

I. 개요

1. 직 종 명 : 응용SW엔지니어링

2. 직종 정의 : 응용소프트웨어 엔지니어링은 컴퓨터 프로그래밍 언어로 각 업무에 맞는 소프트웨어의 기능에 관한 설계, 구현 및 테스트를 수행하고, 사용자에게 배포하며, 버전관리를 통해 제품의 성능을 향상시키고, 서비스를 개선하는 업무에 종사.

3. 훈련이수체계(수준별 이수 과정/과목)

8	임원	SW아키텍처 비전 수립			
7	부장/차장	SW아키텍처 수평 관리 SW아키텍처 요구사항 탐색화 후보 SW아키텍처의 경쟁화 SW아키텍처 설계 SW아키텍처 논사화	제품리케이스선 요구사항 분석		개념데이터 모델링
6	과장/대리	SW아키텍처 요구사항전달 SW아키텍처 테스트 SW아키텍처 변경 관리	제품리케이스선 리퍼모델링 인티 레이스 설계 기능 모델링 제품리케이스선 설계 장기모델 설계 동적모델 설계 소프트웨어공학 활용 소프트웨어개발 모델론 활용	운영체계 이식 데이터베이스 도라이버 분석 데이터베이스 도라이버 설계 데이터베이스 도라이버 구현 임베디드 SW 요구사항 분석	데이터베이스 요구사항 분석 논리 데이터베이스 설계 물리 데이터베이스 설계 데이터 품질관리 데이터베이스 성능 확보 데이터리플론 설계
5	사원	SW 개발 지원	요구사항 확인 데이터 입출력 구현 플래어 구현 정보시스템 이해 제품소프트웨어 패키징 서버프로그램 구현 인티 레이스 구현 화면 설계 제품리케이스선 테스트 관리	임베디드 시스템 테스트 플래어 분석 플래어 설계 플래어 구현 운영체계 커널 분석 데이터베이스 도라이버 구현 환경 구축 임베디드 제품리케이스선 분석 임베디드 제품리케이스선 설계 임베디드 제품리케이스선 구현	데이터 표준화 데이터전환 SQL응용
4	사원	SW아키텍처 이해		하드웨어 분석 기술문서 개발 플래어 구현 환경 구축 임베디드 제품리케이스선 구현 환경 구축	데이터베이스 구현
3	사원		제품리케이스선 배포 프로그램밍 언어 활용 응용 SW 기초 기술 활용 화면 구현 제품리케이스선 테스트 수행	오른 플랫폼 활용	SQL활용
-		직업기초능력			
수준	직종	SW아키텍처	응용SW엔지니어링	임베디드SW엔지니어링	DB엔지니어링

7	부장/차장	<div>네트워크 환경 분석</div> <div>네트워크 도중도서 설계</div> <div>네트워크 자원관리 설계</div> <div>네트워크 QoS 제어 설계</div> <div>네트워크 품질 평가</div> <div>네트워크 프로세스 관리</div> <div>네트워크 프로토콜 분석</div> <div>네트워크 프로토콜 설계</div>		<div>UI/UX 계획 수립</div>	<div>시스템SW 요구사항 분석</div> <div>시스템SW 아키텍처 설계</div>
6	과장/대리	<div>네트워크 소프트웨어 아키텍처 수립</div>	<div>관리적 보안 구축</div> <div>보안인증 관리</div>	<div>UI/UX 환경 분석</div> <div>사용자 리서치</div>	<div>시스템SW 상세 설계</div> <div>시스템SW 테스트</div> <div>시스템SW 기술문서 작성</div>
5	사원	<div>네트워크 소프트웨어 개발 방법 수립</div>	<div>보안 구축 계획 수립</div> <div>보안 구축 요구사항 분석</div> <div>소프트웨어 개발 보안 구축</div> <div>데이터베이스 보안 구축</div> <div>시스템 보안 구축</div> <div>네트워크 보안 구축</div>	<div>UI/UX 요구 분석</div> <div>UI/UX 콘셉트 기획</div> <div>UI 아키텍처 설계</div>	<div>시스템SW 통합 구현</div> <div>시스템SW 배포</div> <div>시스템SW 단위 모듈구현</div> <div>시스템SW 인터페이스 구현</div>
4	사원			<div>UI 디자인</div> <div>UI/UX 가이드 제작</div>	
3	사원	<div>네트워크 프로그램링 구현</div>	<div>물리적 보안 구축</div>	<div>UI 구현</div>	
2				<div>UI 테스트</div>	
-		직업기초능력			
수준	직종	NW엔지니어링	보안엔지니어링	UI/UX엔지니어링	시스템SW엔지니어링

<div>5과정/대리</div>		<div>핀테크 요구사항 분석</div>			
<div>5사원</div>		<div>핀테크 인프라 설계</div> <div>핀테크 인터페이스 설계</div> <div>핀테크 기능 설계</div> <div>핀테크 보안·인증 설계</div>			
<div>4사원</div>		<div>핀테크 인터페이스 구현</div> <div>핀테크 기능 구현</div> <div>핀테크 보안·인증 구현</div>			
<div>3사원</div>		<div>핀테크 테스트</div>			
-		직업기초능력			
수준	직종				
		빅데이터플랫폼구축	핀테크엔지니어링		

※ 해당직종(음영)의 훈련과정을 편성하는 경우 훈련과정별 목표에 부합한 수준으로 해당 직종에서 제시한 능력단위를 기준으로 과정/과목을 편성하고, 이외 직종의 능력단위를 훈련과정에 추가 편성하려는 경우 유사 직종의 동일 수준의 능력단위를 추가할 수 있음

4. 훈련시설

시설명 \ 훈련인원	기준인원	면 적	기준인원 초과 시 면적 적용	시설활용구분 (공용/전용)
강의실	20	45㎡	1명당 1.5㎡씩 추가	공용
컴퓨터실	20	45㎡	1명당 1.5㎡씩 추가	공용
실습실			해당없음	공용
공구·재료실			해당없음	공용

※ 훈련시설은 훈련과정/과목에 필요한 시설을 구축

5. 교사

○ 「근로자직업능력 개발법」 제33조와 관련 규정에 따름

II. 훈련과정

○ 과정/과목명 : 직업기초능력

- 훈련개요

훈련목표	직업인으로서 갖추어야할 기본적인 소양을 함양
수 준	-
훈련시간	훈련과정 전체 교육시간의 10% 이내에서 자율편성
훈련가능시설	강의실 또는 컴퓨터실
권장훈련방법	집체훈련 또는 원격훈련

- 편성내용

단 원 명	학 습 내 용
의사소통능력	문서이해능력, 문서작성능력, 경청능력, 의사표현능력, 기초외국어능력
수리능력	기초연산능력, 기초통계능력, 도표분석 능력, 도표작성능력
문제해결능력	사고력, 문제처리능력
자기개발능력	자아인식능력, 자기관리능력, 경력개발 능력
자원관리능력	시간자원관리능력, 예산자원관리능력, 물적자원관리능력, 인적자원관리능력
대인관계능력	팀웍능력, 리더십능력, 갈등관리능력, 협상능력, 고객서비스능력
정보능력	컴퓨터활용능력, 정보처리능력
기술능력	기술이해능력, 기술선택능력, 기술적용능력
조직이해능력	국제감각, 조직체제이해능력, 경영이해능력, 업무이해능력
직업윤리	근로윤리, 공동체 윤리

○ 과정/과목명 : 2001020201_16v3 요구사항 확인

- 훈련개요

훈련목표	요구사항 확인이란 업무 분석가가 수집 분석 정의한 요구사항과 이에 따른 분석모델에 대해서 확인과 현행 시스템에 대해 분석하는 능력을 함양.
수 준	5수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
현행 시스템 분석하기	<ol style="list-style-type: none"> 1.1 개발하고자 하는 응용소프트웨어에 대한 이해를 높이기 위해, 현행 시스템의 적용현황을 파악함으로써 개발범위와 향후 개발될 시스템으로의 이행방향성을 분석할 수 있다. 1.2 개발하고자 하는 응용소프트웨어와 관련된 운영체제, 데이터베이스관리시스템, 미들웨어 등의 요구사항을 식별할 수 있다. 1.3 현행 시스템을 분석하여, 개발하고자 하는 응용소프트웨어가 이후 적용될 목표시스템을 명확하고 구체적으로 기술할 수 있다.
요구사항 확인하기	<ol style="list-style-type: none"> 2.1 소프트웨어 공학기술의 요구사항 분석 기법을 활용하여 업무 분석가가 정의한 응용소프트웨어의 요구사항을 확인할 수 있다. 2.2 업무 분석가가 분석한 요구사항에 대해 정의된 검증기준과 절차에 따라서 요구사항을 확인할 수 있다. 2.3 업무 분석가가 수집하고 분석한 요구사항이 개발하고자 하는 응용소프트웨어에 미칠 영향에 대해서 검토하고 확인할 수 있다.
분석모델 확인하기	<ol style="list-style-type: none"> 3.1 소프트웨어 공학기술의 요구사항 도출 기법을 활용하여 업무 분석가가 제시한 분석모델에 대해서 확인할 수 있다. 3.2 업무 분석가가 제시한 분석모델이 개발할 응용소프트웨어에 미칠 영향을 검토하여 기술적인 타당성 조사를 할 수 있다. 3.3 업무 분석가가 제시한 분석모델에 대해서 응용소프트웨어를 개발하기 위해 필요한 추가적인 의견을 제시할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 산업 분야에 대한 분석 • 요구사항 명세화의 정확성 여부 • 서버 하드웨어 용량 산정 방법 숙지 및 수행 내역 • 운영체제의 주요 기능, 구조, 특성에 대한 분석 • 미들웨어(DBMS/WAS)의 주요 기능, 구조, 특성에 대한 분석
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> 가상화 관련 지식 모델링 기법 소프트웨어 개발 방법론 업무 특성에 대한 이해 요구공학 방법론 요구분석기법 용어사전 작성 방법 클라우드 컴퓨팅 관련 지식 타당성 분석기법 통계학 프로젝트 환경 및 특수성 플랫폼에 따른 기능 및 성능 특성 해당 산업 분야에 대한 지식 해당 플랫폼에 대한 지식
기술	<ul style="list-style-type: none"> DBMS 구성 및 관리 기술 UML 작성 기술 가상화 관련 기술 내부 및 외부 환경 분석 기술 네트워크 구성 및 관리 능력 리뷰 진행 기술 분석 자동화도구 도구 사용 능력 분석 자동화도구 도구 활용 기술 요구사항 관리 도구 사용 기술 운영체제 구성 및 관리 능력 유즈케이스 작성 능력 저장장치 구성 및 관리 능력
태도	<ul style="list-style-type: none"> 검증할 항목 분석을 위한 치밀한 태도 고객의 요청을 수용하고자하는 태도 기술 관련 각종 정보 수집에 대한 적극성 분류 및 정리 태도 비즈니스의 내용을 정확히 반영하는 프로세스 모델을 만들려는 치밀한 자세 시스템과 개발 소프트웨어와의 관계를 파악하려는 태도 시스템에 대한 정확성과 이해의 완전성을 갖고자 하는 태도 요구사항에 대한 일관성을 부여하는 자세 요구사항을 유기적 관계를 발견하는 자세 요구사항을 정확하게 기술하려는 자세 요구사항의 명확성을 찾으려는 자세 요구사항의 정확성과 완전성을 확보하려는 자세 용어정의 정확성과 완전성을 기하고자 하는 의지 정확한 유즈케이스를 이해하고 분석하려는 자세 책임감 및 검증에 대한 완벽함을 추구하는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성 도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020205_16v4 데이터 입출력 구현

- 훈련개요

훈련목표	데이터 입출력 구현이란 응용소프트웨어가 다루어야 하는 데이터 및 이들 간의 연관성, 제약조건을 식별하여 논리적으로 조직화 하고, 소프트웨어 아키텍처에 기술된 데이터저장소에 조직화된 단위의 데이터가 저장될 최적화된 물리적 공간을 구성하고 데이터 조작언어를 이용하여 구현하는 능력을 함양.
수 준	5수준
훈련시간	시간
훈련가능시설	
권장훈련방법	

