLA DE ORIGEN

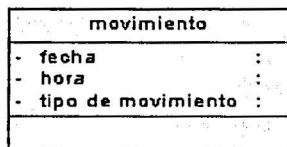


Figura 5.37: Clase movimiento

- Inventario
- Ajuste de posición
- Surtido
- Traslado

Un empleado puede realizar diversas tareas o actividades en el almacén, lo cual se refleja en nuestro diseño en una asociación entre las clases empleado y movimiento (fig. 5.38).

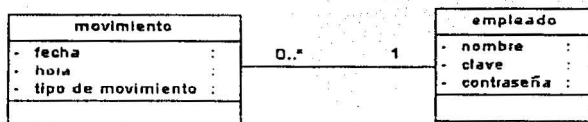


Figura 5.38: Asociación entre la clase empleado y movimiento

Para lograr información respecto a las actividades que los empleados realizan en el almacén, cada vez que se ejecute una acción utilizando el lector óptico de códigos de barras se registrará en la clase movimiento una acción con una fecha/hora determinada, así como el empleado que realizó la acción y el tipo de movimiento específico. (fig. 5.39)

Traslado

De acuerdo a la práctica de pequeñas entregas de XP, el desarrollo de una opción para realizar traslados de libros de una posición a otra fue desechado

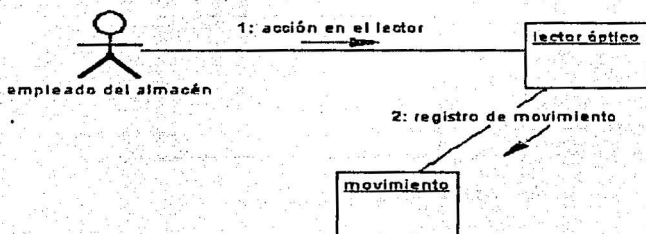


Figura 5.39: Registro de movimientos en el almacén

de la primer iteración debido a que había una manera alterna de solucionar el problema. En este punto del desarrollo se tiene un conocimiento mayor de las actividades que se realizan en el almacén y de la frecuencia con que se realizan, lo cual ha sido reforzado por la utilización del software. Con base en esta experiencia se llegó a la conclusión junto con el equipo de negocios de desarrollar la opción de traslado en esta cuarta iteración por dos razones:

1. El traslado de libros se ha vuelto una actividad muy frecuente en el almacén, por lo cual la creación de una opción explícita de traslado ayuda en la labor de los empleados y permite un ahorro de tiempo.
2. La inclusión de un repositorio de información por actividad como la clase "movimiento" obliga a crear la opción traslado, ya que si se continuaban simulando los traslados mediante la opción "ajuste de posición" las estadísticas por actividad no resultan reales.

La actividad de traslado consiste en tomar una cierta cantidad de libros de una posición del almacén (posición origen del traslado) y depositarlos en otra posición (posición destino del traslado).

La idea original obtenida en las reuniones con el cliente era que se permitiera mover cualquier cantidad de libros y títulos en un movimiento de traslado. Posteriormente al desarrollar surgieron casos no considerados originalmente que complicaban el desarrollo, por lo cual se decidió tomar una decisión junto con el cliente, aplicar el principio de simplicidad de XP y considerar un traslado como un movimiento de libros de un mismo título de una posición a otra.

Esta decisión tuvo influencia en la simplicidad en el diseño y en la reducción en los tiempos de desarrollo de esta opción cuya utilidad quedará de manifiesto al concluir la presente iteración y ser utilizada por el personal del almacén.

De acuerdo a las pláticas sostenidas entre cliente y equipo de desarrollo la secuencia para la opción "Traslado" es la siguiente:

1. El empleado ingresa a la opción "Traslado" en el lector óptico de códigos de barras.
2. El lector óptico solicita digitar la clave o disparar la etiqueta de la posición origen del traslado.
3. Se solicita el título y la cantidad de libros a trasladar.
4. El sistema valida que el título y la cantidad se encuentren en la posición indicada.
5. Se solicita al usuario la posición destino para depositar los libros.

