

Contenidos

Prólogo	XIX	1.4.2 Formación de un ingeniero de software	11
Prefacio	XXI	1.5 Nuestro caso testigo.....	11
Guía de lectura.....	XXIII	1.5.1 Descripción del problema a resolver.....	11
		1.5.2 Observaciones obtenidas en la empresa cliente	12
		1.5.3 Característica de la empresa desarrolladora	13
		1.6 Conclusión.....	14
		1.7 Contenido de la página web de apoyo.....	14
		1.7.1 Mapa conceptual	14
		1.7.2 Autoevaluaciones	14
		1.7.3 Presentaciones en Power Point (exclusivo para el docente).....	14
		1.8 Referencias	14
Parte I - Introducción		Capítulo 2 – Condiciones de trabajo en el Desarrollo de Software	
Capítulo 1 - Evolución		2.1 Proyectos de Software en diferentes ámbitos.....	18
1.1 Introducción.....	2	2.1.1 Desarrollo de sistemas	18
1.2 Los hitos en la evolución histórica del desarrollo de software	2	2.1.2 Desarrollo de productos	18
1.2.1 Los sistemas integrados hardware-software	2	2.1.3 Desarrollo remoto.....	20
1.2.2 Los primeros sistemas de software independientes	2	2.1.4 Software factory	22
1.2.3 La crisis del software	4	2.1.5 Proyectos de código abierto	23
1.2.4 La creación de la Ingeniería de Software	4	2.1.6 Servicios para clientes internos	25
1.2.5 La aparición de la PC	5	2.2 Proyectos de Software y las personas que participan	26
1.2.6 La interconexión de las PCs	5	2.2.1 Trabajo en equipo	27
1.2.7 El surgimiento de la Internet	6	2.3 Proyectos de software y las condiciones de trabajo.....	28
1.2.8 La evolución de Internet y las arquitecturas corporativas	6	2.3.1 Condiciones versus procesos.....	28
1.2.9 El futuro cercano.....	6	2.3.2 Condiciones y metodologías	29
1.3 Problemas y soluciones.....	7	2.3.3 Manifestación en la gestión de los proyectos.....	29
1.3.1 Fallas, malas estimaciones y metodologías	7	2.4 Buenas prácticas	30
1.3.2 Herramientas de desarrollo.....	8	2.4.1 Trabajo iterativo	30
1.3.3 El concepto de calidad en el software.....	8	2.4.2 Comunicación fluida y cerrada	31
1.3.4 Venta de proyectos de desarrollo de software.....	8	2.4.3 Entregables frecuentes.....	31
1.3.5 Visibilidad de los proyectos de desarrollo	9	2.4.4 Reflexión y mejoras	31
1.3.6 Calidad versus velocidad de desarrollo ..	9		
1.4 Incumbencias de la nueva ingeniería	10		
1.4.1 Conocimientos de procesos versus conocimientos tecnológicos	10		

2.4.5 Seguridad personal	32	3.7 Programación Genérica	43
2.4.6 Focalización en tareas de los proyectos	32	3.7.1 Historia.....	43
2.4.7 Fácil acceso a los clientes	32	3.7.2 Conceptos	44
2.4.8 Ambiente de desarrollo con integración frecuente.....	32	3.7.3 Su uso en la actualidad	44
2.4.9 Colaboración de la organización a partir del compromiso con los proyectos.....	32	3.8 El modelo de múltiple paradigma	45
2.4.10 Alta visibilidad de los proyectos.....	33	3.8.1 Historia.....	45
2.4.11 Inducción a los proyectos	33	3.8.2 Conceptos	45
2.4.12 Asignación de tareas clausurando funcionalidades	33	3.8.3 Su uso en la actualidad	45
2.4.13 Arquitecto programador	34	3.9 Ejemplo	45
2.4.14 Programación compartida	34	3.9.1 Especificación	45
2.5 Conclusión.....	34	3.9.2 Solución	46
2.6 Contenido de la página web de apoyo.....	34	3.10 Lenguajes de programación.....	46
2.6.1 Mapa conceptual	34	3.10.1 Clasificación	46
2.6.2 Autoevaluaciones	34	3.11 Métricas del grado de utilización.....	49
2.6.3 Presentaciones en Power Point (exclusivo para el docente).....	34	3.12 Conclusiones.....	50
2.7 Referencias	35	3.13 Contenido de la página web de apoyo.....	51
		3.13.1 Mapa conceptual	51
		3.13.2 Autoevaluaciones	51
		3.13.3 Presentaciones en Power Point (exclusivo para el docente).....	51
		3.14 Referencias	51