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
논리 데이터저장소 확인하기	1.1 업무 분석가, 데이터베이스 엔지니어가 작성한 논리 데이터저장소 설계 내역에서 정의된 데이터의 유형을 확인하고 식별할 수 있다. 1.2 논리 데이터저장소 설계 내역에서 데이터의 논리적 단위와 데이터 간의 관계를 확인할 수 있다. 1.3 논리 데이터저장소 설계 내역에서 데이터 또는 데이터간의 제약조건과 이들 간의 관계를 식별할 수 있다.
물리 데이터저장소 설계하기	2.1 논리 데이터저장소 설계를 바탕으로 응용소프트웨어가 사용하는 데이터저장소의 특성을 반영한 물리 데이터저장소 설계를 수행할 수 있다. 2.2 논리 데이터저장소 설계를 바탕으로 목표 시스템의 데이터 특성을 반영하여 최적화된 물리 데이터저장소를 설계할 수 있다. 2.3 물리 데이터저장소 설계에 따라 데이터저장소에 실제 데이터가 저장될 물리적 공간을 구성할 수 있다.
데이터 조작 프로시저 작성하기	3.1 응용소프트웨어 설계와 물리 데이터저장소 설계에 따라 데이터 저장소에 연결을 수행하는 프로시저를 작성할 수 있다. 3.2 응용소프트웨어 설계와 물리 데이터저장소 설계에 따라 데이터 저장소로부터 데이터를 읽어 오는 프로시저를 작성할 수 있다. 3.3 응용소프트웨어 설계와 물리 데이터저장소 설계에 따라 데이터 변경 내용 또는 신규 입력된 데이터를 데이터 저장소에 저장하는 프로시저를 작성할 수 있다. 3.4 구현된 데이터 조작 프로시저를 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화 할 수 있다.
데이터 조작 프로시저 최적화하기	4.1 프로그래밍 언어와 도구에 대한 이해를 바탕으로 응용소프트웨어 설계, 물리 데이터저장소 설계와 운영 환경을 고려하여 데이터 조작 프로시저의 성능을 예측할 수 있다. 4.2 업무 분석가에 의해 정의된 요구사항을 기준으로, 성능측정 도구를 활용하여 데이터 조작 프로시저의 성능을 측정할 수 있다. 4.3 실 데이터를 기반으로 테스트를 수행하여 데이터 조작 프로시저의 성능에 영향을 주는 병목을 파악할 수 있다. 4.4 테스트 결과와 정의된 요구사항을 기준으로 데이터조작 프로시저의 성능에 따른 이슈 발생 시 이에 대해 해결할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 업무영역에서 발생하는 데이터와 그 용처에 대한 이해 • 분석 산출물 및 다이어그램 분석 • 데이터 저장소 관련 설계 산출물 및 다이어그램 이해 • 개발 애플리케이션의 주요 매커니즘 및 적용 개발언어 분석 능력 • 데이터 조작 전용 프로그래밍 언어의 이해
----------	---

평가시 고려사항	<ul style="list-style-type: none"> • 이름규칙, 주석처리 등의 이해 • 형상관리 도구 방법과 소프트웨어 아키텍처와 일관성 • 사용하는 물리데이터저장소의 성능과 특성에 대해 이해
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • E-R Modeling • SQL(Structure Query Language) • 구현에 요구되는 아키텍처의 이해 • 데이터베이스에 대한 이해 • 설계 산출물의 이해 • 성능 평가 기준에 대한 이해 • 소프트웨어 테스트에 대한 이해 • 수학, 통계 지식 • 응용소프트웨어가 사용하는 데이터저장소에 따른 특성(file, RDBMS, XML Database 등) • 자료구조 • 프로그래밍언어(C, C++, Java-JSP/Servlet 등)와 도구(IDE 등)의 활용방법에 대한 이해 • 프로그래밍언어(C, C++, Java-JSP/Servlet 등)와 도구(IDE 등)의 활용방법의 이해
기술	<ul style="list-style-type: none"> • DBMS 관리 도구 활용 능력 • DBMS 사용 기술 • E-R 모델링 도구 활용 능력 • E-R 모델링 도구 활용능력 • ORM 프레임워크 활용 능력 • RDBMS를 통한 구현 능력 • UML 모델링 도구 활용 능력 • 개발에 필요한 프로그래밍 언어 및 도구 활용 능력 • 개발에 필요한 프로그래밍 언어 표준과 코드 검토 기법 • 단위테스트 도구 활용기법 • 모델링 검증 도구 활용 능력 • 성능 측정 도구 활용 능력 • 소스코드 인스펙션 능력 • 소스코드 표준 문서화 능력 • 컴파일러 사용법 • 쿼리(Query) 성능 측정 도구 활용 능력 • 트랜잭션 인터페이스 설계 능력 • 표준 단어, 용어, 도메인, 코드에 대한 정의 능력 • 프로그램 디버깅 기법 • 프로그램 코드 검토 기법
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준을 준수하려는 태도 • 객관적이고 논리적인 평가를 수행할 수 있는 태도 • 기술 및 제품에 대한 정보수집과 학습에 대한 적극성 • 데이터베이스 기술 동향에 대한 이해와 관련 제품들에 대해 파악하기 위한 노력 • 산출물 완성도를 위한 적극적인 태도 • 자신이 수행한 작업에 대한 평가의 객관성 • 정확성과 완전성을 기하고자 하는 의지 • 제품의 품질 및 고객 우선주의 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
.

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020206_16v4 통합 구현

- 훈련개요

훈련목표	통합 구현이란 모듈간의 분산이 이루어진 경우를 포함하여 단위 모듈간의 데이터 관계를 분석하여 이를 기반으로 한 메커니즘을 통해 모듈간의 효율적인 연계를 구현하고 검증하는 능력을 함양.
수 준	5수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
연계 데이터 구성하기	1.1 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 데이터 연계 요구사항을 분석할 수 있다. 1.2 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계가 필요한 데이터를 식별할 수 있다. 1.3 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계를 위한 데이터 표준을 설계할 수 있다.
연계 매카니즘 구성하기	2.1 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 특성을 고려하여 효율적 데이터 송수신 방법을 정의할 수 있다. 2.2 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 데이터 연계 요구사항을 고려하여 연계주기를 정의할 수 있다. 2.3 개발하고자 하는 응용소프트웨어와 연계 대상 내외부 모듈 간의 연계 목적을 고려하여 데이터 연계 실패 시 처리방안을 정의할 수 있다. 2.4 응용소프트웨어와 관련된 내외부 모듈 간의 연계 데이터의 중요성을 고려하여 송수신 시 보안을 적용할 수 있다.
내외부 연계 모듈 구현하기	3.1 구성된 연계 메카니즘에 대한 명세서를 참조하여 연계모듈구현을 위한 논리적, 물리적 환경을 준비할 수 있다. 3.2 구성된 연계 메카니즘에 대한 명세서를 참조하여 외부 시스템과의 연계 모듈을 구현할 수 있다. 3.3 연계모듈의 안정적인 작동여부와 모듈 간 인터페이스를 통해 연동된 데이터의 무결성을 검증할 수 있다. 3.4 구현된 연계모듈을 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화 할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 연계 매카니즘에 대한 이해도 • 연계 표준/프로토콜에 대한 이해도 • 미들웨어, 인터페이스 소프트웨어에 대한 이해도 • 모니터링 도구 사용법 및 모니터링 산출물에 대한 이해도
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발 방법론 이해 • 구현에 요구되는 아키텍처의 이해 • 네트워크 이해 • 데이터베이스 이해 • 분산 프로그래밍에 대한 이해 • 설계 모델링 기법 • 설계 산출물의 이해 • 소프트웨어 아키텍처에 대한 이해 • 소프트웨어 테스트에 대한 이해 • 업무 특성에 대한 이해 • 프로그래밍어 활용방법의 이해 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • IDE 및 개발환경 도구 활용 • UML 작성 기술 • 기술영역별 미들웨어/솔루션 활용 • 단위테스트 도구 활용기법 • 데이터 보안 기술 • 데이터 연계/이관 도구 활용 능력 • 설계 모델링 기술 • 프레임워크(애플리케이션, 웹, 데이터 액세스, 테스트 자동화, 기타) 활용 • 프로그램 디버깅 기법 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준을 준수하려는 태도 • 개발 팀원 간의 원활한 협업을 추구하는 태도 • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도를 위한 적극적인 태도 • 성공적인 개발을 위한 의지 • 정확성과 완전성을 기하고자 하는 의지 • 주어진 과제를 완수하는 책임감

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성 도구	세트	공용	1
• 빔 프로젝트	대	공용	30
• SW개발 도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020208_16v3 정보시스템 이행

- 훈련개요

훈련목표	정보시스템 이행이란 개발자 환경에서 개발한 결과물을 운영 환경에 설치하고, 사용자 요구사항과 최종적으로 일치하는 지에 대해 승인을 얻어 응용소프트웨어 결과물을 사용자에게 전달하여 인계하고 시스템을 운영할 수 있도록 교육하고 지원하는 능력을 함양.
수 준	5수준
훈련시간	48시간
훈련가능시설	컴퓨터실
권장훈련방법	집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
정보시스템 사용자 교육하기	1.1 개발된 응용소프트웨어 사용자 교육을 위해 교육 항목과 대상자, 강사선정, 방법, 시간, 횟수 등을 포함한 교육계획을 수립할 수 있다. 1.2 개발된 응용소프트웨어의 기능, 사용법과 필요 기술에 관한 사항을 포함한 사용자 교육 교재를 개발할 수 있다. 1.3 효과적인 사용자 교육이 이루어질 수 있도록 사전에 필요한 교육환경을 구축하고 점검 할 수 있다. 1.4 교육계획에 따라 사용자 교육을 실시하고, 평가기준에 따라 교육결과 및 성과를 평가할 수 있다.
정보시스템 이행하기	2.1 하드웨어, 시스템 소프트웨어, 네트워크 등 시스템 운영환경을 설치하고, 개발된 애플리케이션을 운영환경에 설치할 수 있다. 2.2 시스템 운영방법 및 사용법과 필요 기술에 관한 사항을 포함한 운영자 매뉴얼을 개발할 수 있다. 2.3 정보시스템 이행을 위한 각 영역 별 사전점검표를 작성하고, 데이터 이행작업 및 검증할 수 있다. 2.4 정보시스템 이행 시 발생할 수 있는 실패나 사고에 대비하여 신속하게 지원할 수 있는 비상 대책을 수립할 수 있다.
정보시스템 안정화하기	3.1 정보시스템 이행 후 개발된 응용소프트웨어의 데이터, 오류, 사용자 추가 요구사항을 수집하고 검토할 수 있다. 3.2 정보시스템 이행 후 실행되는 과정 중에 발생할 수 있는 정보시스템의 자원 사용량, 운영 상황을 분석하여 정보시스템을 안정화할 수 있다. 3.3 개발된 응용소프트웨어가 적용되는 정보시스템의 성능을 분석하여 문제점을 파악한 뒤 성능을 개선할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> 교육일정 계획의 실현가능성과, 교육시행 후 교육성과 작성된 매뉴얼(사용자, 개발자, 운영자 등)의 구체성과 이해 용이성 시스템 운영환경 설정과 애플리케이션의 배치 상황 이해 시간대별 영역별 사전점검표의 구체성 데이터 이행작업 점검 및 검증 이행여부 결정을 위한 과정 및 결정근거 이행 후 오류 및 조치 내역과 사용자 추가요구사항 수집 및 반영내역 시스템 안정화 및 성능개선을 위한 조치 내역
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 국제 표준 제품 품질특성에 대한 사항 • 문서작성 도구 활용 방법 • 소프트웨어 아키텍처에 대한 이해 • 시스템 성능 분석 및 진단에 대한 지식 • 시스템 운영 환경에 대한 이해 • 시스템 운영에 대한 데이터(오류, 자원사용량 등) 수집 및 분석에 대한 이해 • 업무 특성에 대한 이해 • 표준 양식(Template) 작성 및 활용법 • 피교육 집단 특성에 대한 이해 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • 결함관리 도구 활용 능력 • 교육성과 분석 기법 • 교육성과 측정 기법 • 멀티미디어 작성 도구 활용 능력 • 문서작성 도구 활용 능력 • 성능 측정 도구 활용 능력 • 애플리케이션 모니터링 도구 활용 능력 • 애플리케이션 배포 도구 활용 능력 • 애플리케이션 패키징 기술 • 표준 용어집 • 프리젠테이션 기법 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 고객 및 제품 요구사항에 대하여 적극적으로 분석하는 태도 • 교육효과를 높이기 위한 기자재 및 장소 확보 의지 • 교재의 일관성 확보를 위해 노력하는 태도 • 기술 및 제품 동향에 관한 각종 정보 수집에 대한 적극성 • 다양한 이해당사자의 의견을 긍정적으로 수용할 수 있는 태도 • 사용자 질의사항에 대한 긍정적 태도 • 성공적 시스템 이행을 위한 의지 • 시스템 성능의 문제점 및 성능 개선을 위한 의지 • 시스템 안정화를 위한 의지 • 오류 및 추가 요구사항에 대한 반영 의지 • 제품에 대한 책임감 및 불량에 대한 재발 방지를 위해 노력하는 태도 • 제품의 품질 및 고객 우선주의 태도 • 표준 양식 준수 의지