Calculadora

Gracias a los períodos en que el equipo de desarrollo tuvo que trasladarse al almacén para corregir errores (cliente en sitio de desarrollo), los empleados del almacén se mostraron dispuestos a comunicar sus dudas e inquietudes respecto al software. En particular nos pidieron facilitarles de alguna manera la realización de operaciones aritméticas desde el lector óptico para agilizar sus labores. Esta necesidad se debe al hecho de que varias actividades del almacén (como traslado y surtido) requieren que los empleados realicen conteos de libros en posiciones en las cuales éstos se encuentran sueltos o agrupados en cajas. (fig. 5.10)

La opción calculadora permitirá realizar las cuatro operaciones aritméticas básicas (suma, resta, multiplicación y división) y será posible acceder a ella desde el lector óptico mediante una secuencia de teclas.

5.5.2. Fin de la iteración

A continuación se enumeran las características agregadas al sistema en la presente iteración:

- Pantalla para poder dar de alta, modificar o eliminar empleados.

TESIS CON
FALLA DE ORIGEN

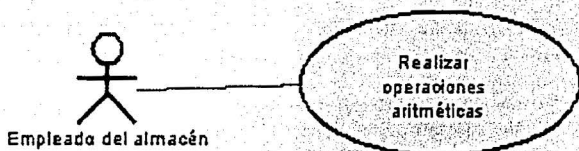


Figura 5.40: Caso de uso de las operaciones de la opción “Calculadora”

- Ingreso personalizado de los empleados al sistema mediante una clave y una contraseña.
- Registro de movimientos en el almacén por actividad y empleado.
- Opción para realizar traslados de libros de manera explícita.
- Opción en el lector para realizar operaciones aritméticas.

El diagrama de clases 5.41 nos muestra el estado alcanzado por el sistema hasta esta iteración resultando las nuevas clases o atributos incorporados.

5.5.3. Acerca del diseño

En el proyecto para el almacén editorial el diseño fue llevado a cabo tanto por el equipo de negocios como por el equipo de desarrollo.

La apariencia visual de la aplicación fue diseñada en su mayor parte por el equipo de negocios con la colaboración de miembros del equipo de desarrollo para guiar y orientar los diseños. El diseño visual de la aplicación fue realizada mediante dibujos en papel que son fácilmente modificables y una vez llegado a un acuerdo respecto al diseño en el papel proceder a construirlo en el entorno de desarrollo.

El diseño estructural del software fue llevado a cabo por el equipo de desarrollo con la participación de analistas del equipo de negocios. El diseño fue realizado de esta manera debido al conocimiento de los analistas sobre conceptos y detalles específicos del negocio editorial que resultaban de importancia en el desarrollo del software. Las modificaciones al diseño que fueron surgiendo con el paso del tiempo se informaban al equipo de negocios para que supiera de ellas y estuviera consciente de sus implicaciones. De la misma forma, cuando surgían cambios a nivel de negocios en la empresa

