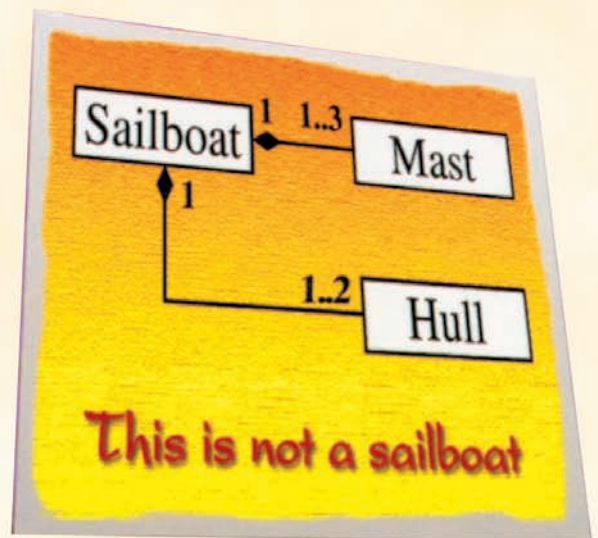


2ª Edición

# UML y Patrones

Una introducción al análisis y diseño orientado a objetos y al proceso unificado

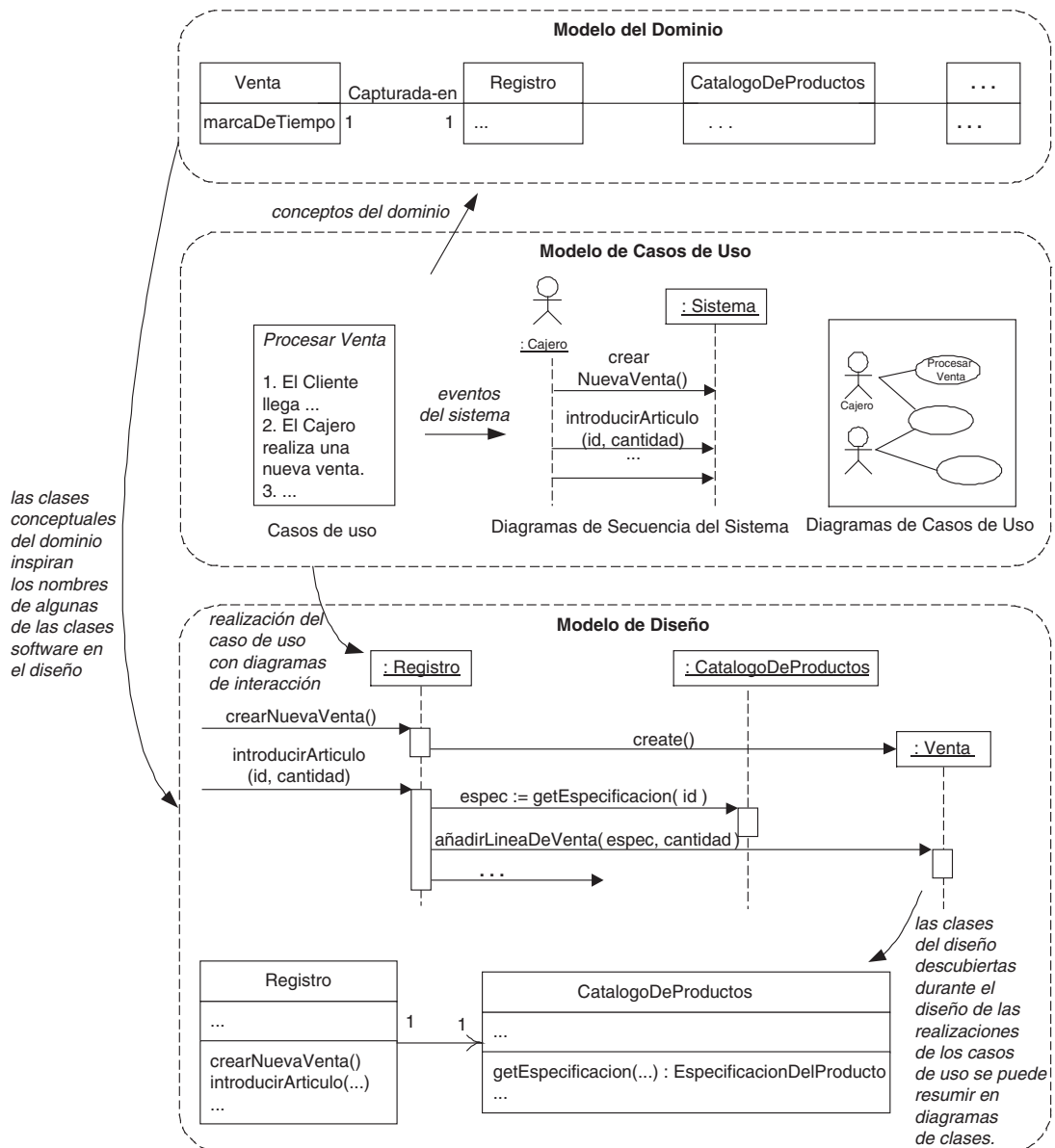
[www.librosite.net/larman](http://www.librosite.net/larman)