Capítulo 3 – Paradigmas y lenguajes de programación

3.1 Paradigmas de programación	38
3.2 Programación con Procedimientos.....	39
3.2.1 Historia.....	39
3.2.2 Conceptos	39
3.2.3 Su uso en la actualidad	39
3.3 Programación Funcional.....	40
3.3.1 Historia.....	40
3.3.2 Conceptos	40
3.3.3 Su uso en la actualidad	40
3.4 Programación Lógica.....	40
3.4.1 Historia.....	40
3.4.2 Conceptos	41
3.4.3 Su uso en la actualidad	41
3.5 Programación Orientada a Objetos	42
3.5.1 Historia.....	42
3.5.2 Conceptos	42
3.5.3 Su uso en la actualidad	42
3.6 Programación Orientada a Aspectos.....	43
3.6.1 Historia.....	43
3.6.2 Conceptos	43
3.6.3 Su uso en la actualidad	43

Parte II – Proceso de desarrollo de Software / Metodologías

Capítulo 4 – Metodologías de desarrollo de software

4.1 Metodologías.....	54
4.1.1 Historia.....	54
4.1.2 Clasificación	54
4.2 Metodologías conducidas por los planes	55
4.2.1 Cascada.....	55
4.2.2 Prototipado	57
4.2.3 DRA (Desarrollo Rápido de Aplicaciones).....	58
4.2.4 Incremental	61
4.2.5 Espiral	62
4.3 Metodologías ágiles.....	64
4.4 Diferencias de enfoque.....	65
4.5 Diferencias de aplicabilidad	68
4.6 Conclusión.....	69
4.7 Contenido de la página web de apoyo.....	70
4.7.1 Mapa conceptual	70
4.7.2 Autoevaluaciones	70

4.7.3 Presentaciones en Power Point (exclusivo para el docente).....	70
4.8 Referencias.....	70

Capítulo 5 – Metodologías conducidas por los planes

5.1 El Proceso Unificado de Desarrollo de Software.....	72
5.2 Conducido por los planes	74
5.3 El flujo de trabajo	75
5.3.1 Las fases y los hitos.....	75
5.4 Las iteraciones	77
5.5 El organigrama, los roles y actividades	78
5.6 Roles asociados a la organización	80
5.6.1 Roles derivados de la forma de trabajo	80
5.7 Los activos como nexo entre los roles	82
5.7.1 Los hitos y entregables	82
5.8 Planes	86
5.8.1 Preventa.....	86
5.9 Plan de proyecto.....	87
5.10 Otros planes	88
5.10.1 Plan de administración de la configuración	88
5.10.2 Plan de administración de la calidad	88
5.11 Activos de trabajo	89
5.11.1 Especificación de arquitectura de software	89
5.12 Conclusión.....	89
5.13 Contenido de la página web de apoyo.....	90
5.13.1 Mapa conceptual	90
5.13.2 Autoevaluaciones	90
5.13.3 Presentaciones en Power Point (exclusivo para el docente).....	90
5.14 Referencias	90

Capítulo 6 – Metodologías ágiles

6.1 Extreme Programming y Scrum	92
6.1.1 Procesos de gestión y procesos técnicos	92
6.2 Extreme Programming (XP)	93
6.2.1 Valores	93
6.2.2 Principios	93

6.2.3 Prácticas	94
6.2.4 Flujo de trabajo	97
6.3 Scrum.....	98
6.4 Integración Xp – Scrum	100
6.5 Ágiles	101
6.6 El flujo de trabajo	102
6.6.1 Ciclo de vida	102
6.7 Organigrama y seguimiento del proyecto	105
6.8 Conclusión.....	107
6.9 Contenido de la página web de apoyo.....	107
6.9.1 Mapa conceptual	107
6.9.2 Autoevaluaciones	107
6.9.3 Presentaciones en Power Point (exclusivo para el docente).....	107
6.10 Referencias	107