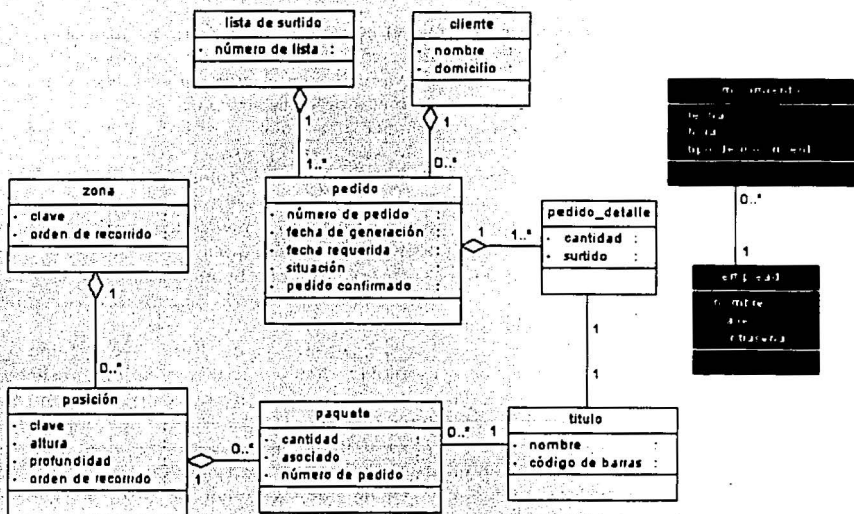


Figura 5.41: Diagrama de clase de la cuarta iteración

editorial, estos se informaban al equipo de desarrollo para saber si tendrían repercusión en el diseño del sistema.

Para el diseño estructural del proyecto se utilizaron ciertos diagramas UML, sobre todo los diagramas de clase mostrados en el presente trabajo para facilitar la comprensión de conceptos, así como para llevar un registro de la evolución y poder comparar los cambios sufridos por el proyecto a lo largo del tiempo.

Sobre los papeles que cada equipo debe desempeñar en el diseño XP no menciona roles específicos, ya que pone más énfasis en el hecho de comenzar con un diseño sencillo y mediante el desarrollo iterativo ampliarlo y expandirlo.

En este proyecto se trabajó de la manera antes descrita debido a que en esa forma nos sentimos a gusto y ambas partes tuvieron libertad para exponer sus puntos de vista. Esta forma de trabajar se dió así debido a

la experiencia con que cuenta el equipo de negocios con sistemas previos y a la comunicación y confianza existente entre ambos equipos debido a la experiencia en el desarrollo de varios sistemas para la empresa editorial, lo cual facilita la asignación de papeles de cada equipo.

Es muy probable que la manera de atacar el diseño deba adaptarse a las condiciones del momento dependiendo en gran parte de la experiencia del equipo de negocios en el ámbito de los sistemas y en la confianza y comunicación entre equipo de desarrollo y de negocios. En algunos escenarios se pueden adoptar esquemas similares al mostrado anteriormente y en otros es posible que el equipo de desarrollo sea quien lleve una participación mayoritaria en el diseño del software. Independientemente de la asignación de papeles adoptada, resulta una adecuada estrategia de diseño seguir los valores de simplicidad y comunicación de XP.

La utilización de UML como herramienta de diseño del sistema sobre otros diagramas o métodos de modelación se debió a su amplia variedad de diagramas y a que se está convirtiendo en un estándar, por lo que resulta importante conocer y entender este lenguaje.

UML es un lenguaje que permite describir una gran variedad de aspectos del sistema, aunque en el proyecto para el almacén editorial sólo fue utilizado para facilitar la comprensión de algunos elementos de difícil explicación verbal y para contar con "documentación visual" del sistema.

El hecho de que UML sea un lenguaje enfocado a la orientación a objetos nos impidió realizar diagramas de aspectos del sistema que estaban realizados en lenguajes de programación no orientados a objetos. Es por esto que el uso de UML no resultó tan extensivo como esperábamos.

5.6. Etapa final

Después de realizar varias iteraciones nos encontramos con un software cada vez más robusto y que cumple cada vez de mejor manera las metas que nos propusimos inicialmente.

Los componentes con que cuenta el sistema hasta este momento son:

- Opción que permite dar de alta o modificar títulos y cantidades en una posición llamada "Ajuste de Posición" en el lector óptico.
- Consulta de existencias de un título en el almacén por zona y posición en lector óptico llamada "Consulta de título".
- Consulta en lector de títulos y cantidades existentes en una posición dada llamada "Consulta de posición".

