

차례

CHAPTER 01 소개

1.1 소프트웨어	17
1.1.2 소프트웨어의 종류	18
1.1.3 시스템	20
1.2 소프트웨어 개발 작업	21
1.2.1 기본 활동	21
1.2.2 개발 작업의 특징	22
1.3 소프트웨어 공학의 접근법	25
1.3.1 소프트웨어 공학의 정의	26
1.3.2 소프트웨어 공학의 목표	26
1.4 소프트웨어 공학의 주제	28
1.4.1 단계적 프로세스	28
1.4.2 품질 보증	30
1.4.3 프로젝트 관리	32
1.5 연관 분야	34
1.5.1 연관 지식 영역	35
1.5.2 소프트웨어 공학과 컴퓨터 과학	36
■ 연습문제	37

CHAPTER 02 프로세스와 방법론

2.1 소프트웨어 생명주기	46
2.2 프로세스	47
2.2.1 프로세스 종류	48
2.2.2 프로세스 정의	50
2.2.3 좋은 프로세스의 특성	51
2.3 프로세스 모델	55
2.3.1 폭포수 모델	56
2.3.2 프로토타이핑 모델	58
2.3.3 나선형 모델	60
2.3.4 진화적 모델	61
2.3.5 Unified Process	63
2.3.6 애자일 프로세스	65
2.4 지원 프로세스	69

2.4.1 관리 프로세스	70
2.4.3 품질보증 프로세스	71
2.4.4 형상 관리 프로세스	73
2.5 방법론	74
2.5.1 구조적 방법론	75
2.5.2 정보공학 방법론	76
2.5.3 객체지향 방법론	77
■연습문제	80

CHAPTER 03 프로젝트 계획과 관리

3.1 프로젝트 시작	90
3.1.1 프로젝트 가치	91
3.1.2 프로젝트 리스크	91
3.1.3 타당성 분석	92
3.2 프로젝트 계획과 스케줄링	92
3.2.1 목표 설정	93
3.2.2 프로젝트 범위	94
3.2.3 WBS	95
3.2.4 스케줄링	96
3.3 비용 예측 기법	101
3.3.1 COCOMO-81	103
3.3.2 COCOMO II	106
3.3.3 기능 점수	109
3.4 프로젝트 팀 조직	115
3.4.1 팀 역할	116
3.4.2 직능별 조직	116
3.4.4 프로젝트별 조직	117
3.4.5 매트릭스 조직	118
3.4.6 애자일 조직	120
3.5 실행과 모니터링	121
3.5.1 프로젝트 실행	121
3.5.2 프로젝트 모니터링	121
3.5.3 번다운 차트	123
3.6 리스크 관리	124
3.6.1 리스크 파악	124
3.6.2 리스크 평가	125
3.6.3 리스크 관리	126
■연습문제	128

CHAPTER 04 요구 분석

4.1 요구	138
4.1.1 기능 요구	171
4.1.2 비기능 요구	143
4.1.3 요구 대상에 의한 분류	144
4.2 요구 추출	145
4.2.1 요구 정보 출처	146
4.2.2 고객의 발표	148
4.2.3 문헌 양식 조사	149
4.2.4 인터뷰	150
4.2.5 설문	152
4.2.6 브레인스토밍	153
4.2.7 프로토타이핑	154
4.3 요구 분석	156
4.3.1 요구 품질	156
4.3.2 도메인 분석	157
4.3.3 시나리오 기반 분석	159
4.4 유스케이스	160
4.4.1 유스케이스 다이어그램	161
4.4.2 유스케이스 명세	164
4.4.3 유스케이스 사이의 관계	166
4.5 요구 명세	169
4.5.1 작성 방법	170
4.6 요구 검증	171
■ 연습문제	173

CHAPTER 05 요구 모델링

5.1 모델링	182
5.1.1 관점과 추상화 수준	183
5.1.2 소프트웨어와 모델링	184
5.1.3 모델 사이의 관계	185
5.2 UML	186
5.2.1 UML 역사	186
5.2.2 UML 다이어그램	188
5.2.3 모델링 과정	191
5.3 정적 모델링	192
5.3.1 객체지향 개념	193
5.3.2 클래스 다이어그램	199
5.4 동적 모델링	207
5.4.1 시퀀스 다이어그램	208

5.4.2 협동 다이어그램	213
5.4.3 상태 다이어그램	214
5.5 제어 모델링	216
5.6 모델 검증	218
■ 연습문제	222

CHAPTER 06 설계 원리

6.1 설계 기본 개념	234
6.1.1 서브시스템, 모듈	235
6.1.2 설계 관점	236
6.1.3 설계 작업 과정	237
6.2 품질 목표	239
6.3 전통적인 설계 원리	241
6.3.1 추상화	242
6.3.2 캡슐화	243
6.3.3 모듈화	244
6.3.4 결합	246
6.3.5 응집	248
6.4 객체지향 설계 원리	251
6.4.1 인터페이스와 구현의 분리	252
6.4.2 단일 책임의 원리	253
6.4.3 개방 폐쇄의 원리	253
6.4.4 리스코프 교체의 원리	254
6.4.5 인터페이스 분리의 원리	255
6.4.6 의존 관계 역전의 원리	256
6.5 설계 메트릭	257
6.5.1 전통적인 메트릭	257
6.5.2 객체지향 메트릭	257
■ 연습문제	259

