**4과목 프로그래밍 언어 활용**

1장 서버 프로그램 구현

127 개발 환경 구축

개발 환경 구축 – 응용 소프트웨어 개발을 위해 개발 프로젝트를 이해하고 소프트웨어 및 하드웨어 장비를 구축하는 것을 의미

하드웨어 환경 – 사용자와의 인터페이스 역할을 하는 클라이언트와 클라이언트와 통신하여 서비스를 제공하는 서버로 구성됨

서버 – 웹 서버 / 웹 애플리케이션 서버 / 데이터베이스 서버 / 파일 서버

웹 서버 기능 – HTTP/HTTPS 지원 / 통신 기록 / 정적 파일 관리 / 대역폭 제한 / 가상 호스팅 / 인증

소프트웨어 환경 – 클라이언트와 서버 운영을 위한 시스템 소프트웨어와 개발에 사용되는 개발 소프트웨어로 구성

시스템 소프트웨어 – 운영체제, 웹 서버, WAS 운용을 위한 서버 프로그램, DBMS

개발 소프트웨어 – 요구사항 관리 도구 / 설계/모델링 도구 / 구현 도구 / 빌드 도구 / 테스트 도구 / 형상 관리 도구

개발 언어 선정 기준 – 적정성 / 효율성 / 이식성 / 친밀성 / 범용성

128 서버 개발

- 웹 애플리케이션의 로직을 구현할 서버 프로그램을 제작하여 웹 애플리케이션 서버에 탑재하는 것을 의미

Java / JavaScropt / Python / PHP / Ruby

서버 개발 프레임워크 – 서버 프로그램 개발 시 다양한 네트워크 설정, 요청 및 응답 처리, 아키텍처 모델 구현 등을 손쉽게 처리할 수 있도록 클래스나 인터페이스를 제공하는 소프트웨어

Spring / Node.js / Django / Codeniter / Ruby on Rails

서버 프로그램 – 응용 소프트웨어와 동일하게 모듈 및 공통 모듈을 개발한 후 모듈들을 통합하는 방식으로 구현

모듈의 독립성은 결합도와 응집도에 의해 측정

독립성을 높이려면 모듈의 결합도를 약하게 하고 응집도를 강하게 하여 모듈의 크기를 작게 만들어야 함

129 보안 및 API

소프트웨어 개발 보안 - 소프트웨어 개발 과정에서 발생할 수 있는 보안 취약점을 최소화하여 보안 위협으로부터 안전한 소프트웨어를 개발하기 위한 일련의 보안 활동을 의미

데이터의 기밀성 / 무결성 / 가용성을 유지하는 것이 목표

소프트웨어 개발 보안 점검 항목

세션 통제 / 입력 데이터 검증 및 표현 / 보안 기능 / 시간 및 상태 / 에러처리 / 코드 오류 / 캡슐화 / API 오용

API – 응용 프로그램 개발 시 운영체제나 프로그래밍 언어 등에 있는 라이브러리를 이용할 수 있도록 규칙 등을 정의해 놓은 인터페이스

130 배치 프로그램

- 사용자와의 상호 작용 없이 여러 작업들을 미리 정해진 일련의 순서에 따라 일괄적으로 처리하는 것

정기 배치 / 이벤트성 배치 / On-Demand 배치

배치 프로그램이 갖추어야 하는 필수 요소 – 대용량 데이터 / 자동화 / 견고성 / 안정성/신뢰성 / 성능

배치 스케줄러 – 일괄 처리 작업이 설정된 주기에 맞춰 자동으로 수행되도록 지원해주는 도구

스프링 배치 – Job / Job Launcher / Step / Job Repository

Quartz – Scheduler / Job / JobDetail /Trigger

131 패키지 소프트웨어

- 기업에서 일반적으로 사용하는 여러 기능들을 통합하여 제공하는 소프트웨어를 의미

요구사항을 분석하여 업무 특성에 맞게 전용으로 개발되는 소프트웨어와 비교하여 안정성, 라이선스, 생산성 등에서 차이가 있음

문제정리

소프트웨어 환경 – 시스템 소프트웨어 / 개발 소프트웨어

가상 호스팅 – 하나의 서버로 여러 개의 도메인 이름을 연결하는 기능

대역폭 제한 – 네트워크 트래픽의 포화를 방지하기 위해 응답 속도를 제한하는 기능

개발 언어 선정 기준 – 적정성 / 효율성 / 이식성 / 친밀성

패키지 소프트웨어 – 결함이 발생했을 때 판매처의 프로세스에 따라 보안되므로 이용자의 사저엥 따라 즉각적이고 능동적인 대처가 어려움

전용 개발 소프트웨어 – 사업 환경에 맞추어 직접 개발한 소프트웨어

라이브러리 – 개발 편의를 위해 자주 사용되는 코드

통합 개발 환경 도구 – 개발에 필요한 편집기, 컴파일러, 디버거 등의 다양한 툴을 하나의 인터페이스로 제공하는 소프트웨어

소프트웨어 개발 보안 점검 항목 – 세션 통제 / 입력 데이터 검증 및 표현 / 보안 기능 / 시간 및 상테 / 에러처리 / 코드 오류 / 캡슐화 / API 오용

입력 데이터 검증 및 표현 – 입력 데이터에 대한 유효성 검증체계를 갖추고 검증 실패 시 이를 처리할 수 있도록 코딩하는 것

캡슐화 – 데이터와 데이터를 처리하는 함수를 하나의 객체로 묶어 코딩하는 것

스프링 배치 주요 구성 요소 – Job / JobLauncher / Step / JobRepository

Quartz – 오류가 발생하면 오류의 발생 위치, 시간 등을 추적할 수 있는 안정성, 신뢰성을 필수적으로 갖추고 있음

API – 응용 프로그램 개발 시 운영체제나 프로그래밍 언어 등에 있는 라이브러리를 이용할 수 있도록 규칙 등을 정의해 놓은 인터페이스

응용 프로그램 – 특정 업무를 처리하기 위해 만들어진 프로그램

ORM – 객체지향 프로그래밍의 객체와 관계형 데이터베이스의 데이터를 연결하는 기술

프로그래밍 언어의 선정 기준

친밀감 / 언어의 능력 / 처리의 효율성 / 프로그램 구조 / 프로그램 길이 / 이식성 / 과거의 개발 실적 / 알고리즘과 계산상의 난이도 / 자료 구조의 난이도 / 성능 고려 사항들 / 대상 업무의 성격 / 소프트웨어의 수행 환경 / 개발 담당자의 경험과 지식 / 사용자의 요구사항 / 컴파일러의 이용 가능성

키워드 : 1. 웹 서버 2. 웹 애플리케이션 서버 3. 개발 언어 선정 기준 4. 프레임워크 5. 소프트웨어 개발 보안 6. API 7. 배치 프로그램 8. 스프링 배치 9. Quartz 10. 패키지 소프트웨어