## Muestra de los artefactos del Proceso Unificado y evolución temporal (c-comenzar; r-refinar)

Disciplina	Artefacto Iteración →	Inicio I1	Elab. E1...En	Const. C1...Cn	Trans. T1...T2
Modelado del Negocio	Modelo del Dominio		c		
Requisitos	Modelo de Casos de Uso	c	r		
	Visión	c	r		
	Especificación Complementaria	c	r		
	Glosario	c	r		
Diseño	Modelo de Diseño		c	r	
	Documento de Arquitectura SW		c		
	Modelo de Datos		c	r	
Implementación	Modelo de Implementación		c	r	r
Gestión del Proyecto	Plan de Desarrollo SW	c	r	r	r
Pruebas	Modelo de Pruebas		c	r	
Entorno	Marco de Desarrollo	c	r		

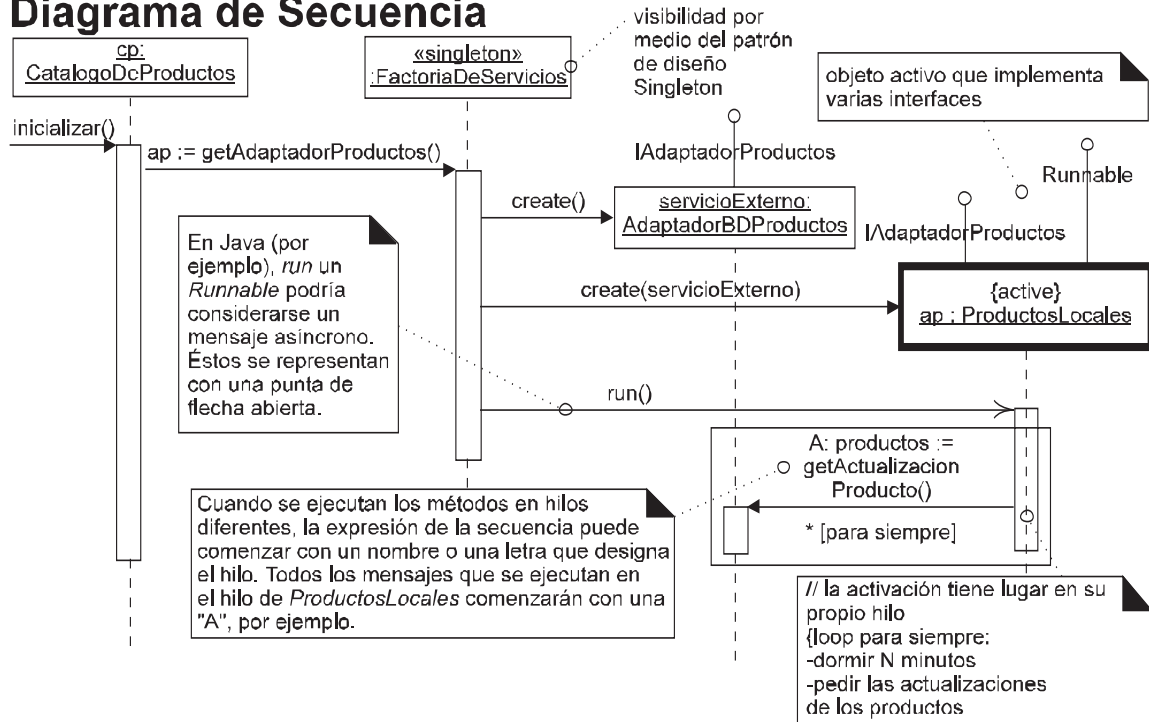
## Muestra de las relaciones entre los artefactos del Proceso Unificado