CHAPTER 07 아키텍처 설계와 패턴

7.1 아키텍처 기초	270
7.1.1 아키텍처란?	270
7.1.2 아키텍처의 역할	270
7.1.3 아키텍처의 표현	271
7.2 아키텍처 스타일	274
7.2.1 클라이언트 서버형	276
7.2.2 계층형	277
7.2.3 이벤트 기반 아키텍처	279

7.2.4 MVC	280
7.2.5 파이프 필터	281
7.2.6 데이터 중심 아키텍처	282
7.2.7 Peer-to-Peer 스타일	283
7.3 디자인 패턴	285
7.3.1 디자인 패턴의 혜택	286
7.3.2 디자인 패턴의 형식	287
7.3.3 싱글톤 패턴	288
7.3.4 반복자 패턴	290
7.3.5 어댑터 패턴	292
7.3.6 데코레이터 패턴	294
7.3.7 팩토리 메소드 패턴	299
7.3.8 추상 팩토리 패턴	301
7.3.9 상태 패턴	303
7.3.10 옵서버 패턴	306
7.5 아키텍처 평가	309
■ 연습문제	313

CHAPTER 08 UI 설계

8.1 UI 기본 개념	324
8.1.1 사용성	324
8.1.2 멘탈 모델	325
8.1.3 피드백	326
8.1.4 제약	327
8.2 UI 설계 원리	328
8.3 UI 설계 과정	333
8.3.1 사용자 분석	334
8.3.2 태스크 분석	335
8.3.3 UI 설계와 구현	335
8.3.4 사용성 테스트	337
8.4 UI 요소	339
8.5 화면 및 출력 설계	342
8.5.1 화면 설계	342
8.5.2 출력 설계	343
■ 연습문제	346

CHAPTER 09 코딩

9.1 코딩 작업	353
9.1.1 작업 과정	354

9.1.2 자주 발생하는 오류	355
9.2 코딩 표준	360
9.2.1 명명 규칙	360
9.2.2 형식	363
9.2.3 문장과 수식	366
9.2.4 오류처리	367
9.2.5 주석	369
9.3 설계에서 코드 생성	372
9.3.1 연관의 코딩	372
9.3.2 시퀀스 다이어그램의 코딩	374
9.4 리팩토링	375
9.4.1 리팩토링 개념	375
9.4.2 리팩토링 과정	377
9.4.3 코드 스멜	378
9.4.4 리팩토링 사례	380
9.5 코드 품질 향상 기법	383
9.5.1 인스펙션	384
9.5.2 정적 분석	386
9.5.3 테스트 중심 개발	388
■ 연습문제	392

CHAPTER 10 테스트

10.1 테스트 기초	404
10.1.1 버그, 오류, 결함, 고장	405
10.1.2 테스트 원리	406
10.1.3 테스트 작업 과정	407
10.1.4 테스트 케이스	409
10.2 블랙박스 테스트	410
10.2.1 동등 분할 기법	411
10.2.2 경계값 분석	413
10.2.3 원인 결과 그래프	414
10.3 화이트박스 테스트	416
10.3.1 논리 흐름의 표현	417
10.3.2 검증 기준	418
10.4 상태기반 테스트	422
10.5 통합 테스트	425
10.5.1 빅뱅 통합	426
10.5.2 하향식 통합	427
10.5.3 상향식 통합	429
10.5.4 연쇄식 통합	430

10.6 시스템 및 인수 테스트	432
10.6.1 기능 테스트	433
10.6.2 성능 테스트	435
10.6.3 보안 테스트	436
10.6.4 UI 테스트	437
10.6.5 인수 테스트	438
■ 연습문제	440

CHAPTER 11 유지보수

11.1 유지보수 소개	452
11.1.1 변경의 이유와 유지보수 유형	452
11.1.2 Lehman의 법칙	454
11.2 유지보수 작업 과정	456
11.2.1 유지보수 작업	456
11.2.2 유지보수 프로세스	458
11.2.3 프로그램의 이해	462
11.2.4 변경 파악과 분석	463
11.3 형상 관리	465
11.3.1 베이스라인	466
11.3.2 형상관리 절차	468
11.4 역공학	472
11.4.1 역공학 작업 순서	473
11.4.2 역공학의 용도	474
11.4.3 재문서화	474
11.4.4 설계 복구	475
11.5 리엔지니어링	475
11.5.1 리엔지니어링 목적	476
11.5.2 리엔지니어링 과정	477
■ 연습문제	478

CHAPTER 12 품질

12.1 소개	488
12.1.1 품질 개념	490
12.1.2 소프트웨어 품질	493
12.2 품질 모델	494
12.3 품질관리	498
12.3.1 품질 보증 조직	500
12.3.2 프로세스와 표준을 정의	501
12.3.3 품질 보증 활동	504

12.3.4 인스펙션	507
12.4 품질 측정	510
12.4.1 품질 측정의 유용성	511
12.4.1 품질 메트릭	512
12.5 프로세스 개선	518
12.5.1 CMMi	519
11.5.2 ISO 9001	525
■ 연습문제	529