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성 도구	세트	공용	1
• 빔 프로젝트	대	공용	30
• SW개발 도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록

• 해당없음

※ 자료는 주제료만 제시한 것으로 그 외의 자료는 별도로 확보

○ 과정/과목명 : 2001020209_16v4 제품소프트웨어 패키징

- 훈련개요

훈련목표	제품소프트웨어 패키징이란 개발이 완료된 제품소프트웨어를 고객에게 전달하기 위한 형태로 패키징하고, 설치와 사용에 필요한 제반 절차 및 환경 등 전체 내용을 포함하는 매뉴얼을 작성하며, 제품소프트웨어에 대한 패치 개발과 업그레이드를 위해 버전관리를 수행하는 능력을 함양.
수 준	5수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
제품소프트웨어 패키징하기	1.1 신규 개발, 변경, 개선된 제품소프트웨어의 소스들로부터 모듈들을 빌드하고 고객의 편의성을 고려하여 패키징 할 수 있다. 1.2 이전 릴리즈 이후의 변경, 개선사항을 포함하여 신규 패키징한 제품소프트웨어에 대한 릴리즈 노트를 작성할 수 있다. 1.3 저작권 보호를 위해 암호화/보안 기능을 제공하는 패키징 도구를 활용하여, 제품소프트웨어의 설치, 배포 파일을 생성할 수 있다.
제품소프트웨어 매뉴얼 작성하기	2.1 사용자가 제품소프트웨어를 설치하는데 참조할 수 있도록 제품소프트웨어 설치 매뉴얼의 기본 구성을 수립하고 작성할 수 있다. 2.2 사용자가 제품소프트웨어를 사용하는데 참조할 수 있도록 제품소프트웨어 사용자 매뉴얼의 기본 구성을 수립하고 작성할 수 있다. 2.3 사용자가 제품소프트웨어를 설치하고 사용하는데 필요한 제품소프트웨어의 설치파일 및 매뉴얼을 배포용 미디어로 제작할 수 있다.
제품소프트웨어 버전관리하기	3.1 형상관리 지침을 활용하여 제품소프트웨어의 신규 개발, 변경, 개선과 관련된 버전을 등록할 수 있다. 3.2 형상관리 지침을 활용하여 제품소프트웨어의 신규 개발, 변경, 개선과 관련된 버전 관리 도구를 사용할 수 있다. 3.3 버전 관리 도구를 활용하여 제품소프트웨어에 대한 버전 현황 관리와 소스, 관련 자료 백업을 수행할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 제품소프트웨어 기획 의도에 대한 이해 • 패키징 산출물 및 다이어그램 이해 • 패키징시 필요한 패키징, 복제방지, 저작권 툴 등에 대한 이해 • 형상관리 지침/도구에 대한 이해 • 버전관리 규칙의 준수 • 제품소프트웨어 UX/UX에 대한 이해 • 제품소프트웨어 UX/UX와 일관된 매뉴얼 기술 • 제품소프트웨어 실행 요구환경, 설치 환경에 대한 이해
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 각종 매뉴얼 작성법 • 국제 표준 제품 품질특성에 대한 사항 • 릴리즈 노트 작성법 • 사용자의 제품 실행 환경에 대한 이해 • 소프트웨어 버전관리 및 도구 사용에 대한 이해 • 장애 복구 처리 절차 • 제품 패키징, 배포 관련 표준에 대한 지식 • 제품소프트웨어 관련 시장에 대한 지식 • 제품소프트웨어 사용에 대한 데이터(오류, 자원사용량 등) 수집 및 분석에 대한 이해 • 제품소프트웨어 아키텍처에 대한 이해 • 제품소프트웨어가 적용되는 업무 특성에 대한 이해
기술	<ul style="list-style-type: none"> • API 문서 생성 기술(JavaDoc, doxygen) • 결함관리 도구 활용 능력 • 도움말 생성 기술 • 멀티미디어 작성 도구 활용 능력 • 문서작성 도구 활용 능력 • 빌드 자동화 도구 활용 능력 • 성능 측정 도구 활용 능력 • 소프트웨어 버전관리 도구 활용 능력 • 암호화/보안 모듈 (DRM) 탑재 기술 • 애플리케이션 모니터링 도구 활용 능력 • 애플리케이션 배포 도구 활용 능력 • 애플리케이션 패키징 기술 • 표준 용어집 • 프리젠테이션 기법 • 협업도구 활용 능력 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 고객 및 제품 요구사항에 대하여 적극적으로 분석하는 태도 • 교재의 일관성 확보를 위해 노력하는 태도 • 문장 구성력 및 시각적 표현력을 갖추하고자 노력하는 태도 • 사용자 중심으로 제품소프트웨어 활용에 대해 이해하고자 하는 태도 • 제품소프트웨어 버전관리의 중요성에 대한 이해도 • 제품소프트웨어에 변경의 필요성에 대해 이해하고자 하는 태도성 • 제품소프트웨어의 특성에 대해 이해하고자 하는 태도 • 제품의 품질 및 고객 우선주의 태도 • 표준 양식 준수 의지

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성 도구	세트	공용	1
• 빔 프로젝트	대	공용	30
• SW개발 도구	세트	공용	1
• SW패키징 도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020211_16v4 서버프로그램 구현

- 훈련개요

훈련목표	서버프로그램 구현이란 애플리케이션 설계를 기반으로 개발에 필요한 환경을 구성하고, 프로그래밍 언어와 도구를 활용하여 공통모듈, 업무프로그램과 배치 프로그램을 구현하는 능력을 함양.
수 준	5수준
훈련시간	48시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
개발환경 구축하기	1.1 응용소프트웨어 개발에 필요한 하드웨어 및 소프트웨어의 필요 사항을 검토하고 이에 따라, 개발환경에 필요한 준비를 수행할 수 있다. 1.2 응용소프트웨어 개발에 필요한 하드웨어 및 소프트웨어를 설치하고 설정하여 개발환경을 구축할 수 있다. 1.3 사전에 수립된 형상관리 방침에 따라, 운영정책에 부합하는 형상관리 환경을 구축할 수 있다.
공통 모듈 구현하기	2.1 공통 모듈의 상세 설계를 기반으로 프로그래밍 언어와 도구를 활용하여 업무 프로세스 및 서비스의 구현에 필요한 공통 모듈을 작성할 수 있다. 2.2 소프트웨어 측정지표 중 모듈간의 결합도는 줄이고 개별 모듈들의 내부 응집도를 높인 공통모듈을 구현할 수 있다. 2.3 개발된 공통 모듈의 내부 기능과 제공하는 인터페이스에 대해 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화 할 수 있다.
서버 프로그램 구현하기	3.1 업무 프로세스 맵과 세부 업무 프로세스를 확인할 수 있다. 3.2 세부 업무프로세스를 기반으로 프로그래밍 언어와 도구를 활용하여 서비스의 구현에 필요한 업무 프로그램을 구현할 수 있다. 3.3 개발하고자 하는 목표 시스템의 잠재적 보안 취약성이 제거될 수 있도록 서버 프로그램을 구현할 수 있다. 3.4 개발된 업무 프로그램의 내부 기능과 제공하는 인터페이스에 대해 테스트를 수행할 수 있다.
배치 프로그램 구현하기	4.1 애플리케이션 설계를 기반으로 프로그래밍 언어와 도구를 활용하여 배치 프로그램 구현 기술에 부합하는 배치 프로그램을 구현 할 수 있다. 4.2 목표 시스템을 구성하는 하위 시스템간의 연동 시, 안정적이고 안전하게 동작할 수 있는 배치 프로그램을 구현 할 수 있다. 4.3 개발된 배치 프로그램을 테스트를 수행할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 분석/설계 산출물 및 다이어그램 분석 • 개발 애플리케이션의 주요 매커니즘 및 적용 개발언어 분석 능력 • 코드 이름규칙, 주석처리 방법 • 형상관리 절차 및 기준의 이해 • 서버프로그램 공통 리소스(예:데이터베이스) 활용의 모듈화 이해 • 프로그램 언어별 데이터베이스 연동 방법 이해 • 프로그램 언어별 웹 서버프로그램 개발 방법의 이해 • 서버프로그램에서 적용하는 주요 모듈구성방법(레이어별 패턴)의 이해 • 컴포넌트 코드작성의 이해
----------	--

평가시 고려사항	<ul style="list-style-type: none">• 프로토콜에 따른 클라이언트와 서버 프로그램의 요청 및 응답 메커니즘 이해• 테스트를 위한 유효값 입력, 비유효값 입력, 예외 조건의 이해• 배치프로그램을 위한 데이터베이스와 연관된 메모리 활용방법의 이해• 네트워크 및 파일처리 기반의 배치프로그램 작성방법 이해• 배치프로그램 트랜잭션 처리 방법의 이해
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발에 필요한 하드웨어 및 소프트웨어 사양 • 개발환경에 요구되는 아키텍처 • 결함 관리 • 구현에 요구되는 아키텍처 • 네트워크 개념 • 단위 업무 • 단위테스트 기술동향 및 도구 • 데이터베이스 개념 • 보안 관련 개념 • 분산 처리 개념 • 분산 프로그래밍 개념 • 서버 개발 프레임워크 개념 • 설계 산출물 해석 방법 • 설계 산출물 해석방법 • 소프트웨어 테스트 • 프로그램 언어 활용방법
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • 개발에 필요한 프로그래밍 언어 및 도구 활용 능력 • 네트워크 활용 기술 • 단위테스트 도구 활용기법 • 모듈화 기법 • 보안도구 활용 능력 • 인터페이스 도구/기법 (API, EAI, WebService, 등) • 재사용 기법 • 컴파일러 사용 능력 • 컴파일러 사용법 • 프레임워크 활용 기술 • 프로그래밍언어 활용 기술 • 프로그램 디버깅 기법 • 프로그램 디버깅 기술 • 프로그램 코드 검토 기법 • 프로그램 코드 검토 기술 • 협업도구 활용 능력 • 형상관리 도구 활용 기법 • 형상관리 도구 활용 능력
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준을 준수하려는 태도 • 개발 팀원 간의 원활한 협업을 추구하는 태도 • 기술 및 도구에 대해 적극적으로 배우고자 하는 태도 • 산출물 완성도를 위한 적극적인 태도 • 성공적인 개발을 위한 의지 • 작업환경 개선 의지 • 적극적이고 세밀히 분석하는 자세 • 적극적인 업무 분석 태도 • 제품의 품질 및 고객 우선주의 태도 • 주어진 과제를 완수하는 책임감

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020212_16v4 인터페이스 구현

- 훈련개요

훈련목표	인터페이스 구현이란 모듈간의 분산이 이루어진 경우를 포함하여 단위 모듈간의 데이터 관계를 분석하고 이를 기반으로 한 메커니즘을 통해 모듈간의 효율적인 연계를 구현하고 검증하는 능력을 함양.
수 준	5수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
인터페이스 설계서 확인하기	1.1 인터페이스 설계서를 기반으로 외부 및 내부 모듈 간의 공통적으로 제공되는 기능과 각 데이터의 인터페이스를 확인할 수 있다. 1.2 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 연계가 필요한 인터페이스의 기능을 식별할 수 있다. 1.3 개발하고자 하는 응용소프트웨어와 관련된 외부 및 내부 모듈 간의 인터페이스를 위한 데이터 표준을 확인할 수 있다.
인터페이스 기능 구현하기	2.1 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 세부 설계서를 확인하여 일관되고 정형화된 인터페이스 기능 구현을 정의할 수 있다. 2.2 개발하고자 하는 응용소프트웨어와 연계 대상 모듈 간의 세부 설계서를 확인하여 공통적인 인터페이스를 구현할 수 있다. 2.3 개발하고자 하는 응용소프트웨어와 연계 대상 내외부 모듈 간의 연계 목적을 고려하여 인터페이스 기능 구현 실패 시 예외처리방안을 정의할 수 있다. 2.4 응용소프트웨어와 관련된 내외부 모듈 간의 연계 데이터의 중요성을 고려하여 인터페이스 보안 기능을 적용할 수 있다.
인터페이스 구현 검증하기	3.1 구현된 인터페이스 명세서를 참조하여 구현 검증에 필요한 감시 및 도구를 준비할 수 있다. 3.2 인터페이스 구현 검증을 위하여 외부 시스템과의 연계 모듈 상태를 확인할 수 있다. 3.3 인터페이스 오류처리 사항을 확인하고 보고서를 작성할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 연계 메커니즘에 대한 이해도 • 연계 표준/프로토콜에 대한 이해도 • 미들웨어, 인터페이스 소프트웨어에 대한 이해도 • 모니터링 도구 사용법 및 모니터링 산출물에 대한 이해도
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발 방법론 • 구현에 요구되는 아키텍처 • 네트워크 • 데이터베이스 • 디자인 패턴 • 분산 프로그래밍 • 산업 분야별 업무특성 • 설계 모델링 기법 • 설계 산출물 • 소프트웨어 아키텍처 • 소프트웨어 테스트
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • IDE 및 개발환경 도구 활용 • UML 작성 기술 • 기술영역별 미들웨어/솔루션 활용 • 기술영역별 미들웨어/솔루션 활용 기술 • 단위테스트 도구 활용기법 • 데이터 보안 기술 • 데이터 연계/이관 도구 활용 능력 • 설계 모델링 기술 • 프레임워크 활용 • 프레임워크 활용 능력 • 프로그램 디버깅 기법 • 프로그램 언어 활용 능력 • 프로그램 코드검토 기법 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준을 준수하려는 태도 • 개발 팀원 간의 원활한 협업을 추구하는 태도 • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도를 위한 적극적인 태도 • 성공적인 개발을 위한 의지 • 정확성과 완전성을 기하고자 하는 의지 • 주어진 과제를 완수하는 책임감