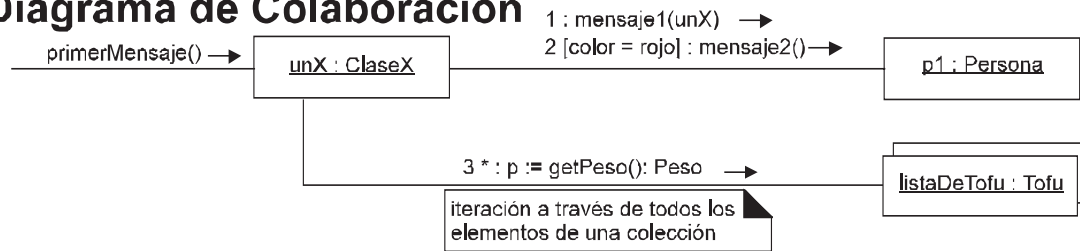
## Patrones de Software Generales para la Asignación de Responsabilidades (GRASP)

<i>Patrón</i>	<i>Descripción</i>
Experto en Información	<p>¿Un principio general del diseño de objetos y la asignación de responsabilidades?</p> <p>Asigne una responsabilidad al experto en información, —la clase que tiene la información necesaria para llevar a cabo la responsabilidad.</p>
Creador	<p>¿Quién crea? (Nótese que la Factoría es una solución alternativa frecuente.)</p> <p>Asigne a la clase B la responsabilidad de crear una instancia de la clase A si se cumple alguno de los puntos siguientes:</p> <ol style="list-style-type: none"> <li>1. B contiene a A</li> <li>2. B agrega a A</li> <li>3. B tiene los datos de inicialización de A</li> <li>4. B registra a A</li> <li>5. B utiliza estrechamente a A</li> </ol>
Controlador	<p>¿Quién gestiona un evento del sistema?</p> <p>Asigne la responsabilidad de gestionar un mensaje de un evento del sistema a una clase que represente una de estas opciones:</p> <ol style="list-style-type: none"> <li>1. Representa el sistema global, dispositivo o un subsistema (controlador de fachada).</li> <li>2. Representa un escenario de caso de uso en el que tiene lugar el evento del sistema (controlador de caso de uso o sesión).</li> </ol>
Bajo Acoplamiento (evaluativo)	<p>¿Cómo dar soporte a las bajas dependencias y al incremento de la reutilización?</p> <p>Asigne responsabilidades de manera que el acoplamiento (innecesario) se mantenga bajo.</p>
Alta Cohesión (evaluativo)	<p>¿Cómo mantener manejable la complejidad?</p> <p>Asigne responsabilidades de manera que la cohesión permanezca alta.</p>
Polimorfismo	<p>¿Quién es el responsable cuando el comportamiento varía en función del tipo?</p> <p>Cuando las alternativas o comportamientos relacionados varían según el tipo (clase), asigne la responsabilidad del comportamiento —utilizando operaciones polimórficas— a los tipos para los que varía el comportamiento.</p>
Fabricación Pura	<p>¿Quién es el responsable cuando está desesperado, y no quiere violar los principios de alta cohesión y bajo acoplamiento?</p> <p>Asigne un conjunto altamente cohesivo de responsabilidades a una clase de “comportamiento” artificial o de conveniencia que no representa un concepto del dominio del problema —algo inventado—, para dar soporte a la alta cohesión, bajo acoplamiento y la reutilización.</p>
Indirección	<p>¿Cómo asignar responsabilidades para evitar el acoplamiento directo?</p> <p>Asigne la responsabilidad a un objeto intermedio para mediar entre otros componentes o servicios, de manera que no se acoplan directamente.</p>
Variaciones Protegidas	<p>¿Cómo asignar responsabilidades a los objetos, subsistemas, y sistemas de manera que las variaciones o inestabilidad en estos elementos no influya de manera no deseable en otros elementos?</p> <p>Identifique los puntos de variaciones predecibles o inestabilidad; asigne las responsabilidades para crear una “interfaz” estable alrededor de ellos.</p>