- Pantalla donde se visualizan los pedidos liberados por el Área de Ventas que permite al usuario seleccionar uno o varios pedidos para su arrastre a la situación "Generado".
- Pantalla para administrar las diferentes posiciones en el almacén.
- Opción "Surtido" que propone al usuario posiciones específicas para surtir material de una lista de surtido dada, haciendo esto de una manera metódica y ordenada. Además esta opción permite seleccionar una posición destino para los títulos surtidos, una posición de inicio de recorrido para realizar surtidos entre varias personas y confirmación automática del surtido.
- Pantalla de monitoreo de cantidades surtidas por pedido llamada "Avance de pedidos".
- Consulta en lector de títulos y cantidades surtidas para una lista de surtido específica.
- Consulta de títulos y cantidades que hacen falta surtir en una lista de surtido.
- Pantalla para administración de zonas en el almacén.
- Opción "Montacargas" que permite a un montacarguista del almacén conocer qué títulos son requeridos de la zona de stock mayor del almacén y por tanto saber qué títulos requieren ser trasladados a zonas accesibles a los surtidores a pie.
- Pantalla de administración de empleados del almacén.
- Ingreso personalizado de los empleados al sistema.
- Registro de movimientos en el almacén por tipo de movimiento o por empleado para realizar consultas, reportes y estadísticas.
- Opción "Traslado" para realizar movimientos de libros entre posiciones del almacén.
- Opción "Calculadora" en el lector que ayude a los empleados en la realización de conteos.

XP no hace mención acerca de un número específico de iteraciones, ya que éstas variarían significativamente dependiendo del tamaño y la complejidad de cada proyecto. Lo que sí podemos asegurar es que con cada iteración

TESIS CON
FALLA DE ORIGEN

se tendrá un mayor conocimiento de los elementos involucrados, así como un paulatino crecimiento y especialización del software, pasando de los elementos estructurales a los detalles más sofisticados.

Es también posible decir que llegará un momento en que el número de requerimientos por iteración comience a disminuir hasta llegar a un punto en que sólo se requieran correcciones menores, adaptaciones a nuevos escenarios y adiciones mínimas, lo cual es mencionado en XP cuando se dice que el estado en el cual un sistema debe permanecer la mayor parte del tiempo es en mantenimiento.

Para fines del presente trabajo esta es la última iteración presentada, ya que la intención es mostrar ciertas prácticas de XP aplicadas en un proyecto concreto, así como la experiencia obtenida a lo largo del desarrollo del mismo.

5.6.1. Perspectiva del proyecto para el almacén editorial

Actualmente el sistema es usado en el almacén editorial por aproximadamente 100 empleados divididos en tres turnos. En los momentos de mayor incidencia en el sistema se encuentran 4 supervisores utilizando las pantallas, mientras que en el almacén se encuentran aproximadamente 40 empleados con sus respectivos lectores ópticos realizando operaciones simultáneas en el sistema, entre las cuales destacan principalmente el surtido y el empaque.

Los siguientes resultados se han obtenido a partir del análisis de los movimientos en el almacén en los meses de abril y mayo de 2003:

- En el almacén se llevan a cabo de manera diaria aproximadamente 12,000 movimientos.
- La actividad con mayor cantidad de movimientos es el empaque de libros concentrando 40 % del total de movimientos del almacén.
- El 59 % de los movimientos del almacén corresponden a las actividades de surtido y empaque.
- La relación entre los movimientos de empaque y de surtido es de 2 a 1, es decir, por cada movimiento de surtido se generan en promedio 2 movimientos de empaque.
- En la primera mitad del día (00:00 hrs. - 12:00 hrs.) se realiza aproximadamente el 33 % de los movimientos diarios en el almacén, mientras que en la segunda parte del día (12:00 hrs. - 24:00 hrs.) se realiza el 66 % restante.