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 자료는 주제료만 제시한 것으로 그 외의 자료는 별도로 확보

○ 과정/과목명 : 2001020214_16v4 애플리케이션 배포

- 훈련개요

훈련목표	애플리케이션 배포란 애플리케이션 배포 환경을 구성하고, 구현이 완료된 애플리케이션의 소스 검증 및 빌드를 수행하여 운영 환경에 배포하는 능력을 함양.
수 준	3수준
훈련시간	30시간
훈련가능시설	컴퓨터실
권장훈련방법	집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
애플리케이션 배포 환경 구성하기	1.1 애플리케이션 빌드와 배포를 위한 환경 구성 방안을 계획할 수 있다. 1.2 애플리케이션 배포를 위한 도구와 시스템을 결정할 수 있다. 1.3 결정한 애플리케이션 배포 환경을 위한 도구와 시스템을 설치할 수 있다. 1.4 설치한 시스템과 도구 운영을 위해 상세 구성 및 설정을 할 수 있다.
애플리케이션 소스 검증하기	2.1 정상적으로 작동하는 소프트웨어 빌드를 위해 형상관리 서버로부터 소스코드를 체크 아웃 할 수 있다. 2.2 소스코드 검증 도구를 활용하여 애플리케이션에서 사용한 라이브러리, 소스, 로직 등의 오류가 있는지 여부를 검증할 수 있다. 2.3 소스 코드의 환경 설정, 운영 환경 정보, 대상 시스템 정보 등에 오류가 있는지 확인할 수 있다.
애플리케이션 빌드 하기	3.1 애플리케이션 소스코드 검증 결과 문제가 없는 경우 해당 소스코드를 빌드 시스템으로 이관할 수 있다. 3.2 애플리케이션 빌드 절차에 따른 빌드 스크립트를 작성할 수 있다. 3.3 작성한 빌드 스크립트 또는 도구를 활용하여 애플리케이션 빌드를 실행할 수 있다. 3.4 애플리케이션 빌드 실행 결과를 확인하여 정상적으로 완료되었는지 여부를 확인할 수 있다. 3.5 애플리케이션 빌드 실패 시 문제 내용과 원인을 파악하여 개발자에게 설명할 수 있다.
애플리케이션 배포 하기	4.1 애플리케이션 실행 환경에 대한 정보를 확인할 수 있다. 4.2 애플리케이션 배포 절차에 따라 운영환경에 적용할 수 있다. 4.3 애플리케이션 배포 후 정상적으로 작동하는지 여부를 확인할 수 있다. 4.4 애플리케이션 배포 결과 문제가 발생했을 경우 적용 내용을 이전 상태로 복원할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 애플리케이션 개발 및 운영 환경에 대한 이해 여부 점검 • 애플리케이션 배포 환경 구성을 위한 관련 도구 설치 및 설정 가능 여부 평가 • 빌드 환경 구성에 따른 각종 도구 사용 능력 평가 • 소스코드 검증 기법에 대한 이해를 바탕으로 도구 사용 능력 점검
----------	--

평가지 고려사항	<ul style="list-style-type: none"> • 애플리케이션 빌드 및 배포 절차에 대한 이해 및 도구 사용 능력 확인 • 애플리케이션 빌드 및 배포 스크립트 작성 실습 및 점검 • 애플리케이션 빌드 및 배포 절차에 따라 빌드 실행 능력 점검 • 애플리케이션 빌드 결과에 대한 분석 및 이해 능력 점검 • 애플리케이션 빌드 실패 시 원복 수행 절차 이해 및 실시 여부 확인
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 배포 도구 설치 및 사용법 • 빌드 결과 분석 지식 • 빌드 도구 설치 및 사용법 • 소스코드 검증 기법 • 소스코드 검증 기준 • 소스코드 검증 도구 • 소스코드 동적 검증 분석 기법 • 소스코드 정적 검증 분석 기법 • 애플리케이션 배포 환경 구성 지식 • 애플리케이션 빌드 스크립트 문법 • 애플리케이션 빌드 시스템 가이드 • 애플리케이션 빌드 절차 • 애플리케이션 빌드(Build) 절차 • 애플리케이션 운영 플랫폼 정보 • 애플리케이션 운영 환경 배포 절차 • 애플리케이션 운영 환경 정보 • 애플리케이션 원복 절차 • 형상관리 도구 사용법 • 형상관리 도구 설치 및 사용법 • 형상관리 지침
기술	<ul style="list-style-type: none"> • 배포 도구 명령어 • 배포 스크립트 작성 기술 • 빌드 검증 기술 • 빌드 도구 명령어 • 빌드 스크립트 작성 기술 • 빌드 실행 기술 • 소스코드 검증 기술 • 소스코드 검증 도구 결과 분석 기술 • 소스코드 검증 도구 사용 기술 • 소스코드 확보 기술 • 애플리케이션 빌드 도구 적용 기술 • 애플리케이션 빌드 스크립트 작성 기법 • 애플리케이션 운영 환경 배포 기술 • 애플리케이션 운영 환경 적용 도구 활용 기술 • 애플리케이션 원복 실행 도구 사용법 • 애플리케이션 원복 절차 • 운영 배포 결과 확인 기법 • 형상관리 도구 명령어 • 환경 설정 기법
태도	<ul style="list-style-type: none"> • 검증이 완료된 소스코드는 빌드 시스템으로 정확하게 이관하는 태도 • 다양한 도구와 연동을 통한 최적의 배포 환경을 구성하고자 하는 태도 • 다양한 분석도구를 다루면서 효율적으로 활용하고자 하는 자세 • 문제 발생 애플리케이션에 대한 원상 복구 여부를 판단하는 자세 • 배포 시 문제 발생 시 즉시 원인 파악을 하고자 하는 자세 • 빌드 결과를 이해하고 문제 발생 시 즉시 조치하는 태도 • 빌드 절차를 준수하여 성공적으로 빌드를 수행하고자하는 자세 • 빌드 진행 상태를 모니터링 하면서 결과를 확인하는 태도 • 소스코드 검증 결과 문제 발생시 해결하고자 하는 적극적인 자세

태도	<ul style="list-style-type: none"> • 애플리케이션 배포 이전에 빌드 후 패키징 결과물의 이상 여부를 검증하는 자세 • 애플리케이션 배포 이전에 정확한 환경 정보로 구성했는지 여부를 확인하는 태도 • 애플리케이션 배포 후 정상 작동 여부를 확인하는 자세 • 애플리케이션 원복 결정 시 즉시 절차에 따라 실행하는 태도 • 애플리케이션의 특성을 고려한 배포 환경을 구성하고자 하는 자세 • 애플리케이션의 품질 향상을 위해 다양한 관점에서 소스코드를 검증하는 자세 • 자동화 도구 외 수작업으로 검토 및 리뷰를 통해 검증을 실시하는 자세 • 자동화 도구를 활용하여 소스코드를 검증하는 노력 • 자동화 도구를 효율적으로 다룰 수 있는 자세
----	---

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020215_15v3 프로그래밍 언어 활용

- 훈련개요

훈련목표	프로그래밍 언어 활용이란 응용소프트웨어 개발에 사용되는 프로그래밍 언어의 기초 문법을 적용하고 언어의 특징과 라이브러리를 활용하여 기본 응용소프트웨어를 구현하는 능력을 함양.
수 준	3수준
훈련시간	80시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
기본문법 활용하기	1.1 응용소프트웨어 개발에 필요한 프로그래밍 언어의 데이터 타입을 적용하여 변수를 사용할 수 있다. 1.2 프로그래밍 언어의 연산자와 명령문을 사용하여 애플리케이션에 필요한 기능을 정의하고 사용할 수 있다. 1.3 프로그래밍 언어의 사용자 정의 자료형을 정의하고 애플리케이션에서 사용할 수 있다.
언어특성 활용하기	2.1 프로그래밍 언어별 특성을 파악하고 설명할 수 있다. 2.2 파악된 프로그래밍 언어의 특성을 적용하여 애플리케이션을 구현할 수 있다. 2.3 애플리케이션을 최적화하기 위해 프로그래밍 언어의 특성을 활용 할 수 있다.
라이브러리 활용하기	3.1 애플리케이션에 필요한 라이브러리를 검색하고 선택할 수 있다. 3.2 애플리케이션 구현을 위해 선택한 라이브러리를 프로그래밍 언어 특성에 맞게 구성 할 수 있다. 3.3 선택한 라이브러리를 사용하여 애플리케이션 구현에 적용할 수 있다.

- 평가시 고려사항

평가지 고려사항	<ul style="list-style-type: none"> • 프로그래밍 언어 활용에 대한 기본 소양 점검 • 프로그래밍 언어 특성 활용 능력 점검 • 프로그래밍 언어 라이브러리 적용을 위한 기본 능력 및 사고 점검 • 능력단위요소에 적절한 도구를 활용할 수 있는지를 점검
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 객체지향 프로그래밍 언어 • 디버깅 기법 • 라이브러리 기능에 대한 이해 • 라이브러리 적용을 위한 기술(파일입출력, 데이터입출력, 예외 처리 등)에 대한 이해 • 선언형 언어 • 스크립트 언어 • 애플리케이션 요구사항에 대한 이해 • 절차적 프로그래밍 언어 • 코드 검토 기법 • 프로그래밍 언어 기능 구현을 위한 알고리즘 • 프로그래밍 언어 기본 문법 • 프로그래밍 언어 문법 오류
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • 개발에 필요한 프로그래밍 언어 및 도구 활용 능력 • 라이브러리 검색 능력 • 라이브러리 설치 능력 • 라이브러리 적용 능력 • 라이브러리를 활용한 프로토타입 작성 능력 • 요구사항 구현에 필요한 라이브러리 선택 능력 • 컴파일러 사용 능력 • 코드 검토 능력 • 프로그램 디버깅 기술 • 프로그램 언어 활용 능력 • 프로그램 코드 검토 능력
태도	<ul style="list-style-type: none"> • 기 작성된 코드를 분석하기 위한 치밀한 태도 • 기본 문법을 이해하고 활용하려는 적극적인 자세 • 라이브러리를 모듈화하려는 의지 • 라이브러리를 활용하는 적극적인 태도 • 라이브러리의 영향범위를 최소화하려는 태도 • 라이브러리의 특성을 이해하려는 의지 • 문법 오류를 이해하고 수정하려는 적극적인 태도 • 복잡한 구조 및 절차를 단순화하려는 의지 • 어플리케이션 구현의 방법과 절차를 문서화 하려는 의지 • 요구사항을 준수하려는 태도 • 코드 검토 결과를 수용하고 반영하려는 태도 • 프로그래밍 언어의 특성을 적용하려는 태도 • 프로그래밍 언어의 규칙과 관습을 준수하려는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020216_15v3 응용 SW 기초 기술 활용