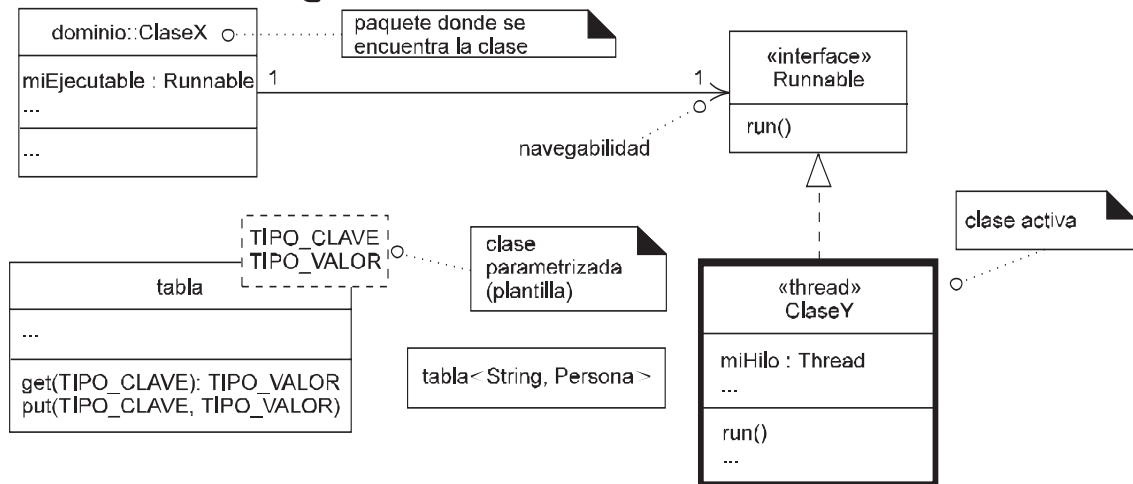
## Diagrama de Secuencia



## Diagrama de Colaboración

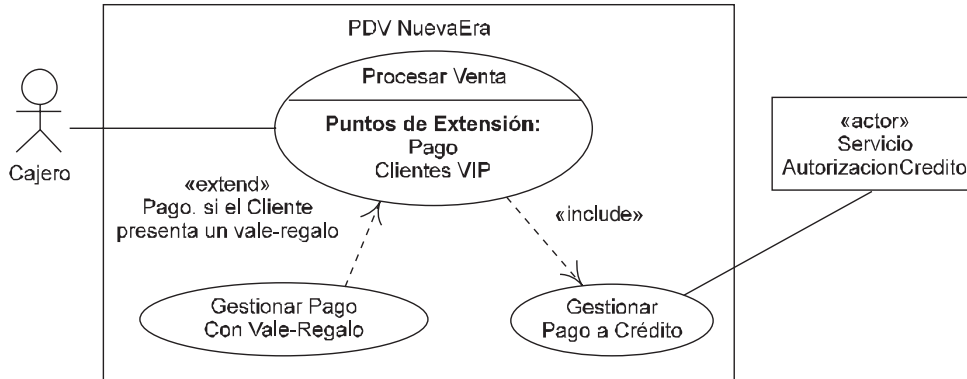


## Notación del Diagrama de Clase Adicional

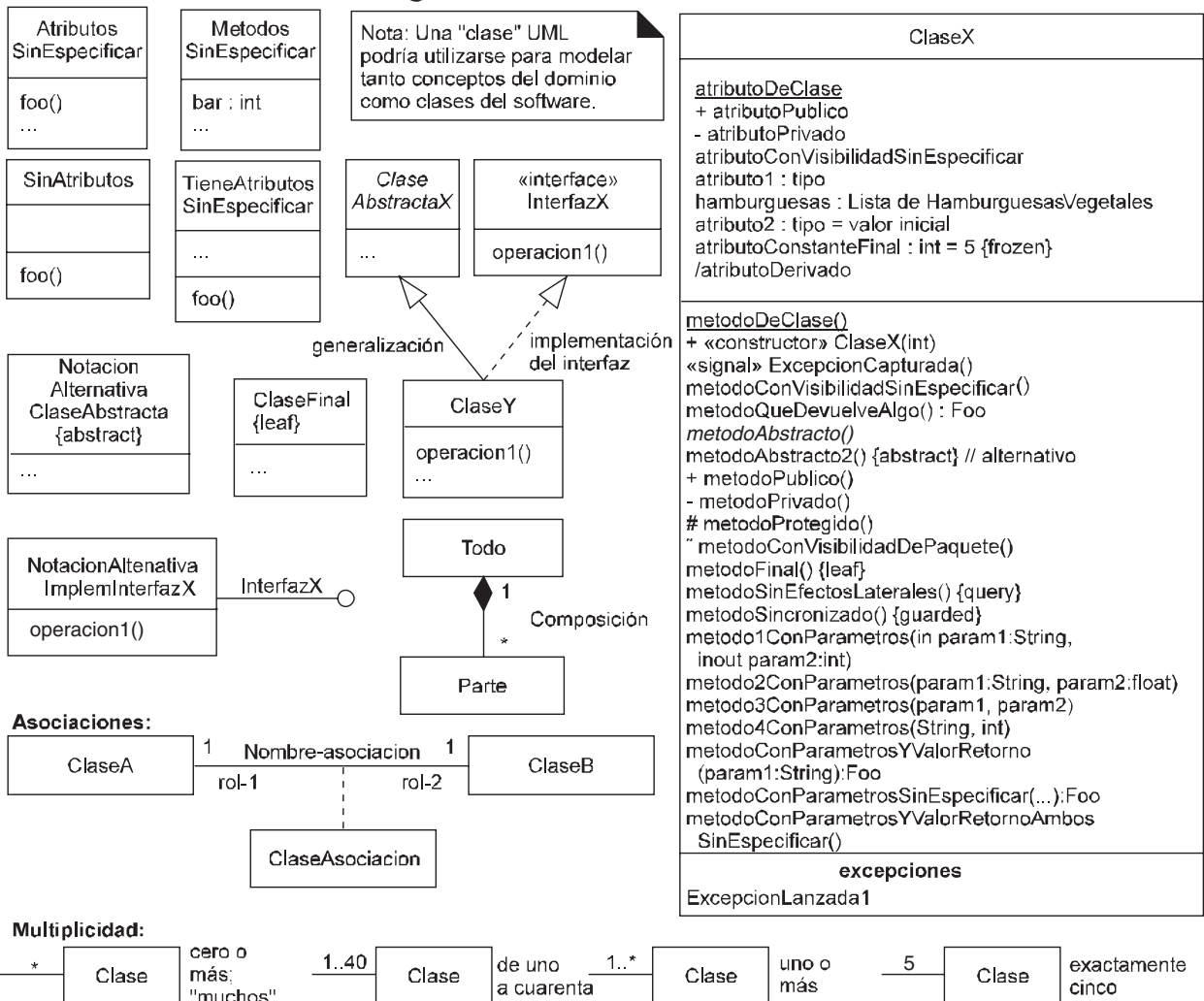


## Notación UML de Muestra

### Diagrama de Caso de Uso



### Diagrama de Clases



# **UML Y PATRONES**

**UNA INTRODUCCIÓN AL ANÁLISIS  
Y DISEÑO ORIENTADO A OBJETOS  
Y AL PROCESO UNIFICADO**

Segunda edición



# UML Y PATRONES

## UNA INTRODUCCIÓN AL ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS Y AL PROCESO UNIFICADO

Segunda edición

**CRAIG LARMAN**

**Traducción:**

**Begoña Moros Valle**  
*Universidad de Murcia*

**Supervisión de la traducción y revisión técnica:**

**Jesús García Molina**  
*Universidad de Murcia*



Madrid • México • Santafé de Bogotá • Buenos Aires • Caracas • Lima  
Montevideo • San Juan • San José • Santiago • São Paulo • White Plains



**LARMAN, C.**

*UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. Segunda edición*

PEARSON EDUCACIÓN, S.A., Madrid, 2003

ISBN eBook: 978-84-832-2927-9

Materia: Informática 681.3

Formato 195 × 250

Páginas: 624

Todos los derechos reservados.