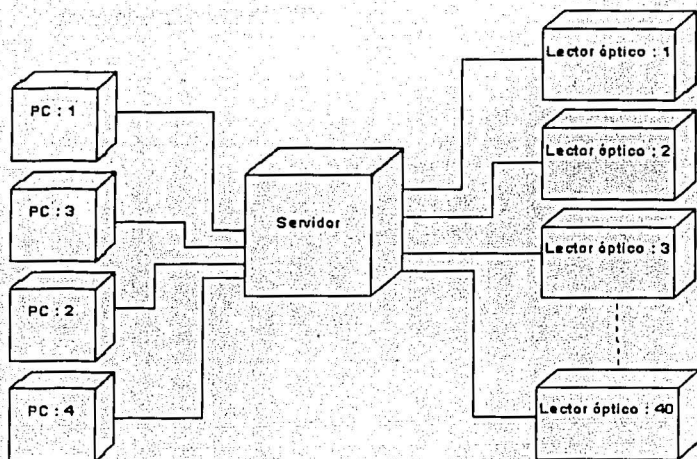


Figura 5.42: Componentes físicos del sistema

- Los títulos con mayor número de movimientos son los libros de texto utilizados como guías escolares para el nivel primaria.
- El empleado con mayor número de movimientos en el período considerado tiene un total de 20.132 de los cuales 75 % representan movimientos de empaque.
- La fecha con mayor cantidad de movimientos de surtido, traslado y empaque en el almacén se registró el día 13 de mayo de 2003 con 23.174 movimientos.
- Fecha con mayor cantidad de movimientos de surtido: 9 de mayo de 2003 con 6274.
- Fecha con mayor cantidad de movimientos de traslado: 20 de mayo de 2003 con 4080.
- Fecha con mayor cantidad de movimientos de empaque: 13 de mayo de 2003 con 16536.

TESIS CON
FALLA DE ORIGEN



TESIS CON
FALLA DE ORIGEN

Conclusiones

En la actualidad las empresas han experimentado una necesidad de información cada vez mayor, lo cual se traduce en requerimientos de sistemas de información cada vez más eficientes, mejor diseñados y producidos en un menor tiempo.

Estas necesidades fomentan el cuestionamiento de los métodos de desarrollo de software existentes y de la forma en que éstos pueden ser modificados para obtener una mayor eficiencia.

La Programación Extrema (XP) ha venido a cuestionar los métodos de desarrollo de software mediante prácticas "ágiles" que intentan reducir el riesgo inherente a todo proyecto de desarrollo de software.

En el presente trabajo se muestran las prácticas de XP utilizadas en un proyecto específico para un almacén editorial, poniendo énfasis en el desarrollo del proyecto a través de pequeños pasos o ciclos de desarrollo.

Con la experiencia adquirida en este proyecto, es notorio que las prácticas de XP son aplicables a proyectos de software bajo las siguientes condiciones:

- La utilidad y aplicabilidad de las reglas se encuentra determinada por muchos factores ajenos al equipo de desarrollo como: tipo de cliente que requiere el software, tamaño y alcances del proyecto de software, cotización del software por duración o alcance, etc.
- Es importante adaptar cada práctica XP al ambiente de trabajo propio del equipo de desarrollo para lograr una buena adaptación a este enfoque de desarrollo y así obtener el mayor provecho de las prácticas.

Las prácticas y los principios de XP resultan una forma innovadora de desarrollar software que implica una gran cantidad de cambios en el entorno del equipo de desarrollo, así como en las prácticas y políticas de las organizaciones que requieren software de cada vez mayor calidad producido eficazmente. En México se requiere aún tiempo para que estas prácticas puedan ser adoptadas de lleno, debido a que una gran cantidad de empresas aún

no aprecian los sistemas de información como una fuente importante de ventajas competitivas. En el momento en que los beneficios de los sistemas de información sean tomados en cuenta tanto por los grandes consorcios como por pequeñas empresas métodos de desarrollo como XP serán utilizados masivamente fomentando la mejora en el desarrollo de software.