- 훈련개요

훈련목표	응용 SW 기초 기술 활용이란 응용소프트웨어개발을 위하여 운영체제, 데이터베이스, 네트워크의 기초 기술을 적용하고 응용개발에 필요한 환경을 구축하는 능력을 함양.
수 준	3수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
운영체제 기초 활용하기	1.1 응용 소프트웨어를 개발하기 위하여 다양한 운영체제의 특징을 설명할 수 있다. 1.2 CLI(Command Line Interface) 및 GUI(Graphic User Interface) 환경에서 운영체제의 기본명령어를 활용할 수 있다. 1.3 운영체제에서 제공하는 작업 우선순위 설정방법을 이용하여 애플리케이션의 작업우선순위를 조정할 수 있다.
데이터베이스 기초 활용하기	2.1 데이터베이스의 종류를 구분하고 응용 소프트웨어 개발에 필요한 데이터베이스를 선정할 수 있다. 2.2 주어진 E-R 다이어그램을 이용하여 관계형 데이터베이스의 테이블을 정의할 수 있다. 2.3 데이터베이스의 기본연산을 CRUD(Create, Read, Update, Delete)로 구분하여 설명할 수 있다.
네트워크 기초 활용하기	3.1 네트워크 계층구조에서 각 층의 역할을 설명할 수 있다. 3.2 응용의 특성에 따라 TCP와 UDP를 구별하여 적용할 수 있다. 3.3 패킷 스위칭 시스템을 이해하고, 다양한 라우팅 알고리즘과 IP 프로토콜을 설명할 수 있다.
기본 개발환경 구축하기	4.1 응용개발을 위하여 선정된 운영체제를 설치하고 운용할 수 있다. 4.2 응용개발에 필요한 개발도구를 설치하고 운용할 수 있다. 4.3 웹서버, DB서버 등 응용개발에 필요한 기반 서버를 설치하고 운용할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 운영체제의 주요 기능, 구조, 특성에 대한 이해 • 데이터베이스의 기본 개념 및 관계형데이터베이스 테이블 구성에 대한 이해 • 네트워크의 기본 개념 및 OSI 7 Layer의 각 특성에 대한 이해 • 기반서버의 주요 기능, 구조, 특성에 대한 이해 • 오픈소스기반의 개발도구에 대한 이해
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • CRUD 연산에 대한 이해 • DBMS 각 유형별 특징 • ER 다이어그램 작성 방법 • IP 주소 체계 • TCP/UDP 방식 개념 • Windows 운영체제 기본 명령어 • 각 운영체제 특징에 대한 이해 • 개발 프로그램 설치 방법 • 네트워크 7 계층의 개념 • 라이브러리 및 필요 패키지 설치 방법 • 리눅스 / 유닉스 계열 운영체제 기본 명령어 • 릴레이션의 특징 • 메모리 관리 기법의 이해 • 무결성 제약조건의 이해 • 버전관리 시스템 구축 • 사용자의 프로그램 실행 환경에 대한 이해 • 운영체제 설치 및 제거 방법 • 운영체제 환경변수 설정 방법 • 인터넷 구성의 개념 • 테이블 선언 및 조작어 • 트랜스포트 방식 개념 • 프로세스 스케줄링 기법의 이해 • 프로토콜 개념 • 필요 응용 소프트웨어에 대한 이해
기술	<ul style="list-style-type: none"> • E-R 다이어그램 작성 기술 • WINDOWS/UNIX/LINUX 인터넷 설정변경 기술 • batch 파일, shell script 활용 능력 • 각 운영체제에 맞는 응용프로그램 선택 능력 • 개발 TOOL 사용 능력 • 개발 TOOL에 맞는 운영체제 선택 능력 • 개발에 필요한 프로그래밍 언어 선택 능력 • 개발환경에 인터넷 구축 기술 • 기본 명령어 활용 능력 • 데이터베이스 개발 TOOL 사용 능력 • 라우터 설정 기술 • 라이브러리 및 필요 패키지 선택 및 활용 능력 • 릴레이션 작성 기술 • 설계 모델링 기술 • 소프트웨어 버전관리 도구 활용 능력 • 오픈소스기반 DBMS 설치 기술 • 운영체제 오류 해결방법 검색 기술 • 운영체제 환경변수 설정 기술 • 인터넷 환경변수 설정 기술 • 작업우선순위 변경 기술 • 테이블 제작 및 관리 언어 활용 능력 • 필요 관련 정보 검색 기술 • 필요 프로그램 관련 정보 검색 기술
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준 절차를 준수하려는 태도 • 개발도구 관련 기술 및 제품에 대한 정보수집에 대한 적극성

태도	<ul style="list-style-type: none"> • 개발도구 설치시 발생한 문제점을 해결하려는 의지 • 네트워크 관련 기술 및 제품에 대한 정보수집에 대한 적극성 • 네트워크 설정 관련 문제점을 해결하려는 의지 • 능동적 태도 • 다양한 운영체제의 특징을 이해하려는 태도 • 데이터베이스 관련 기술 및 제품에 대한 정보수집에 대한 적극성 • 문제점을 해결하려는 의지 • 사용할 프로그램의 특징을 이해하려는 태도 • 상황에 맞게 데이터베이스를 작성하고 변경하는 능동적인 태도 • 오픈소스운동에 참여하려는 의지 • 운영체제 관련 기술 및 제품에 대한 정보수집에 대한 적극성 • 운영체제 활용 중 발생한 문제를 해결하고자 하는 의지 • 자신이 수행한 작업에 대한 평가의 객관성 • 적극적이고 세밀히 각 단계를 분석하는 자세 • 정규화 과정을 지속적으로 관찰하는 태도 • 주어진 문제를 완수하는 책임감 • 콘솔/터미널을 활용하려는 태도 • 활용에 필요한 기술을 이해하려는 의지
----	--

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020217_16v4 애플리케이션 리팩토링

- 훈련개요

훈련목표	애플리케이션 리팩토링이란 소스코드가 수행하는 기능을 유지하면서 코드 가독성 및 잠재적 결함을 제거하도록 코드의 구조를 개선하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
리팩토링 기준 수립하기	1.1 리팩토링을 적용하여 달성하려는 품질 목표를 수립할 수 있다. 1.2 리팩토링 결과의 완전성을 검증할 수 있는 체크리스트를 수립할 수 있다. 1.3 리팩토링에 활용할 수 있는 도구 및 절차를 수립할 수 있다.
소스 코드 분석하기	2.1 개발 표준을 만족하지 않는 소스 코드를 식별할 수 있다. 2.2 성능개선과 이해도 증진을 위하여 불필요한 소스 코드를 식별할 수 있다. 2.3 잠재 결함 요소를 내포한 소드 코드를 식별할 수 있다.
리팩토링 적용하기	3.1 리팩토링을 위한 소스 코드 구조를 개선할 수 있다. 3.2 개선한 소스 코드가 초기 기능을 유지하는지 확인할 수 있다. 3.3 개선한 소스 코드가 리팩토링 체크리스트를 만족하는지 검토할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 리팩토링 전/후의 기능성 유지 여부 확인 • 분석 대상에서 발견한 개선 점의 정확성 여부 확인 • 적용한 패턴의 정확성 확인
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 국제 코딩 표준 • 디자인 패턴 • 리팩토링 기법 • 리팩토링 적용 기법 • 비기능적 품질 특성 • 비기능적 품질 특성 충족 기법 • 코드 스멜 유형 • 클린 코드 특징 • 클린 코드의 필요성 및 특징 • 회귀 테스트
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • 리팩토링 도구 활용 능력 • 설계 패턴 적용 기술 • 역공학 도구 활용 기술 • 좋은 설계 (Good Design) 기준 • 체크리스트 작성 기술 • 코드 동적 분석 도구 활용 기술 • 코드 인스펙션 기술 • 코드 정적 분석 도구 활용 기술 • 테스트 도구 활용 기술 • 프로그램 디버깅 능력
태도	<ul style="list-style-type: none"> • 결함 없는 코드로 개선하려는 태도 • 이해하기 쉽고 확장하기 쉬운 코드로 개선하려는 태도 • 지속적으로 리팩토링을 수행하려는 적극적 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020218_16v4 인터페이스 설계

- 훈련개요

훈련목표	인터페이스 설계란 응용소프트웨어 개발을 위해 정의된 시스템 인터페이스 요구사항을 확인하고 인터페이스 대상을 식별하여 인터페이스 설계서를 작성하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
인터페이스 요구사항 확인하기	1.1 정의 된 응용소프트웨어 요구사항을 참조하여 개발하고자 하는 응용소프트웨어의 인터페이스 비 기능 요구사항을 분석할 수 있다. 1.2 정의 된 응용소프트웨어 요구사항을 참조하여 개발하고자 하는 응용소프트웨어의 인터페이스 기능 요구사항을 분석할 수 있다. 1.3 개발하고자 하는 응용소프트웨어의 인터페이스 요구사항의 정확성과 완전성을 확인한다.
인터페이스 대상 식별하기	2.1 개발하고자 하는 응용소프트웨어의 내부와 외부 인터페이스 대상 시스템을 식별할 수 있다. 2.2 개발하고자하는 응용소프트웨어의 내부와 외부 인터페이스를 위해 필요한 연계 시스템을 식별할 수 있다. 2.3 개발하고자하는 응용소프트웨어의 내부와 외부 인터페이스를 위해 필요한 송수신 데이터를 식별할 수 있다.
인터페이스 상세 설계하기	3.1 개발하고자하는 응용소프트웨어의 내부와 외부 인터페이스를 위한 송수신 방법을 명세화할 수 있다. 3.2 개발하고자하는 응용소프트웨어의 내부와 외부 인터페이스를 위해 필요한 데이터를 명세화할 수 있다. 3.3 개발하고자하는 응용소프트웨어의 내부와 외부 인터페이스의 오류 시 처리 방안을 명세화할 수 있다. 3.4 소프트웨어 아키텍처에서 정의한 인터페이스 설계 기준에 따라 외부와 내부 시스템 간의 인터페이스를 설계서를 작성할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 인터페이스 요구사항 분석 및 확인 절차 • 인터페이스 대상 시스템 및 연계 시스템 식별 과정 • 인터페이스 대상 송수신 데이터 식별 및 구성 • 내외부 시스템간 인터페이스 설계서 작성 절차 • 소프트웨어 아키텍처와 일관성 여부
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 네트워크 • 산업 분야별 업무특성 • 설계 모델링 방법론 • 소프트웨어 아키텍처 • 시스템 아키텍처 • 요구공학 • 인터페이스 시스템
기술	<ul style="list-style-type: none"> • IDE 및 개발환경 도구 활용 • UML 작성 기술 • 기술영역별 미들웨어와 솔루션 활용 능력 • 리뷰 진행 기술 • 설계 모델링 기술 • 요구사항 관리 도구 사용 능력 • 프레임워크 활용
태도	<ul style="list-style-type: none"> • 사용자의 요구사항에 대한 적극적 수용 태도 • 산출물 완성도 확보를 위한 적극적인 자세 • 이해 당사자와의 협업을 위한 적극적 의사소통 • 정확한 요구사항의 이해를 위한 분석 자세 • 정확한 인터페이스 시스템 식별을 위한 의지 • 조직의 개발방법론을 준수하려는 태도 • 조직의 품질관리 기준을 준수하려는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020219_16v4 애플리케이션 요구사항 분석

- 훈련개요

훈련목표	애플리케이션 요구사항 분석이란 구현하고자 하는 애플리케이션의 요구사항을 도출, 분석, 명세화 및 요구사항 검증을 수행하는 능력을 함양.
수 준	7수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
요구사항 도출하기	<ol style="list-style-type: none"> 1.1 비즈니스 목적, 도메인 지식, 이해당사자, 비즈니스 규칙, 조직 환경 등 다양한 요구사항 도출을 위한 정보를 수집 할 수 있다. 요구사항 도출을 위한 정보를 수집 할 수 있다. 1.2 인터뷰, 시나리오 작성, 프로토타입 제작, 그룹회의, 관찰, 사용자 스토리 작성 등 제시된 기법을 통해 어플리케이션의 요구사항을 도출 할 수 있다. 1.3 요구사항 추출 내용을 관리하기 위해 요구사항의 출처와 요구 내용을 상세하게 기록할 수 있다.
요구사항 분석하기	<ol style="list-style-type: none"> 2.1 요구사항 분석과 협상을 위하여 도출한 요구사항을 기능적 요구사항과 비기능적 요구사항을 분류 할 수 있다. 2.2 도출한 요구사항의 내용을 분석하여 구현을 위한 우선순위를 결정할 수 있다. 2.3 도출한 요구사항이 개발하고자 하는 애플리케이션 전체 또는 단위 컴포넌트에 향해 영향을 미치는지 여부를 분석하여 요구사항의 적용 범위를 확인할 수 있다. 2.4 요구사항 분석 결과를 토대로 사용자와 협상을 통해 요구사항, 우선순위, 영향 범위 등을 확정할 수 있다.
요구사항 명세화하기	<ol style="list-style-type: none"> 3.1 애플리케이션 사용자 또는 활용 분야의 관점에서 요구사항을 분석한 결과를 정리한 상위 수준의 사용자 요구사항 정의서를 작성할 수 있다. 3.2 구현하고자 하는 애플리케이션이 구동되는 시스템이나 하드웨어 관련 내용을 구체화 하여 시스템 요구사항 문서를 작성할 수 있다. 3.3 구현하고자 하는 애플리케이션에 대해 설명, 검증 및 테스트, 계약을 위해 사용자 요구사항 정의서와 시스템 요구사항 문서를 토대로 요구사항을 구체화하여 소프트웨어 요구사항 명세서(SRS, Software Requirement Specification)를 작성할 수 있다.
요구사항 검증하기	<ol style="list-style-type: none"> 4.1 소프트웨어 요구사항 명세서가 문서 표준을 준수하여 사용자가 이해 가능한 내용으로 일관성 있고, 완성도 있게 작성되었는지를 검증할 수 있다. 4.2 작성한 요구사항 명세서의 오류, 잘못된 가정, 불명확한 설명, 비표준적인 내용 등이 있는지 확인을 위해 조사(Inspection) 혹은 검토(Review)를 실시할 수 있다. 4.3 기술 환경, 구현 가능성, 안정성 등을 고려하여 요구된 도출 사항들을 검증할 수 있다. 4.4 분석 단계의 개념 모델링 결과물이 있을 경우 표기법, 모델 원칙 준수 여부 판단 등을 통해 모델을 검증할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 요구사항 도출 기법에 대한 이해 • 사용자 요구사항 정의서 작성 방법 • 사용되는 시스템이나 하드웨어에 대한 이해 • 사용되는 소프트웨어에 대한 이해
----------	--