Queda prohibida, salvo excepción prevista en la ley, cualquier forma de reproducción, distribución, comunicación pública y transformación de esta obra sin contar con autorización de los titulares de propiedad intelectual. La infracción de los derechos mencionados puede ser constitutiva de delito contra la propiedad intelectual (*arts. 270 y sgts. Código Penal*).

#### DERECHOS RESERVADOS

© 2003 respecto a la segunda edición en español por:

PEARSON EDUCACIÓN, S.A.

Núñez de Balboa, 120

28006 Madrid

**LARMAN, C.**

*UML Y PATRONES. Una introducción al análisis y diseño orientado a objetos y al proceso unificado. Segunda edición*

**ISBN: 84-205-3438-2**

Depósito Legal: M-

PRENTICE HALL es un sello editorial autorizado de PEARSON EDUCACIÓN, S.A.

*Traducido de:*

*APPLYING UML AND PATTERNS: An Introduction to Object-Oriented Analysis and Design and the Unified Process*, Second edition, by Craig Larman.

Published by Pearson Education, Inc. Publishing as Prentice Hall PTR.

Prentice Hall, Inc.

© 2002 All rights reserved.

ISBN: 0-13-092569-1

*Edición en español:*

#### **Equipo editorial:**

**Editor:** David Fayerman Aragón

**Técnico editorial:** Ana Isabel García

#### **Equipo de producción:**

**Director:** José Antonio Clares

**Técnico:** José Antonio Hernán

**Diseño de cubierta:** equipo de diseño de PEARSON EDUCACIÓN, S.A.

**Composición:** COPIBOOK, S.L.

**Impreso por:**

IMPRESO EN ESPAÑA - PRINTED IN SPAIN



*Para Julie*

*Sin su apoyo, esto no habría sido posible.*

*Para Haley y Hannah*

*Gracias por aguantar a un papá distraído, ¡otra vez!*



# RESUMEN DE CONTENIDO

## Parte 1: Introducción

1. Análisis y diseño orientado a objetos .....	3
2. Desarrollo iterativo y el Proceso Unificado .....	13
3. Caso de estudio: el sistema de punto de venta NuevaEra .....	27

## Parte 2: Inicio

4. Inicio .....	33
5. Comprensión de los requisitos .....	39
6. Modelo de Casos de Uso: escritura de requisitos en contexto .....	43
7. Identificación de otros requisitos .....	79
8. Del inicio a la elaboración .....	103

## Parte 3: Elaboración en la Iteración 1

9. Modelo de Casos de Uso: representación de los diagramas de secuencia del sistema .....	113
10. Modelo del Dominio: visualización de conceptos .....	121
11. Modelo del Dominio: añadir asociaciones .....	145
12. Modelo del Dominio: añadir atributos .....	157
13. Modelo de Casos de Uso: añadir detalles con los contratos de las operaciones .....	167
14. De los requisitos al diseño en esta iteración .....	181
15. Notación de los diagramas de interacción .....	185
16. GRASP: diseño de objetos con responsabilidades .....	201
17. Modelo de Diseño: realización de los casos de uso con los patrones GRASP .....	231
18. Modelo de Diseño: determinación de la visibilidad .....	261

## VIII RESUMEN DE CONTENIDO

19. Modelo de Diseño: creación de los diagramas de clases de diseño . . . . .	267
20. Modelo de Implementación: transformación de los diseños en código . .	281

### Parte 4: Elaboración en la Iteración 2

21. La Iteración 2 y sus requisitos . . . . .	291
22. GRASP: más patrones para asignar responsabilidades . . . . .	305
23. Diseño de las realizaciones de casos de uso con los patrones de diseño GoF .	321