El uso de diagramas UML en el presente trabajo se debe principalmente a las siguientes razones:

- Utilizar el lenguaje para diseñar ciertos aspectos del proyecto para el almacén editorial, con la idea de mejorar la comprensión de los conceptos.
- Apreciar ciertos diagramas UML y de esta forma dar una idea de su utilización, ya que este lenguaje de modelado se está convirtiendo en un estándar para la industria y su uso como herramienta de diseño resulta cada vez más necesaria.

En el caso del lenguaje UML, se utilizaron sólo algunos diagramas debido a que se quería mostrar mediante diagramas ciertos aspectos del proyecto. Es cierto que UML es una herramienta poderosa de modelado, pero en este trabajo se prefirió dar énfasis a XP como aproximación práctica al desarrollo de software y dejar UML como complemento para dar una idea más clara de los conceptos.

Una de las ventajas de la utilización de los diagramas UML en conjunto con XP, es que si los diagramas son bien aprovechados permiten comparar el avance entre diferentes etapas del período, resaltando de manera gráfica la evolución que sufre un proyecto a través de los ciclos cortos de desarrollo.

A su vez, en estos tiempos de globalización el tema de los almacenes y su implicación en los negocios resulta particularmente actual, debido a que cada vez hay mayor competencia entre las empresas y deseo de mejorar la calidad de productos y servicios.



TESIS CON
FALLA DE ORIGEN

Bibliografía

- [1] Beck, Kent: *Extreme Programming Explained*, Addison-Wesley, 1999
- [2] McBreen, Pete: *Questioning Extreme Programming*, Addison-Wesley, 2003
- [3] www.extremeprogramming.org
- [4] www.xprogramming.com
- [5] Booch, Grady; Rumbaugh, James; Jacobson, Ivar: *The Unified Modeling Language User Guide*, Addison-Wesley, 1999
- [6] Muller, Pierre Alain: *Instant UML*, Wrox Press Ltd, 1997
- [7] Yourdon, Edward: *Análisis Estructurado Moderno*, Prentice Hall Hispanoamericana, 1993
- [8] Kendall, Kenneth E; Kendall, Julie E: *Análisis y Diseño de Sistemas*, Prentice Hall Hispanoamericana, 1991
- [9] OMG: *Unified Modeling Language Specification ver. 1.4*, www.rational.com, Septiembre 2001
- [10] Ackerman, Kenneth; Nieto, Alejandro: *Almacenamiento Productivo: Herramienta de Logística Internacional*, Ackerman Publications, 2000
- [11] Real Academia Española: *Diccionario de la Lengua Española*, www.rae.es

Índice alfabético

- "Gran diseño previo", 12
- 40 horas semanales, 8
- acción, 29
- actividad, 29
- actor, 22
 - tipos de, 23
- agregación, 21
- almacén, 33, 45
 - capacidad ideal, 36
 - capacidad real, 36
 - características generales, 46
 - procesos en el, 48
 - tipos de libros en el, 46
- almacenamiento, 33
 - cuantificación de espacio de, 36
 - equipo de, 35
 - equipo en el almacén, 47
 - especializado, 34
 - por unidades, 34
- asociación, 20
- códigos de barras, 40
- caso de uso, 24
- clase, 17
 - jerarquías de, 18
 - representación en UML, 20
- cliente en el sitio de desarrollo, 9
- composición, 22
- comunicación, 3
- cuarta iteración, 93
 - conclusiones, 101
- desarrollo, 94
 - requerimientos, 94
- Diagrama
 - de objeto, 24
 - diagrama UML,
 - de clase, 20
- Diagramas
 - de actividades, 30
 - de casos de uso, 22
 - de colaboración, 25
 - de componentes, 30
 - de estado, 28
 - de instalación, 31
 - de secuencia, 27
 - diseño simple, 6
- ejemplar, 45
- empleados a pie, 49
- enfoque orientado a objetos, 15
- equipo de almacenamiento, 35
 - útil, 36
 - picking, 36
 - racks, 35
- equipo de negocios, 4
- equipo de transporte, 36
 - tipos de, 37
- equipo técnico, 4
- especialización, 18
- estándares de codificación, 9
- estándares de codificación, 81
- estado, 28