평가시 고려사항	<ul style="list-style-type: none">• 소프트웨어 요구사항 명세서 작성 방법에 대한 이해• 도출한 요구사항에 대한 기준 수립에 대한 이해• 요구사항 명세서에 대한 검증 및 검토에 대한 이해
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • BABOK (Business Analysis Body of Knowledge) • BABOK (Business Analysis Body of Knowledge) • IEEE830 (IEEE Recommended Practice for Software Requirements Specifications) • SW 제품 개발 수명 주기 이론 • SW 제품 요구사항 관리 절차 • SW 제품 요구사항 도출 지식 • SW 제품 요구사항 정의 기준 • SWEBOK(Software Engineering Body of Knowledge) • SW제품 개발 요구 공학 • SW제품 품질 속성 항목 • SW제품 품질 평가 항목 • 요구 공학(Requirement Engineering) 이론 • 요구사항 명세서(Software Requirement Specification) 작성 기준 • 요구사항 영향범위 설정 기준 • 요구사항 우선 순위 결정 기준 • 요구사항 적용 범위 확인 지식 • 정보시스템마스터플랜 ISMP(Information System Master Plan) 방법론 • 정보시스템의 기술적 지식
기술	<ul style="list-style-type: none"> • SW제품 요구사항 변경 및 추적 기술 • 그룹 회의 토의 진행 기법 • 비기능 요구사항 식별 기술 • 사용자 스토리 작성 기법 • 사용자와 협상 능력 • 상위레벨 SW제품 아키텍처 수립 기술 • 수집된 자료에 대한 체계적인 분류 및 정리 기술 • 시나리오 작성 기법 • 요구사항 검증 기법 • 요구사항 검증 기준 • 요구사항 기록 방법 • 요구사항 도출 기법 • 요구사항 명세 기술 • 요구사항 명세서 검토 기법 • 요구사항 명세서 조사 기법 • 요구사항 분류 기술 • 요구사항 분석 기술 • 요구사항 분석 능력 • 요구사항 정의서 문서 작성 능력 • 요구사항의 우선순위를 식별하는 능력 • 이해당사자와 커뮤니케이션 능력 • 인터뷰 기법 • 인터뷰 기술 • 품질 속성 Trade-off 방법 • 프로토타입 제작 기법
태도	<ul style="list-style-type: none"> • 개발 제품의 요건을 명확하게 도출 할 수 있는 태도 • 고객관점의 요구사항을 이해하려는 태도 • 구체적이고 명확한 요구사항 정의서를 작성하고자 하는 태도 • 구체적인 요구사항 도출을 위해 다양한 기법을 활용하고자 하는 열린 자세 • 다양한 출처로 부터의 요구사항을 명확하게 도출하고자 노력하는 태도

태도	<ul style="list-style-type: none"> • 도출한 요구사항을 사용자 관점에서 정확하게 이해하고자 하는 태도 • 사용자 요구사항 반영 여부를 확인하고자 하는 태도 • 요구 분석 내용을 체계적으로 이해하고자 하는 태도 • 요구사항 도출 후 정확하게 기록하고자 하는 태도 • 요구사항 명세서의 내용을 고객과 개발팀이 이해할 수 있도록 기술하는 노력 • 요구사항 명세서의 오류를 발견하여 조치하고자 하는 태도 • 요구사항 명세서의 일관성과 완성도를 검증하고자 하는 자세 • 요구사항 분석 기준을 통해 우선순위를 결정하는 태도 • 요구사항 영향범위를 명확히하고자 하는 태도 • 요구사항 정의서 양식에 맞추어 체계적으로 작성하고자 하는 태도 • 요구사항을 명확하게 하기 위해 사용자와 협상하는 태도 • 요구사항의 구현 가능성, 안정성, 기술적 환경 분석 자세
----	---

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020220_16v1 기능 모델링

- 훈련개요

훈련목표	기능모델 설계란 애플리케이션 요구사항에 의해 도출, 분석된 내용을 애플리케이션의 기능으로 명세화하여 요구사항을 검증하고 소프트웨어 개발범위를 확정하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
기능 모델 작성하기	1.1 기능모델을 작성하기 위해 필요한 다이어그램을 선정할 수 있다. 1.2 기능모델링을 위해 선정한 다이어그램의 각 구성요소를 식별하여 다이어그램을 작성할 수 있다. 1.3 기능모델링을 위해 선정한 다이어그램을 설명하기 위한 기능모델 설계 명세서를 작성할 수 있다.
기능 모델 검증하기	2.1 도식된 기능모델이 표준 표기법을 준수하고 있는지 검증할 수 있다. 2.2 작성된 기능모델이 표준으로 제시된 산출물 양식과 지침을 따르고 있는지 검증할 수 있다. 2.3 작성된 기능모델이 정의된 소프트웨어 요구사항을 모두 포함하고 있는지 검증할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 요구사항 도출 기법에 대한 이해 • 사용자 요구사항 정의서 작성 방법 • 사용되는 시스템이나 하드웨어에 대한 이해 • 사용되는 소프트웨어에 대한 이해 • 소프트웨어 요구사항 명세서 작성 방법에 대한 이해 • 도출한 요구사항에 대한 기준 수립에 대한 이해 • 요구사항 명세서에 대한 검증 및 검토에 대한 이해
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • SWEBOK(Software Engineering Body of Knowledge) • 개발 방법론 이해 • 업무 특성에 대한 이해 • 요구 공학(Requirement Engineering) 이론 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • UML 작성 기술 • 모델 검증 기술 • 모델링 기술 • 모델링 명세화 기술 • 의사소통 기술
태도	<ul style="list-style-type: none"> • 고객과의 지속적인 소통을 위한 책임감 있는 자세 • 고객과의 지속적인 소통을 위한 책임감 있는 태도 • 고객의 요청을 적극적으로 수용하고자 하는 의지 • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 기능모델의 일관성과 완성도를 검증하고자 하는 자세 • 기능의 누락이나 오류를 발견하여 조치하고자 하는 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도 확보를 위한 적극적인 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020221_16v4 애플리케이션 설계

- 훈련개요

훈련목표	애플리케이션 설계란 요구사항 확인을 통한 상세 분석 결과, 소프트웨어 아키텍처 가이드라인 및 소프트웨어 아키텍처 산출물에 의거하여 이에 따른 애플리케이션 구현을 수행하기 위해 공통모듈 설계, 타 시스템 연동에 대하여 상세 설계하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
공통 모듈 설계하기	1.1 재사용성 확보와 중복개발을 회피하기 위하여, 전체 시스템 차원과 단위 시스템 차원의 공통 부분을 식별하여 이에 대한 상세 명세를 작성할 수 있다. 1.2 개발할 응용소프트웨어의 전반적인 기능과 구조를 이해하기 쉬운 크기로 공통 모듈을 설계할 수 있다. 1.3 소프트웨어 측정지표 중 모듈간의 결합도는 줄이고 개별 모듈들의 내부 응집도는 높이기 위한 공통모듈을 설계할 수 있다. 1.4 전반적인 처리 논리 구조에 예기치 못한 영향을 끼치지 않도록 공통 모듈 인터페이스의 인덱스 번호나 기능 코드를 설계할 수 있다.
타 시스템 연동설계하기	2.1 소프트웨어 아키텍처에서 정의한 타 시스템 연동 리스트 및 연동 방안을 참조하여, 타 시스템 연동 상세 설계의 가이드라인을 작성할 수 있다. 2.2 소프트웨어 아키텍처의 정의를 반영한 연동 상세 설계 가이드라인에 따라, 타 시스템 연동 상세 설계할 수 있다. 2.3 소프트웨어 아키텍처에 따라 선정된 개발 및 운영 환경에 사용될 기술영역별 미들웨어솔루션에 대하여 명세를 작성할 수 있다. 2.4 소프트웨어 아키텍처에 따른 시스템간의 연동 시, 발생할 수 있는 오류를 예측하고 이의 대응 방안에 대해 제시할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> 요구사항 추적성 확보 소프트웨어 아키텍처와 일관성 확인 프로토타입 작성 및 검증 능력
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발 방법론 이해 • 네트워크 이해 • 분산 프로그래밍에 대한 이해 • 설계 모델링 기법 • 소프트웨어 아키텍처에 대한 이해 • 업무 특성에 대한 이해 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • E-R 모델 작성 기술 • IDE 및 개발환경 도구 활용 • UML 작성 기술 • 기술영역별 미들웨어/솔루션 활용 • 설계 모델링 기술 • 프레임워크(애플리케이션, 웹, 데이터 액세스, 테스트 자동화, 기타) 활용
태도	<ul style="list-style-type: none"> • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도를 위한 적극적인 태도 • 아키텍트와 설계자간의 협업을 추구하는 태도 • 주어진 과제를 완수하는 책임감 • 책임감 및 설계문서에 완벽함을 추구하는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020222_16v4 정적모델 설계

- 훈련개요

훈련목표	정적모델 설계란 요구사항 확인을 통한 상세 분석 결과, 소프트웨어 아키텍처 가이드라인 및 소프트웨어 아키텍처 산출물에 의거하여 애플리케이션 구현을 위한 정적 모델을 설계하고 검증하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
정적 분석모델 검증하기	1.1 소프트웨어 아키텍처 설계 가이드라인을 참조하여 정적모델 상세분석 내역을 확인할 수 있다. 1.2 분석된 정적모델을 대상으로 분석모델의 정합성을 검증할 수 있다.
정적모델 상세화하기	2.1 분석된 정적모델을 정적 설계모델로 변환하기 위해 변환규칙을 정의할 수 있다. 2.2 정적 설계모델의 엔티티 클래스, 경계 클래스, 제어 클래스를 상세화할 수 있다. 2.3 상세화된 정적 설계모델의 명세서를 작성할 수 있다. 2.4 설계용 도구를 활용하여 정적 설계모델을 도식할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> 요구사항 추적성 확보 소프트웨어 아키텍처와 일관성 확인 프로토타입 작성 및 검증 능력
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발 방법론 이해 • 네트워크 이해 • 데이터베이스 이해 • 설계 모델링 기법 • 소프트웨어 아키텍처에 대한 이해 • 업무 특성에 대한 이해 • 해당 산업 분야에 대한 업무 지식 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • E-R 모델 작성 기술 • IDE 및 개발환경 도구 활용 기술 • UML 작성 기술 • 기술영역별 미들웨어/솔루션 활용 기술 • 모델 검증 기술 • 모델링 기술 • 설계 모델링 기술 • 프레임워크(애플리케이션, 웹, 데이터 액세스, 테스트 자동화, 기타) 활용 기술
태도	<ul style="list-style-type: none"> • 고객과의 지속적인 소통을 위한 책임감 있는 태도 • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 도출된 모듈 및 컴포넌트를 분석하기 위한 치밀한 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도 확보를 위한 적극적인 태도 • 산출물 완성도를 위한 적극적인 태도 • 아키텍트와 설계자간의 협업을 추구하는 태도 • 작업환경 개선 의지 • 주어진 과제를 완수하는 책임감 • 책임감 및 설계문서에 완벽함을 추구하는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020223_16v4 동적모델 설계