Parte III – Modelos y Documentación

Capítulo 7 – El lenguaje UML y sus modos de utilización

7.1 Introducción.....	110
7.1.1 ¿Qué es UML?.....	110
7.1.2 Preguntas frecuentes	110
7.2 Evolución histórica	111
7.3 ¿Cómo es utilizado en la comunidad informática?.....	113
7.3.1 Dibujo (<i>Sketch</i>)	113
7.3.2 Plan (<i>Blueprint</i>)	113
7.3.3 Lenguaje de programación	113
7.4 Perspectivas.....	114
7.4.1 Problema (Concepción y Análisis).....	114
7.4.2 Solución (Diseño).....	114
7.4.3 Respuestas a las preguntas frecuentes	114
7.4.4 Madurez y nivel de conocimiento de los desarrolladores	117
7.5 Diagramas	117
7.5.1 Diagramas Estáticos.....	117
7.5.2 Diagramas Dinámicos.....	118
7.6 Relación con el proceso de desarrollo	118
7.7 Selección del modo de uso	119
7.8 Contenido de la página web de apoyo.....	119
7.8.1 Mapa conceptual	119
7.8.2 Autoevaluaciones	119

7.8.3 Presentaciones en Power Point (exclusivo para el docente).....	119
7.9 Referencias.....	119

Capítulo 8 – El lenguaje UML - Diagramas estáticos

8.1 Introducción.....	122
8.2 Diagramas	122
8.3 Clases	123
8.4 Paquetes.....	124
8.5 Componentes	125
8.6 Instalación	126
8.7 Ejemplo	127
8.8 Conclusión.....	131
8.9 Contenido de la página web de apoyo.....	131
8.9.1 Mapa conceptual	131
8.9.2 Autoevaluaciones	131
8.9.3 Presentaciones en Power Point (exclusivo para el docente).....	131
8.10 Referencias.....	131

Capítulo 9 – El lenguaje UML - Diagramas dinámicos

9.1 Introducción.....	134
9.2 Actividades.....	134
9.3 Casos de uso.....	135
9.4 Interacción.....	137
9.5 Estado.....	138
9.6 Ejemplo	138
9.7 Conclusión.....	141
9.8 Contenido de la página web de apoyo.....	141
9.8.1 Mapa conceptual	141
9.8.2 Autoevaluaciones	141
9.8.3 Presentaciones en Power Point (exclusivo para el docente).....	141
9.9 Referencias.....	141

Parte IV – Relevamiento, modelado y
análisis de requerimientos

Capítulo 10 – Relevamiento de requerimientos

10.1 Relevamiento de requerimientos.....	144
---	------------

10.1.1 Objetivos	144
10.1.2 Técnicas de relevamiento	148
10.2 Casos de uso.....	150
10.2.1 Paquetes funcionales	150
10.3 Organización del trabajo con requerimientos.....	161
10.3.1 Planificación	161
10.4 Priorización de requerimientos.....	162
10.5 Conclusión.....	166
10.6 Contenido de la página web de apoyo...	166
10.6.1 Mapa conceptual	166
10.6.2 Autoevaluaciones	166
10.6.3 Presentaciones en Power Point (exclusivo para el docente).....	166
10.7 Referencias	166

Capítulo 11 – Análisis de requerimientos

11.1 Análisis de requerimientos	168
11.1.1 Objetivos	168
11.1.2 Modelo del comportamiento y control	171
11.1.3 Trazabilidad de los requerimientos.....	175
11.2 Conclusión	176
11.3 Contenido de la página web de apoyo...	177
11.3.1 Mapa conceptual	177
11.3.2 Autoevaluaciones	177
11.3.3 Presentaciones en Power Point (exclusivo para el docente).....	177
11.4 Referencias	177