- eventos, 28
- factor de desarrollo, 11, 12
- FIFO, 40
- generalización, 18, 22
- herencia, 17
- identificación electrónica, 40
- integración continua, 8, 81
- integración de código, 13
- juego de planeación, 4
- lector óptico, 51
- libros
 - de texto, 46
 - obra general, 46
- liga, 25
- lista de surtido, 52
 - global, 52
 - individual, 52
- manejo de carga, 36
- mensaje, 17
 - representación en UML, 25
 - tipos de transmisión de, 28
- mercancía
 - embarque de, 37
 - recepción de, 37
- metáfora, 6
- montacargas, 37, 83
 - tipos de, 37
- montacarguistas, 49
- multiplicidad de asociaciones, 21
- objeto, 16
 - activación de un, 26
 - características, 16
 - comportamiento de un, 16
 - composición de, 25
 - estado de un, 16
 - identidad de un, 17
 - representación en UML, 24
 - orden de recorrido, 70
 - orientado a objetos
 - enfoque, 15, 18
 - paquetes, 47
 - pedidos, 50
 - estados, 50
 - pequeñas entregas, 5
 - picking, 36, 47
 - planeación, 4, 11, 66
 - polimorfismo, 18
 - posición, 60
 - posiciones, 47
 - prácticas de XP
 - 40 horas semanales, 8
 - cliente en el sitio de desarrollo, 9
 - diseño simple, 6
 - estándares de codificación, 9
 - integración continua, 8
 - juego de planeación, 4
 - metáfora, 6
 - pequeñas entregas, 5
 - programación por pares, 7
 - propiedad colectiva, 8
 - pruebas, 6
 - refabricación, 7
 - primera iteración, 54
 - conclusiones, 65
 - desarrollo, 55
 - requerimientos, 54
 - Programación Extrema, 3
 - prácticas, 4
 - secuencia de desarrollo extremo, 9
 - valores, 3
 - programación por pares, 80

- programación por pares , 7
- propiedad colectiva, 8
- pruebas, 6, 13
- rack, 35, 47
- recolección
 - de caja abierta, 38
 - de carga unitaria, 38
 - métodos de, 38
 - moderada, 38
 - por cajas, 38
 - por lotes, 38
 - por oleadas, 38
 - por zonas, 38
- refabricación, 7, 80
- reinicio del ciclo, 14
- requerimiento de usuario, 10
- retroalimentación, 4
- reuniones diarias, 13
- secuencia de desarrollo extremo, 9
 - iteraciones, 11
 - desarrollo, 13
 - diseño, 12
 - planeación, 11
 - pruebas, 13
 - planeación, 10
 - reinicio del ciclo, 14
 - segunda iteración, 67
 - conclusiones, 78
 - desarrollo, 69
- semanas de programación ideal, 10
- simplicidad, 4
- sistemas
 - de información, 39
 - de localización de mercancía, 39, 50
- sistemas de localización
 - ventajas, 39
- stock mayor, 47, 82
- surtido, 38
 - categorías, 38
 - cuarta iteración, 93
 - metas del sistema, 53
 - primera iteración, 54
 - proceso en el almacén, 51
 - segunda iteración, 67
 - tercera iteración, 81
- título, 45
- tareas de programación, 11
- tarima, 34
- tercera iteración, 81
 - conclusiones, 91
 - desarrollo, 82
 - requerimientos, 82
- transición, 28
- UML
 - diagramas, 19
 - historia, 18
 - objetivos, 19
- valentía, 4
- valores de XP
 - comunicación, 3
 - retroalimentación, 4
 - simplicidad, 4
 - valentía, 4
- zona, 82