### Parte 5: Elaboración en la Iteración 3

24. La Iteración 3 y sus requisitos . . . . .	359
25. Relaciones entre casos de uso . . . . .	361
26. Modelado de la generalización . . . . .	369
27. Refinamiento del modelo del dominio . . . . .	383
28. Añadir nuevos DSSs y contratos . . . . .	401
29. Modelado del comportamiento con diagramas de estado . . . . .	407
30. Diseño de la arquitectura lógica con patrones . . . . .	417
31. Organización de los paquetes de los modelos de diseño e implementación .	443
32. Introducción al análisis arquitectural y el SAD . . . . .	451
33. Diseño de más realizaciones de casos de uso con objetos y patrones . . . .	473
34. Diseño de un framework de persistencia con patrones . . . . .	501

### Parte 6: Temas especiales

35. Sobre el dibujo de diagramas y las herramientas . . . . .	531
36. Introducción a cuestiones relacionadas con la planificación iterativa y el proyecto . . . . .	539
37. Comentarios acerca del desarrollo iterativo y el UP . . . . .	553
38. Más notación UML . . . . .	567

# CONTENIDO

<b>Presentación</b> .....	XIX
<b>Prólogo</b> .....	XXI

## Parte 1: Introducción

<b>Capítulo 1: Análisis y diseño orientado a objetos</b> .....	3
Aplicación de UML y patrones en el A/DOO .....	3
Asignación de responsabilidades .....	5
¿Qué es análisis y diseño? .....	6
¿Qué son el análisis y diseño orientados a objetos? .....	6
Un ejemplo .....	7
UML .....	10
Lecturas adicionales .....	10
<b>Capítulo 2: Desarrollo iterativo y el Proceso Unificado</b> .....	13
La idea más importante del UP: desarrollo iterativo .....	14
Conceptos y buenas prácticas del UP adicionales .....	18
Las fases del UP y términos orientados a la planificación .....	19
Las disciplinas del UP (eran flujos de trabajo) .....	20
Adaptación del proceso y el Marco de Desarrollo .....	22
El UP ágil .....	23
El ciclo de vida “en cascada” secuencial .....	24
No se entendió el UP cuando... ..	25
Lecturas adicionales .....	25
<b>Capítulo 3: Caso de estudio: el sistema de punto de venta NuevaEra</b> .	27
El sistema de punto de venta NuevaEra .....	27
Capas arquitectónicas y el énfasis del caso de estudio .....	28
Estrategia del libro: aprendizaje y desarrollo iterativo .....	29

## Parte 2: Inicio

<b>Capítulo 4: Inicio</b>	33
Inicio: una analogía	34
La fase de inicio podría ser muy breve	35
¿Qué artefactos podrían crearse en la fase de inicio?	35
No se entendió la fase de inicio cuando...	36
<b>Capítulo 5: Comprensión de los requisitos</b>	39
Tipos de requisitos	40
Lecturas adicionales	41
<b>Capítulo 6: Modelo de Casos de Uso: escritura de requisitos en contexto</b>	43
Objetivos e historias	44
Antecedentes	44
Casos de uso y valor añadido	45
Casos de uso y requisitos funcionales	46
Tipos de casos de uso y formatos	46
Ejemplo completo: Procesar Venta	47
Explicación de las secciones	52
Objetivos y alcance de un caso de uso	56
Descubrimiento de actores principales, objetivos y casos de uso	60
Enhorabuena: se han escrito los casos de uso y no son perfectos	64
Escritura de casos de uso en un estilo esencial independiente de la interfaz de usuario	65
Actores	67
Diagramas de casos de uso	68
Requisitos en contexto y lista de características de bajo nivel	70
Los casos de uso no son orientados a objetos	71
Casos de uso en el UP	72
Caso de estudio: casos de uso en la fase de inicio de NuevaEra	76
Lecturas adicionales	76
Artefactos UP y contexto del proceso	77
<b>Capítulo 7: Identificación de otros requisitos</b>	79
Ejemplos del PDV NuevaEra	80
Ejemplo NuevaEra: Especificación Complementaria (Parcial)	80
Comentario: Especificación Complementaria	84
Ejemplo NuevaEra: Visión (Parcial)	87
Comentario: Visión	90
Ejemplo NuevaEra: un Glosario (Parcial)	94
Comentario: Glosario (Diccionario de Datos)	95
Especificaciones fiables: ¿un Oxímoron?	96
Artefactos disponibles en el sitio web del proyecto	97
¿Poco UML durante la fase de inicio?	97
Otros artefactos de requisitos en el UP	97