

Índice

I	Fundamentos de la Ingeniería del Software	xix
1	Introducción a la Ingeniería del Software	5
1.1	¿Arte o ingeniería?	5
1.2	Objetivos	7
1.3	Introducción	7
1.4	¿Qué es la ingeniería?	8
1.5	Ingeniería y ciencias de la ingeniería	10
1.6	El software como artefacto tecnológico	12
1.6.1	¿Qué es el software?	13
1.6.2	La complejidad inherente al software	13
1.7	Sistematicidad, disciplina y cuantificación	14
1.8	La Ingeniería del Software como disciplina profesional	16
1.8.1	Breve historia de la Ingeniería del Software	16
1.8.2	Elementos de la Ingeniería del Software como disciplina profesional	17
1.9	Conceptos básicos de la Ingeniería del Software	19
1.9.1	Actividades y artefactos	19
1.9.2	Métodos, especificaciones y modelos	20
1.9.3	Procesos y ciclos de vida	21

2	Modelos y procesos	31
2.1	El proceso del proceso	31
2.2	Objetivos	32
2.3	Introducción	33
2.3.1	Una definición de proceso	34
2.3.2	Modelos del ciclo de vida, marcos de procesos y procesos	35
2.3.3	Características de las definiciones de procesos de software	37
2.3.4	Lenguajes para la especificación de procesos	38
2.4	Modelos de ciclo de vida del software	39
2.4.1	Modelo en cascada	40
2.4.2	Modelo en «V»	41
2.4.3	Modelos de proceso basados en prototipos	42
2.4.4	Modelo en espiral	44
2.5	Procesos de software	46
2.5.1	¿Qué se define en un proceso de software?	46
2.5.2	El modelo de referencia ISO 12207	48
2.5.3	Iteraciones e incrementos	52
2.6	Algunos tipos de procesos importantes	55
2.6.1	Procesos estructurados y procesos orientados a objetos	56
2.6.2	Procesos ágiles	56
2.6.3	Procesos basados en componentes	58
2.6.4	Especificaciones de proceso abiertas	60
2.7	Resumen	62
2.8	Notas bibliográficas	63
2.9	Cuestiones de autoevaluación	63
2.10	Actividades propuestas	65
3	Medición	67
3.1	La necesidad de medir	67
3.2	Objetivos	68
3.3	Introducción	68
3.3.1	Conceptos básicos	69
3.3.2	Tipos de escalas de medición	70
3.3.3	Clasificación de las medidas	71
3.3.4	Evaluación de las métricas	72
3.3.5	¿Qué medir en la Ingeniería del Software?	73
3.4	Medidas del producto: atributos internos	75
3.4.1	Medidas del tamaño de los sistemas	75
3.4.2	Medidas de la complejidad del software	76
3.4.3	Medidas de la documentación	80
3.4.4	Medidas de reutilización	81

3.4.5	Medidas de la eficiencia	81
3.5	Medidas del producto: atributos externos	82
3.6	Medidas del proceso y los recursos	84
3.6.1	Medidas relacionadas con el proceso	84
3.6.2	Medidas relacionadas con los recursos	84
3.7	Metodologías y estándares para la medición	87
3.7.1	Método Objetivo-Pregunta-Métrica (GQM)	87
3.7.2	El estándar IEEE 1061-1998	90
3.7.3	PSM y el estándar ISO/IEC 15939	91
3.7.4	Otras metodologías y estándares para la medición	92
3.8	Estudios empíricos	93
3.8.1	Encuestas	95
3.8.2	Casos de estudio	96
3.8.3	Experimentación formal	97
3.9	Resumen	99
3.10	Notas bibliográficas	100
3.11	Cuestiones de autoevaluación	101
3.12	Ejercicios y actividades propuestas	102
3.12.1	Ejercicios resueltos	102
3.12.2	Actividades propuestas	104

II Procesos fundamentales de la Ingeniería del Software

107

4	Requisitos	111
4.1	La difícil tarea de determinar qué debe hacerse	111
4.2	Objetivos	112
4.3	Introducción	113
4.4	Definiciones preliminares y características	115
4.4.1	El concepto de requisito	116
4.4.2	Actividades de requisitos	116
4.4.3	Actores	118
4.4.4	Características de los requisitos	119
4.4.5	El documento de especificación de requisitos	120
4.5	Tipos de requisitos	121
4.5.1	Requisitos funcionales	122
4.5.2	Requisitos no funcionales	123
4.5.3	Otras clasificaciones de los requisitos	125
4.6	Las actividades de requisitos	127
4.6.1	Obtención de requisitos	128
4.6.2	Análisis de requisitos	132
4.6.3	Especificación de requisitos	137