Capítulo 12 – Pruebas a los requerimientos

12.1 Pruebas a los requerimientos	180
12.2 Validación	180
12.2.1 Validación de la aplicación.....	180
12.2.2 Validación del modelo de negocio.....	182
12.3 Verificación	187
12.3.1 Casos de pruebas.....	187
12.3.2 Ciclo de vida y herramientas....	194
12.4 Conclusión	196
12.5 Contenido de la página web de apoyo...	197
12.5.1 Mapa conceptual	197
12.5.2 Autoevaluaciones	197

12.5.3 Presentaciones en Power Point (exclusivo para el docente).....	197
12.6 Referencias	197

Parte V – Arquitectura y diseño de software

Capítulo 13 – Arquitectura de software

13.1 Definiciones	200
13.1.1 El concepto de arquitectura	200
13.1.2 Documentación de la arquitectura	208
13.1.3 Vista estática.....	209
13.2 Selección de componentes.....	211
13.2.1 Plataformas marco (<i>Framework</i>)	211
13.2.2 Comunicación con sistemas externos.....	213
13.2.3 Atributos de calidad	217
13.2.4 Pruebas a la arquitectura	218
13.2.5 Herramientas de desarrollo.....	218
13.3 Conclusión	219
13.4 Contenido de la página web de apoyo...	219
13.4.1 Mapa conceptual	219
13.4.2 Autoevaluaciones	219
13.4.3 Presentaciones en Power Point (exclusivo para el docente).....	219
13.5 Referencias	219

Capítulo 14 – Diseño de software

14.1 El proceso de diseño	222
14.1.1 Criterios de buen diseño	222
14.1.2 Patrones de diseño	224
14.2 Desarrollo del Caso Testigo.....	224
14.2.1 Presentación	224
14.2.2 Acceso al negocio	226
14.2.3 Negocio	230
14.2.4 Persistencia	230
14.2.5 Conexión con sistemas externos.....	233
14.2.6 Web Service.....	234
14.3 Conclusión	237
14.4 Contenido de la página web de apoyo...	237
14.4.1 Mapa conceptual	237
14.4.2 Autoevaluaciones	237
14.4.3 Presentaciones en Power Point (exclusivo para el docente).....	237
14.5 Referencias	237

Capítulo 15 – Métricas de software

15.1 El proceso de evaluación	240
15.1.1 Construyendo un sistema.....	240
15.1.2 Manteniendo un sistema	240
15.2 Primera aproximación al diseño de un sistema.....	240
15.3 Adentrándonos en los subsistemas.....	245
15.3.1 Métricas	245
15.3.2 Polimétricas.....	248
15.3.3 Desarmonías	250
15.4 Buenas prácticas de codificación.....	251
15.4.1 Estándares	252
15.4.2 Chequeos de estilo	252
15.5 Conclusión	257
15.6 Contenido de la página web de apoyo...	257
15.6.1 Mapa conceptual	257
15.6.2 Autoevaluaciones	257
15.6.3 Presentaciones en Power Point (exclusivo para el docente).....	257
15.7 Referencias	258

Parte VI – Codificación y pruebas

Capítulo 16 – Pruebas de software

16.1 Definiciones.....	260
16.1.1 Objetivos y filosofía.....	261
16.1.2 Principios	262
16.2 Niveles y Tipos de pruebas	264
16.2.1 Métodos de prueba.....	266
16.2.2 Caja Negra	266
16.2.3 Caja Blanca	267
16.3 Estrategias de prueba	270
16.3.1 Propiedades básicas	271
16.3.2 Evolución de las pruebas con el desarrollo.....	271
16.4 Tácticas.....	273
16.5 Claves del cambio en la forma de trabajo.....	275
16.5.1 Razones para automatizar las pruebas	276
16.5.2 Pruebas sistemáticas	278
16.6 Ejemplo.....	281
16.7 Conclusión	283
16.8 Contenido de la página web de apoyo...	284

20.5.3 Presentaciones en Power Point (exclusivo para el docente).....	359
20.6 Referencias.....	359
20.7 Apéndice - Visión del proyecto	359
20.7.1 Requerimientos del negocio	359
20.7.2 Visión de la solución.....	360
20.7.3 Características salientes	361
20.7.4 Alcance y limitaciones	361
20.7.5 Contexto del proyecto.....	362