- 훈련개요

훈련목표	동적모델 설계란 요구사항 확인을 통한 상세 분석 결과, 소프트웨어 아키텍처 가이드라인 및 소프트웨어 아키텍처 산출물에 의거하여 애플리케이션 구현을 위한 동적 모델을 설계하고 검증하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
동적 분석모델 검증하기	1.1 소프트웨어 아키텍처 설계 가이드라인을 참조하여 동적모델 상세분석 내역을 확인할 수 있다. 1.2 분석된 동적모델을 대상으로 분석모델의 정합성을 검증할 수 있다.
동적모델 상세화하기	2.1 검증된 동적 분석모델을 동적 설계모델로 상세화하기 위한 적절한 다이어그램을 선정할 수 있다. 2.2 동적 설계모델을 선정한 다이어그램을 이용하여 상세 설계할 수 있다. 2.3 상세 설계된 동적 모델을 설계용 도구를 활용하여 도식화할 수 있다. 2.4 도식화한 동적 설계모델을 통해 기능모델과 정적모델을 상호 검증하고 조정할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> 요구사항 추적성 확보 소프트웨어 아키텍처와 일관성 확인 프로토타입 작성 및 검증 능력
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 개발 방법론 이해 • 네트워크 이해 • 분산 프로그래밍에 대한 이해 • 설계 모델링 기법 • 소프트웨어 아키텍처에 대한 이해 • 업무 특성에 대한 이해 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • E-R 모델 작성 기술 • IDE 및 개발환경 도구 활용 • IDE 및 개발환경 도구 활용 기술 • UML 작성 기술 • 기술영역별 미들웨어/솔루션 활용 • 설계 모델링 기술 • 프레임워크(애플리케이션, 웹, 데이터 액세스, 테스트 자동화, 기타) 활용 • 프레임워크(애플리케이션, 웹, 데이터 액세스, 테스트 자동화, 기타) 활용 기술
태도	<ul style="list-style-type: none"> • 고객의 요청을 적극적으로 수용하고자 하는 태도 • 도출된 모듈/기능 컴포넌트를 분석하기 위한 치밀한 태도 • 산출물 완성도를 위한 적극적인 태도 • 아키텍트와 설계자간의 협업을 추구하는 태도 • 주어진 과제를 완수하는 책임감 • 책임감 및 설계문서에 완벽함을 추구하는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020224_16v4 화면 설계

- 훈련개요

훈련목표	화면 설계란 요구사항분석 단계에서 파악된 화면에 대한 요구사항을 소프트웨어 아키텍처 단계에서 정의된 구현 지침 및 UI/UX 엔지니어가 제시한 UI표준과 지침에 따라 화면을 설계하는 능력을 함양.
수 준	5수준
훈련시간	48시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
UI 요구사항 확인하기	1.1 응용소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, 개발하고자 하는 응용소프트웨어에 적용될 UI 요구사항을 확인할 수 있다. 1.2 응용소프트웨어 개발을 위한 UI 표준 및 지침에 의거하여, UI 요구사항을 반영한 프로토타입을 제작할 수 있다. 1.3 작성한 프로토타입을 활용하여 UI/UX엔지니어와 향후 적용할 UI의 적정성에 대해 검토할 수 있다.
UI 설계하기	2.1 UI 요구사항과 UI 표준 및 지침에 따라, 화면과 폼의 흐름을 설계하고, 제약사항을 화면과 폼 흐름 설계에 반영할 수 있다. 2.2 UI 요구사항과 UI 표준 및 지침에 따라, 사용자의 편의성을 고려한 메뉴 구조를 설계할 수 있다. 2.3 UI 요구사항과 UI 표준 및 지침에 따라, 하위 시스템 단위의 내·외부 화면과 폼을 설계할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • UI 요건에 대한 이해도 • UI 설계서에 대한 이해도 • UI 표준 및 지침에 대한 이해도 • 최신 UI 기술들에 대한 이해도 • 사용자 경험 분석에 대한 이해도 • 프로토타입 작성 및 검증 능력 • 소프트웨어 아키텍처 상세구현 지침에 대한 이해도 • UI 설계를 소프트웨어 아키텍처 상세구현 지침과 UI 표준 및 지침에 따라 개발하는 능력
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • UI 기획/설계/구현 절차 • UI 표준 및 지침 • 감성공학에 대한 이해 • 구현에 요구되는 아키텍처의 이해 • 데이터 타입 특성 • 사용자 기반 메뉴 구조 작성 방법 • 유용성 개념 및 적용원리 • 유즈케이스에 대한 이해 • 프로토타입 작성법
기술	<ul style="list-style-type: none"> • IDE 도구 활용 능력 • UI 네비게이션 작성 도구 활용 능력 • UI 설계도구 활용 능력 • UI 작성 그래픽 도구 활용 능력 • 스토리보드 작성 기술 • 유즈케이스 작성 능력 • 프로토타입 도구 사용 능력 • 화면 구성 능력 • 화면/폼 구성 도구 활용 능력
태도	<ul style="list-style-type: none"> • UI 관련자들과 원활한 관계를 유지하려는 태도 • UI 설계 변경 요구 피드백에 대해 열린 마음으로 듣는 태도 • UI 작성 그래픽 도구 활용 능력 • UI 표준 및 지침을 준수하려는 태도 • 개발 일정을 준수하려는 태도 • 고객을 적극적으로 이해하고 수용하기 위한 자세 • 고품질의 소프트웨어 개발에 대한 의지 • 기술적 위험에 적극적으로 대비하려는 노력

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020225_16v4 화면 구현

- 훈련개요

훈련목표	화면 구현이란 UI 요구사항을 확인하여 설계한 UI 설계를 기반으로 화면을 구현하는 능력을 함양.
수 준	3수준
훈련시간	48시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
UI 설계 확인하기	1.1 설계된 화면과 폼의 흐름을 확인하고, 제약사항과 화면의 폼 흐름을 구현에 반영하도록 설계를 확인 할 수 있다. 1.2 UI 요구사항과 UI 표준 및 지침에 따라 설계된 메뉴 구조를 해석 할 수 있다. 1.3 구현을 위해 하위 시스템 단위의 내·외부 화면과 폼을 설계를 확인할 수 있다.
UI 구현하기	2.1 소프트웨어 아키텍처 세부 구현 지침과 UI 표준 및 지침을 반영하여, 확인된 UI 설계를 구현할 수 있다. 2.2 확인된 화면과 폼 흐름 설계에 따라, 사용자 접근성을 고려한 화면과 폼의 흐름 제어를 구현할 수 있다. 2.3 확인된 화면과 폼 흐름 설계에 따라, 감성공학 기법을 고려하여 사용자가 접하는 화면, 폼, 메뉴, 흐름을 구현할 수 있다. 2.4 구현된 화면, 폼, 메뉴, 흐름을 테스트할 수 있는 테스트 케이스를 작성하고 단위 테스트를 수행하기 위한 테스트 조건을 명세화 할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • UI 요건에 대한 이해도 • UI 설계서에 대한 이해도 • UI 표준 및 지침에 대한 이해도 • 최신 UI 기술들에 대한 이해도 • 사용자 경험 분석에 대한 이해도 • 프로토타입 작성 및 검증 능력 • 소프트웨어 아키텍처 상세구현 지침에 대한 이해도 • UI 설계를 소프트웨어 아키텍처 상세구현 지침과 UI 표준 및 지침에 따라 개발하는 능력
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • CSS의 개념과 적용원리 • HTML 개념과 적용원리 • UI 기획/설계/구현 절차 • 감성공학에 대한 이해 • 구현에 요구되는 아키텍처의 이해 • 데이터 타입 특성 • 사용자 기반 메뉴 구조 작성 방법 • 설계 산출물의 이해 • 유용성 개념 및 적용원리 • 유즈케이스에 대한 이해 • 자바스크립트의 개념과 적용원리
기술	<ul style="list-style-type: none"> • HTML, CSS, 자바스크립트 등 표준기술 활용 능력 • UI Framework(jQuery, AngularJS등) 활용 능력 • UI 네비게이션 작성 도구 활용 능력 • UI 설계도구 활용 능력 • UI 작성 그래픽 도구 활용 능력 • 유즈케이스 작성 능력 • 화면 구성 능력 • 화면/폼 구성 도구 활용 능력
태도	<ul style="list-style-type: none"> • UI 관련자들과 원활한 관계를 유지하려는 태도 • UI 설계 변경 요구 피드백에 대해 열린 마음으로 듣는 태도 • UI 표준 및 지침을 준수하려는 태도 • 개발 일정을 준수하려는 태도 • 고품질의 소프트웨어 개발에 대한 의지

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020226_16v4 애플리케이션 테스트 관리

- 훈련개요

훈련목표	애플리케이션 테스트 관리란 요구사항대로 응용소프트웨어가 구현되었는지를 검증하기 위해서 테스트케이스를 작성하고 개발자 통합 테스트를 수행하여 애플리케이션의 성능을 개선하는 능력을 함양.
수 준	5수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
애플리케이션 테스트케이스 설계하기	1.1 개발하고자 하는 응용소프트웨어의 특성을 반영한 테스트 방식, 대상과 범위를 결정하여 테스트케이스를 작성 할 수 있다. 1.2 개발하고자 하는 응용소프트웨어의 특성을 반영한 테스트 방식, 대상과 범위가 적용된 시나리오를 정의할 수 있다. 1.3 애플리케이션 테스트 수행에 필요한 테스트 데이터, 테스트 시작 및 종료 조건 등을 준비 할 수 있다.
애플리케이션 통합 테스트하기	2.1 개발자 통합테스트 계획에 따라 통합 모듈 및 인터페이스가 요구사항을 충족하는지에 대한 테스트를 수행할 수 있다. 2.2 개발자 통합테스트 수행 결과 발견된 결함에 대한 추이 분석을 통하여 잔존 결함을 추정할 수 있다. 2.3 개발자 통합테스트 결과에 대한 분석을 통해 테스트의 충분성 여부를 검증하고, 발견된 결함에 대한 개선 조치사항을 작성할 수 있다.
애플리케이션 성능 개선하기	3.1 애플리케이션 테스트를 통하여 애플리케이션의 성능을 분석하고, 성능 저하 요인을 발견할 수 있다. 3.2 코드 최적화 기법, 아키텍처 조정 및 호출 순서 조정 등을 적용하여 애플리케이션 성능을 개선할 수 있다. 3.3 프로그래밍 언어의 특성에 대한 이해를 기반으로 소스코드 품질 분석 도구를 활용하여 애플리케이션 성능을 개선할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 애플리케이션 성능 개선 평가 시 문제해결 시나리오를 활용하여 문제 해결 여부를 확인한다. • 사례연구는 개발자 테스트 수행 사례를 도출하여 점검한다. • 평가자 질문은 개발자 테스터에게 질문을 통하여 능력 향상여부를 확인한다. • 평가자 체크리스트는 개발자 테스트 수행 결과를 평가자가 직접 점검하여 체크리스트에 반영한다. • 피평가자 체크리스트는 개발자 테스터가 직접 자가 점검할 때에 사용한다.
----------	--

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 결함 관리 • 결함 관리에 대한 지식 • 구현에 요구되는 아키텍처 • 국제 표준 제품 품질특성(ISO 25000) • 국제 표준 제품 품질특성에 대한 사항 • 산업 분야별 업무특성 • 성능 평가 기준 • 소스코드 인스펙션 • 소프트웨어 아키텍처 국제 표준(IEEE 1471) • 소프트웨어 아키텍처에 대한 이해 • 알고리즘 • 업무 특성에 대한 이해 • 자료구조 • 테스트 레벨(단위/통합/시스템/인수)별 테스트 접근 방법 • 테스트 지식 체계(ISTQB Syllabus, CSTE CBOK) • 테스트 지식 체계(ISTQB Syllabus, CSTE CBOK)에 대한 지식 • 프로그래밍 표준 • 프로그램 언어 활용 방법 • 해당 산업 분야에 대한 지식
기술	<ul style="list-style-type: none"> • APM(Application Performance Management) 기술 • IDE 도구 활용 능력 • 개발에 필요한 프로그래밍 언어 표준과 코드 검토 기법 • 성능 측정 도구 활용 능력 • 소스코드 인스펙션 능력 • 소스코드 표준 문서화 능력 • 소스코드 품질분석 도구(PMD, SONAR 등) • 컴파일러 사용법 • 테스트 자동화 도구 활용기법 • 표준 단어, 용어, 도메인, 코드에 대한 정의 능력 • 프로그램 디버깅 기법 • 프로그램 코드 검토 기법 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 개발 수행에 요구되는 표준을 준수하려는 태도 • 고객 및 제품 요구사항에 대하여 적극적으로 분석하는 자세 • 고객 및 제품 요구사항에 대하여 적극적으로 분석하는 태도 • 기술 및 제품 동향에 관한 각종 정보 수집에 대한 적극성 • 적극적이고 세밀히 분석하는 자세 • 제품에 대한 책임감 및 불량에 대한 재발 방지를 위해 노력하는 태도 • 제품의 품질 및 고객 우선주의 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020227_16v4 애플리케이션 테스트 수행