4.6.4	Validación de requisitos	143
4.7	Notaciones para el modelado conceptual	147
4.7.1	Casos de uso	147
4.7.2	Modelos entidad-relación	151
4.7.3	Diagramas de clases UML	152
4.7.4	Notaciones formales	155
4.8	Gestión del proceso de requisitos	156
4.8.1	Seguimiento	157
4.8.2	Métricas de los requisitos	158
4.8.3	Herramientas para la gestión de requisitos	160
4.9	Resumen	162
4.10	Notas bibliográficas	163
4.11	Cuestiones de autoevaluación	164
4.12	Ejercicios y actividades propuestas	166
4.12.1	Ejercicios resueltos	166
4.12.2	Actividades propuestas	170
5	Diseño	173
5.1	No es posible construir sin diseñar	173
5.2	Objetivos	174
5.3	Introducción	175
5.4	Conceptos fundamentales de diseño	176
5.4.1	Abstracción	176
5.4.2	Componentes e interfaces	177
5.4.3	Descomposición y modularización	177
5.4.4	Medición de la modularidad	178
5.4.5	Arquitectura de sistemas	180
5.4.6	Notaciones de diseño	184
5.5	Métodos de diseño	184
5.5.1	Métodos estructurados	184
5.5.2	Métodos orientados a datos	189
5.5.3	Diseño orientado a objetos	190
5.6	Otras técnicas relacionadas con el diseño	199
5.6.1	Los patrones de diseño software	199
5.6.2	Software <i>frameworks</i> , <i>plug-ins</i> y componentes	203
5.6.3	Diseño por contrato	205
5.7	Diseño de sistemas distribuidos	207
5.8	Evaluación y métricas en el diseño	209
5.9	Resumen	213
5.10	Notas bibliográficas	214
5.11	Cuestiones de autoevaluación	215

5.12	Ejercicios y actividades propuestas	216
5.12.1	Ejercicios resueltos	216
5.12.2	Actividades propuestas	219
6	Construcción	223
6.1	No da igual cómo esté construido	223
6.2	Objetivos	225
6.3	Introducción	225
6.4	Lenguajes de construcción	227
6.5	Reutilización del código	230
6.6	Principios fundamentales de la construcción de software	232
6.6.1	Minimizar la complejidad	232
6.6.2	Anticipar los cambios	250
6.6.3	Construir para verificar	253
6.6.4	Utilización de estándares	257
6.7	La calidad en la construcción de software	260
6.7.1	Aserciones y diseño por contrato	260
6.7.2	Análisis de rendimiento	262
6.7.3	Depuración	264
6.8	Gestión de la construcción	266
6.8.1	Planificación de la construcción	266
6.8.2	Métricas de construcción	267
6.9	Resumen	268
6.10	Notas bibliográficas	269
6.11	Cuestiones de autoevaluación	269
6.12	Ejercicios y actividades propuestas	271
6.12.1	Ejercicios resueltos	271
6.12.2	Actividades propuestas	276
7	Pruebas	279
7.1	El porqué de las pruebas	279
7.2	Objetivos	280
7.3	Introducción	280
7.3.1	Conceptos fundamentales	284
7.3.2	Limitaciones en la realización de pruebas	286
7.3.3	Las pruebas y el riesgo	287
7.4	Técnicas de prueba	289
7.4.1	Pruebas de caja blanca y de caja negra	290
7.4.2	Clasificación exhaustiva de las técnicas de prueba	295
7.5	Niveles de prueba	298
7.5.1	Pruebas según su objeto	298
7.5.2	Pruebas según el objetivo que persiguen	305

7.6	Pruebas unitarias con JUnit	308
7.6.1	Ejemplo sencillo de uso de JUnit	309
7.6.2	Complicación del ejemplo inicial	312
7.6.3	Colecciones de pruebas	314
7.6.4	JUnit 4	315
7.7	Métricas relacionadas con las pruebas	317
7.7.1	Medidas durante las pruebas	318
7.7.2	Evaluación de las pruebas realizadas	319
7.8	El proceso de prueba	320
7.9	Resumen	322
7.10	Notas bibliográficas	324
7.11	Cuestiones de autoevaluación	325
7.12	Ejercicios y actividades propuestas	326
7.12.1	Ejercicios resueltos	326
7.12.2	Actividades propuestas	330
8	Mantenimiento	333
8.1	La mente de los otros	333
8.2	Objetivos	334
8.3	Introducción	335
8.4	Conceptos fundamentales	336
8.4.1	¿Qué es el mantenimiento del software?	336
8.4.2	La facilidad de mantenimiento	337
8.4.3	Mantenimiento y calidad	339
8.4.4	Aspectos de la facilidad de mantenimiento	340
8.5	La práctica del mantenimiento del software	340
8.5.1	El mantenimiento del software como un caso especial de mantenimiento	342
8.5.2	La evolución del software y sus leyes	342
8.6	El proceso de mantenimiento	344
8.6.1	Las actividades de mantenimiento	345
8.6.2	El mantenimiento como preparación	348
8.7	Técnicas para el mantenimiento del software	348
8.7.1	Ingeniería inversa	350
8.7.2	Reingeniería	352
8.7.3	Reestructuración	354
8.8	Métricas de mantenimiento	357
8.8.1	Métricas del producto	357
8.8.2	Métricas relacionadas con el proceso	360
8.9	Resumen	361
8.10	Notas bibliográficas	363