Capítulo 21 – Gestión del proyecto del caso testigo - Seguimiento

21.1 Metodología conducida por los planes	364
21.1.1 Monitoreo y control del proyecto	364
21.1.2 Cierre del Proyecto.....	370
21.2 Metodología ágil.....	371
21.2.1 Monitoreo y control del proyecto	371
21.2.2 Cierre del Proyecto.....	374
21.3 Conclusión.....	375
21.4 Contenido de la página web de apoyo...	375
21.4.1 Mapa conceptual	375
21.4.2 Autoevaluaciones	375
21.4.3 Presentaciones en Power Point (exclusivo para el docente).....	375
21.5 Referencias.....	376

Parte VIII – Calidad de Software

Capítulo 22 – Calidad de procesos y productos de software

22.1 Calidad de procesos y productos de software	378
22.1.1 El concepto de calidad aplicado al software	378
22.1.2 Validación y verificación	379
22.1.3 Ejemplo de uso	380
22.2 Métricas en el software	385
22.2.1 Las medidas como mecanismo de aseguramiento y control de calidad	385
22.2.2 Medidas de la calidad de productos de software.....	387
22.2.3 Medidas de la calidad de procesos de software.....	389

22.3 Conclusiones.....	394
22.4 Contenido de la página web de apoyo	394
22.4.1 Mapa conceptual	394
22.4.2 Autoevaluaciones	394
22.4.3 Presentaciones en Power Point (exclusivo para el docente).....	394
22.5 Referencias.....	395

Capítulo 23 – Organización del área de calidad de una empresa

23.1 Introducción.....	398
23.2 Origen	398
23.2.1 Primeros pasos	398
23.3 Implementación del área de calidad.....	399
23.3.1 Alternativas de organigrama	401
23.3.2 Roles.....	403
23.3.3 Actividades.....	405
23.3.4 Recursos.....	409
23.3.5 Ejemplo de uso	410
23.4 Conclusiones.....	414
23.5 Contenido de la página web de apoyo...	415
23.5.1 Mapa conceptual	415
23.5.2 Autoevaluaciones	415
23.5.3 Presentaciones en Power Point (exclusivo para el docente).....	415
23.6 Referencias.....	415

Capítulo 24 – Planificación de las actividades de control de calidad de un proyecto de desarrollo

24.1 Introducción.....	418
24.2 Plan de calidad.....	420
24.2.1 Ejemplo del caso testigo	422
24.2.2 Objetivos	431
24.2.3 Flujo de trabajo	431
24.3 Conclusiones.....	432
24.4 Contenido de la página web de apoyo	432
24.4.1 Mapa conceptual	432
24.4.2 Autoevaluaciones	432
24.4.3 Presentaciones en Power Point (exclusivo para el docente).....	432
24.5 Referencias.....	432

24.6 Apéndice – Ejemplo del caso testigo	432
--	-----

Capítulo 25 – Estándares de calidad de software

25.1 Introducción.....	440
25.2 Normas y modelos de referencia	440
25.3 Norma ISO 9001-2000.....	444
25.4 Modelo CMMi.....	446
25.5 Enfoque y alcance de aplicación	448
25.6 Conclusiones.....	448
25.7 Contenido de la página web de apoyo...	450
25.7.1 Mapa conceptual	450
25.7.2 Autoevaluaciones	450
25.7.3 Presentaciones en Power Point (exclusivo para el docente).....	450

25.8 Referencias.....	450
-----------------------	-----

Parte IX – Cierre

Capítulo 26 – Conclusiones

26.1 Acerca de la tecnología	452
26.2 Acerca de la forma de trabajo.....	452
26.3 Acerca de la calidad	453
26.4 Acerca de la Ingeniería de Software.....	453
26.5 Contenido de la página web de apoyo...	454
26.5.1 Mapa conceptual	454
26.5.2 Autoevaluaciones	454
26.5.3 Presentaciones en Power Point (exclusivo para el docente).....	454
26.6 Referencias.....	454