- 훈련개요

훈련목표	애플리케이션 테스트 수행이란 요구사항대로 응용소프트웨어가 구현되었는지를 검증하기 위해서 분석된 테스트 케이스에 따라 테스트를 수행하고 결함을 조치하는 능력을 함양.
수 준	3수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
애플리케이션 테스트 수행하기	1.1 애플리케이션 테스트 계획에 따라 서버모듈, 화면모듈, 데이터입출력, 인터페이스 등 기능단위가 요구사항을 충족하는지에 대한 테스트를 수행할 수 있다. 1.2 애플리케이션 테스트 수행으로 발견된 결함을 유형별로 기록 할 수 있다. 1.3 애플리케이션 테스트 수행 결과 발견된 결함에 대해서 원인을 분석하고 개선 방안을 도출할 수 있다.
애플리케이션 결함 조치하기	2.1 애플리케이션 테스트 수행 결과에서 발견된 결함을 식별하고 조치에 대한 우선순위를 결정하고 적용할 수 있다. 2.2 결함이 발생한 소스를 분석하고 기존에 구현된 로직과의 연관성을 고려하여 부작용이 최소화되도록 결함을 제거할 수 있다. 2.3 애플리케이션 테스트 결과 결함 조치로 변경되는 소스의 버전을 관리하고 결함 조치 결과에 대한 이력을 관리할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 테스트 계획의 구성항목을 이해하고 테스트 케이스 작성 능력 • 테스트 전략을 이해하고 테스트 케이스 작성 시 적용할 수 있는 능력 • 테스트 케이스를 이해하고 테스트를 수행할 수 있는 능력 • 테스트 관련 국제 표준 지식체계에 대하여 이해하고 숙지하고 있는지의 여부
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 결함 관리 • 국제 표준 제품 품질특성(ISO 25000) • 산업 분야별 업무특성 • 소프트웨어 아키텍처 국제 표준(IEEE 1471) • 테스트 레벨(단위/통합/시스템/인수)별 테스트 접근 방법 • 테스트 지식 체계(ISTQB Syllabus, CSTE CBOK)
기술	<ul style="list-style-type: none"> • APM(Application Performance Management) 기술 • IDE 도구 활용 능력 • 컴파일러 사용법 • 테스트 자동화 도구 활용기법 • 프로그램 디버깅 기법 • 프로그램 코드 검토 기법 • 형상관리 도구 활용 기법
태도	<ul style="list-style-type: none"> • 고객 및 제품 요구사항에 대하여 적극적으로 분석하는 태도 • 기술 및 제품 동향에 관한 각종 정보 수집에 대한 적극성 • 적극적이고 세밀히 분석하는 자세 • 제품에 대한 책임감 및 불량에 대한 재발 방지를 위해 노력하는 태도 • 제품의 품질 및 고객 우선주의 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주재료만 제시한 것으로 그 외의 재료는 별도로 확보

○ 과정/과목명 : 2001020228_16v4 소프트웨어공학 활용

- 훈련개요

훈련목표	소프트웨어공학 활용이란 응용 소프트웨어 개발과 프로세스 적용활동의 관련된 지식을 소프트웨어의 완전성을 보장하고, 소프트웨어 품질을 평가하기 위해 CASE 도구와 형상관리를 통해 소프트웨어 공학 기술을 적용하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
CASE 도구 활용하기	1.1 개발하고자 하는 응용소프트웨어에 적용할 개발방법론을 지원하는 최적의 CASE 도구를 선정할 수 있다. 1.2 CASE 도구가 제공하는 다양한 기능들 중 응용 소프트웨어 개발 시 활용할 기능을 식별할 수 있다. 1.3 CASE 도구 활용을 위한 절차와 표준을 정의하고 CASE 도구 사용 중 발생하는 이슈를 해결할 수 있다.
품질요구사항 확인하기	2.1 요구사항 명세서에 기술된 요구사항을 바탕으로 품질 표준을 정의하고 품질 평가항목과 지침을 제공할 수 있다. 2.2 요구사항 명세서에 기술된 요구사항들이 품질 표준에 따라 올바르게 기술되었는지를 검증하기 위한 품질 특성과 평가 기준을 제공할 수 있다. 2.3 개발 공정 품질 관점에서 표준 준수 여부를 확인하고, 응용소프트웨어 제품 품질 관점에서 결함을 식별하여 해결할 수 있다.

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 소프트웨어 공학기술에 대한 기본 소양 점검 • 소프트웨어 개발방법론을 적용하기 위한 기본 능력 및 사고 점검 • 관련 업무 영역의 개선 수행을 위한 소양 점검 • 관련 업무 경험의 실효성/충분성 점검 • 능력단위요소에 적절한 도구를 활용할 수 있는지를 점검
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 감사 프로세스 • 국제 표준 제품 품질특성에 대한 사항 • 비용산정 모델 • 소프트웨어 개발 방법론 • 소프트웨어 개발 표준 • 소프트웨어 개발 프레임워크에 대한 지식 • 소프트웨어 요구사항 명세 기준에 따른 품질 평가에 대한 지식 • 시정 조치 관리 프로세스 • 아키텍처 평가 • 요구공학 방법론 • 품질 관리 프로세스 • 품질 점검 기법(리뷰, 워크스루, 인스펙션 등) • 프로젝트 관리 방법론
기술	<ul style="list-style-type: none"> • UML 작성기술 • 감사 기술 • 리뷰 진행 기술 • 모델링 기법 • 소프트웨어 요구사항 품질 측정, 점검 기술 • 요구사항명세서 작성기술 • 유즈케이스 모델 정의할 수 있는 기술 • 의사결정 기술 • 인스펙션 진행 기술 • 프로세스 테일러링 기술 • 협상 기술 • 형상관리 기법
태도	<ul style="list-style-type: none"> • 객관적인 평가방법을 작성하려는 객관성 • 문장 구성력 및 시각적 표현력을 갖추고자 노력하는 태도 • 새로운 공학도구를 사용하고 하는 도전의식 • 소프트웨어 개발방법론을 적용하기 위해 프로젝트를 분석하려는 자세 • 소프트웨어의 품질에 대해 이해하고자 하는 태도 • 표준을 이해하고 준수하려는 의지 • 품질 측정의 근거를 확보하기 위해 노력하는 태도 • 품질요구사항에 대해 정확한 기준을 제시하려는 태도 • 품질요구사항에 대해 철저히 분석하려는 태도

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 자료는 주제료만 제시한 것으로 그 외의 자료는 별도로 확보

○ 과정/과목명 : 2001020229_16v4 소프트웨어개발 방법론 활용

- 훈련개요

훈련목표	소프트웨어개발방법론 활용이란 응용소프트웨어 특성에 따라 정형화된 개발방법론을 선정하고, 프로젝트 특성에 맞도록 테일러링 하여 응용소프트웨어 개발에 활용하는 능력을 함양.
수 준	6수준
훈련시간	40시간
훈련가능시설	컴퓨터실
권장훈련방법	원격훈련, 집체훈련 실습

- 편성내용

단 원 명 (능력단위 요소명)	훈 련 내 용 (수행준거)
소프트웨어개발 방법론 선정하기	1.1 개발하여야 할 응용소프트웨어의 특성을 파악하여 정형화된 개발방법론 선정기준으로 활용할 수 있다. 1.2 정형화된 개발방법론의 특징을 고려하여 응용소프트웨어 특성에 맞는 개발방법론을 선정할 수 있다.
소프트웨어개발 방법론 테일러링하기	2.1 프로젝트의 일정, 비용, 투입자원과 재사용 현황을 고려하여 반복전략을 개발방법론에 반영할 수 있다. 2.2 확정된 생명주기와 개발방법론에 맞춰 소프트웨어 개발 단계, 활동, 작업, 절차 및 각 단계별 종결기준을 정의할 수 있다. 2.3 응용소프트웨어 개발에 사용할 표준으로 활동 및 절차 수행에 필요한 기법과 표준, 산출물 표준 양식 및 작성 기법, 적용도구를 정립할 수 있다. 2.4 테일러링 된 적용 개발방법론에 따른 작성 산출물 유형을 확정할 수 있다

- 평가시 고려사항

평가시 고려사항	<ul style="list-style-type: none"> • 소프트웨어 공학기술에 대한 기본 소양 점검 • 소프트웨어 개발방법론을 적용하기 위한 기본 능력 및 사고 점검 • 관련 업무 영역의 개선 수행을 위한 소양 점검 • 관련 업무 경험의 실효성/충분성 점검 • 능력단위요소에 적절한 도구를 활용할 수 있는지를 점검
----------	---

- 지식 · 기술 · 태도

구 분	주 요 내 용
지식	<ul style="list-style-type: none"> • 반복, 점증 프로세스 • 비용산정 모델 • 소프트웨어 개발 방법론 • 소프트웨어 개발 표준 • 소프트웨어 개발 프레임워크에 대한 지식 • 소프트웨어 생명주기 모델 • 요구공학 방법론 • 프로젝트 관리 방법론
기술	<ul style="list-style-type: none"> • UML 작성기술 • 모델링 기법 • 모델링 도구 활용기술 • 유즈케이스 모델 정의할 수 있는 기술 • 의사결정 기술 • 프로세스 테일러링 기술 • 협상 기술 • 형상관리 기법
태도	<ul style="list-style-type: none"> • 문장 구성력 및 시각적 표현력을 갖추하고자 노력하는 태도 • 새로운 공학도구를 사용하고 하는 도전의식 • 소프트웨어 개발방법론을 적용하기 위해 프로젝트를 분석하려는 자세 • 소프트웨어의 품질에 대해 이해하고자 하는 태도 • 표준을 이해하고 준수하려는 의지

- 장비

장 비 명	단 위	활용구분(공용/전용)	1대당 활용인원
• 컴퓨터	대	공용	1
• 문서작성도구	세트	공용	1
• SW개발도구	세트	공용	1

※ 장비는 주장비만 제시한 것으로 그 외의 장비와 공구는 별도로 확보

- 재료

재 료 목 록
• 해당없음

※ 재료는 주제료만 제시한 것으로 그 외의 재료는 별도로 확보

Ⅲ. 고려사항

1. 활용방법

- 훈련기준에서 제시한 이외의 과정수립에 필요한 사항은 근로자직업능력개발법 등 관련 규정을 참고하시기 바랍니다.
- 본 훈련기준의 훈련과정은 모듈식으로, 장-단기과정 모두에서 활용가능하며, 훈련사업별로 요구하는 훈련과정 편성지침에 따라 편성할 수 있습니다.
- 3월 350시간 이상의 장기 훈련과정을 편성하는 경우, 수강생의 수준에 적합하게 훈련이수 체계도에서 제시한 해당직종의 훈련과정/과목을 필수로 반영하고, 이외 관련 직종의 과정/과목을 선택하여 편성할 수 있습니다.
 - * 단, 훈련생이 ‘필수과정’의 일부 훈련 과정/과목을 이수하거나, 직무수행경력이 있는 경우에는 해당 훈련과정/과목을 제외하고 훈련할 수 있습니다.
 - * 효율적으로 훈련하기 위해 둘 이상의 과정/과목을 결합하여 대(大)과목으로 편성하거나, 하나의 과정/과목을 둘 이상의 세(細)과목으로 편성하여 훈련할 수 있습니다.
 - * 훈련과정/과목에서 제시한 훈련시간은 훈련생의 학습능력을 고려하여 최대 50%까지 증감 조정하여 훈련할 수 있습니다.

2. 참고사항

가. 관련자격종목

- 정보관리기술사
- 컴퓨터시스템응용기술사
- 정보처리(산업)기사
- 정보처리기능사
- 멀티미디어콘텐츠제작전문가

나. 직업활동 영역

- 소프트웨어 개발업체
- IT컨설팅 회사
- IT교육기관
- 시스템 통합 및 구축 업체
- 기업전산실

다. 국가직무능력표준 관련 직종

- SW아키텍처
- 응용SW엔지니어링
- DB엔지니어링
- NW엔지니어링
- 보안엔지니어링
- UI/UX엔지니어링
- 임베디드SW엔지니어링

라. 관련 홈페이지 안내

○ 훈련기준 및 NCS : <http://www.ncs.go.kr>

○ 훈련정보 : <http://www.hrd.go.kr>

○ 자격정보 : <http://www.q-net.or.kr>