8.11	Cuestiones de autoevaluación	364
8.12	Ejercicios y actividades propuestas	365
8.12.1	Ejercicios resueltos	365
8.12.2	Actividades propuestas	369

III Gestión y Calidad en la Ingeniería del Software 373

9 Calidad 377

9.1	La especial naturaleza de la calidad	377
9.2	Objetivos	378
9.3	Introducción	379
9.3.1	Cultura y ética de la calidad	379
9.3.2	Valor y costes de la calidad	381
9.3.3	Los multiples aspectos de la calidad	382
9.4	Calidad del producto	384
9.4.1	El modelo de calidad de McCall	384
9.4.2	El modelo de Boëhm	387
9.4.3	El modelo de calidad ISO/IEC 9126	388
9.4.4	Otros modelos de calidad	391
9.5	Calidad del proceso	392
9.5.1	Aseguramiento de la calidad	392
9.5.2	El modelo CMMI	394
9.5.3	Modelo SPICE: El estándar ISO/IEC 15504	399
9.5.4	Los estándares de la familia ISO 9000	401
9.5.5	Otros modelos, estándares y especificaciones	403
9.6	Resumen	410
9.7	Notas bibliográficas	412
9.8	Cuestiones de autoevaluación	413
9.9	Ejercicios y actividades propuestas	414
9.9.1	Ejercicios resueltos	414
9.9.2	Actividades propuestas	416

10 Gestión 419

10.1	El desarrollo de proyectos no es sólo tecnología	419
10.2	Objetivos	421
10.3	Visión general de la gestión de proyectos	422
10.4	La estimación de coste, plazos y esfuerzo	424
10.4.1	Estimación mediante juicio de expertos	425
10.4.2	Puntos de función	426
10.4.3	Modelos algorítmicos o paramétricos	429
10.4.4	Modelos basados en la inteligencia artificial	436

10.4.5	Sistemas dinámicos	439
10.4.6	Evaluación de modelos	440
10.4.7	Calibración de modelos	443
10.5	Planificación y seguimiento del proyecto	443
10.5.1	Estructura de descomposición del trabajo	444
10.5.2	Los métodos gráficos CPM y PERT	445
10.5.3	Diagramas de Gantt	450
10.5.4	Método del valor conseguido	451
10.6	Revisiones y cierre del proyecto	458
10.7	Gestión de los recursos humanos	459
10.8	Gestión y análisis del riesgo	460
10.9	Resumen	462
10.10	Notas bibliográficas	463
10.11	Cuestiones de autoevaluación	464
10.12	Ejercicios y actividades propuestas	465
10.12.1	Ejercicios resueltos	465
10.12.2	Actividades propuestas	468
11	Gestión de la configuración del software	471
11.1	La importancia de poner las cosas en su sitio	471
11.2	Objetivos	473
11.3	La configuración del software	473
11.4	Actividades de gestión de la configuración del software	477
11.4.1	Identificación de la configuración del software	478
11.4.2	Control de los cambios en el software	481
11.4.3	Gestión de entregas	484
11.5	Planificación y gestión	487
11.5.1	Contabilidad y medición en gestión de la configuración	488
11.5.2	Auditoría de la configuración software	489
11.6	Técnicas y herramientas para el control de versiones	489
11.6.1	Versiones, divisiones y deltas	490
11.6.2	Políticas de control de versiones en grupos de trabajo	491
11.7	Resumen	494
11.8	Notas biobliográficas	495
11.9	Cuestiones de autoevaluación	496
11.10	Ejercicios y actividades propuestas	497
11.10.1	Ejercicios resueltos	497
11.10.2	Actividades propuestas	502

12 Herramientas	505
12.1 Las herramientas nos diferencian	505
12.2 Objetivos	506
12.3 Introducción	507
12.3.1 Justificación de las herramientas CASE	508
12.3.2 Ventajas e inconvenientes del uso de herramientas CASE	509
12.4 Clasificación de las herramientas CASE	510
12.4.1 Herramientas CASE según el ciclo de vida	511
12.4.2 Herramientas CASE según su nivel de integración	519
12.5 Selección y evaluación de herramientas	526
12.5.1 Identificación de las necesidades	527
12.5.2 Selección de herramientas candidatas	528
12.5.3 Evaluación técnica	529
12.5.4 Toma de la decisión final	530
12.6 Resumen	531
12.7 Notas bibliográficas	531
12.8 Cuestiones de autoevaluación	532
12.9 Ejercicios y actividades propuestas	533
12.9.1 Ejercicios resueltos	533
12.9.2 Actividades propuestas	536