

UML:

프로그램의 구조/설계를 이해하기 쉽도록 도식화 할 수 있는 가시화되고 명세화된 언어

베이스라인:

형상 관리 활동에서 관리의 기준이 되는 지점

결함 추이 분석:

추후에 어떤 부분에서 결함이 발생할지 추정하는 기법

시스템 테스트:

분할된 개별 시스템을 통합한 시스템에서 오류가 없는지 테스트 하는 단계

IaaS(인프라형 서비스):

스토리지 같은 하드웨어를 가상화된 형태로 제공하는 서비스

프로토콜:

서로 다른 시스템이나 장치가 원활하게 통신하기 위한 규칙 등을 정해놓은 표준화된 규칙

모듈화:

프로그램을 하나의 기능을 가진 모듈로 나누어 설계하는 기법

정규화:

중복된 엔터티/관계/속성 등을 제거하여 이상현상을 방지하는 기법

제품 소프트웨어 패키징 도구:

완성된 제품 소프트웨어를 사용자가 사용할 수 있는 형태로 필요한 정보를 포함해 패키징 하는 도구

데이터 사전(DD):

데이터의 의미와 속성등을 취합하여 정리, 명세화한 문서

INDEX:

검색 능력의 향상을 위해 컬럼을 기반으로 만든 자료구조

링크 상태 알고리즘:

링크 상태를 기반으로 최적의 경로를 탐색하며 다익스트라 알고리즘을 사용하는 라우팅 알고리즘

TDD(Test Driven Development) :

작성할 프로그램에 대해 테스트를 먼저 수행한 뒤 테스트를 통과할 수 있도록 코드를 작성하는 원리

델파이 기법:

전문가의 경험과 지식에 기반해 문제를 식별하고 해결방안을 모색하는 기법

MVC 패턴:

대화형 애플리케이션을 모델, 뷰, 컨트롤러로 나누어 구조화한 패턴

디자인 패턴:

소프트웨어 설계 과정에서 자주 발생하는 문제를 해결할 수 있도록 자주 쓰이는 설계 방법을 정리한 패턴

사용성 테스트:

사용자가 직접 미리 준비된 시나리오를 따라 과제를 해결하고 질문에 응답하는 테스트

UI흐름설계:

업무 흐름이나 수행과 관련된 화면 흐름과 위치를 흐름도 형식으로 표현하는 활동

UX:

사용자가 프로그램 혹은 소프트웨어를 직/간접적으로 사용하며 느끼고 생각하는 총체적인 경험

SWOT분석:

기업 내/외부 환경을 분석해 강점, 약점, 기회, 위협 요인을 확인하여 기업 경영에 반영하는 전략

UML:

산출물을 명세화, 시각화, 문서화 할 때 사용되는 모델링 방법론과 기술을 통합한 표준화된 범용 모델링 언어

E-R모델:

현실세계에 존재하는 데이터와 데이터의 관계를 사람이 이해하기 쉽게 표현한 모델

ERD:

엔티티와 엔티티간의 관계를 도식화한 다이어그램

DB:

데이터를 동시에 여러 명의 사용자와 애플리케이션에서 사용할 수 있도록 통합되어 관리되는 데이터 집합

DBMS:

데이터의 삽입, 삭제, 수정, 갱신을 가능하게 하고 관리를 편리하게 해주는 소프트웨어

맵리듀스:

대용량의 데이터를 분산 병렬 컴퓨팅 환경에서 처리하기 위해 구글이 개발한 프레임워크

시스템 연계 방식 中 API:

응용 프로그램이 사용하기 위해 프로그래밍 언어나 운영체제에서 제공하는 기능을 제어할 수 있도록 규격을 정해놓은 인터페이스

REST:

웹과 같은 분산 하이퍼미디어 환경에서 자원의 존재/상태 정보를 표준화된 HTTP 메서드를 이용해 주고 받는 웹기술

IPSec:

3계층에서 AH, IKE, ESP 같은 프로토콜을 이용해 보안 서비스를 제공하는 터널링 프로토콜

스니핑:

직접적으로 피해자를 공격하지 않고 몰래 데이터만 엿보는 수동적인 공격기법

절차형 프로그래밍 언어:

컴퓨터에 저장된 명령어들을 절차적으로 수행하는 프로그래밍 방식을 사용하는 언어

함수형 언어:

수학적인 수식과 유사한 함수를 이용해 프로그래밍하는 언어

논리형 언어:

논리적인 문장을 이용해 프로그래밍 하고 계산을 수행하는 개념에 기반한 언어

패키지:

모듈을 디렉토리로 시켜 모아놓은 라이브러리

DB스키마:

자료형, 데이터 길이, 제약사항 등을 정의한 기본 구조

트랜잭션:

하나의 논리적인 기능을 수행하는 작업의 기본 단위

형상 통제:

형상통제위원회를 운영하고 변경 여부와 변경 활동을 통제하는 활동

루틴:

특정 동작을 수행하는 코드로 기능을 가진 명령들의 모임

모듈:

그 자체로 하나의 완전한 기능을 수행하는 독립된 실체

소프트웨어 형상 관리:

소프트웨어를 개발하는 전 과정에서 발생하는 모든 변경사항을 관리하는 활동

베이스라인:

산출물의 변화를 통제하는 시점의 기준

컴퓨터 과학에서의 추상화:

복잡한 자료, 시스템 등으로 부터 핵심 개념, 기능을 간추려내는 기법

배치프로그램:

사용자의 개입 없이 다수의 작업을 일괄처리 하거나 정기적으로 반복 실행하는 프로그램

DRDos:

공격자가 본인의 IP를 피해자의 IP로 위조하고 다수의 반사서버에 요청을 보내면 피해자는 반사 서버로부터 다량의 응답을 받고 가용성이 침해되는 공격

APR스푸핑:

공격자가 본인의 MAC주소를 다른 MAC주소로 위조해 지속적으로 패킷을 보내면 피해 서버는 해당 MAC주소를 공격자의 MAC 주소로 인지하고 해당 MAC 주소로 가는 패킷을 공격자가 스니핑 할 수 있는 공격

임의적 접근 통제(DAC)

사용자의 신분, 소속 그룹을 기반으로 접근을 통제하는 기법

VPN:

인터넷 같은 공중망 위에 가상의 사설망을 구축하여 전용망을 이용하는 것과 같은 효과를 가지는 기술

BIA:

재난/재해로 인해 중대한 손실이 발생한다고 가정하여 시간의 흐름에 따른 영향도 조사, 손실 평가를 하는 비즈니스 영향 분석

보안에서의 무결성:

인가된 방법으로만 데이터가 변경될 수 있으며 인가되지 않은 방법으로는 데이터가 변조/훼손 될 수 없음을 보장한다.

SYN 플러딩:

동시 가용 사용자 수를 SYN 패킷만을 다수 보내 점유함으로서 다른 사용자가 서버를 이용할 수 없게 만드는 공격

RUDY(Slow HTTP POST Dos):

요청 헤더의 Content-length를 매우 크게 설정하고 바디를 매우 소량씩만 보내 연결을 지속함으로써 가용성을 침해하는 공격

버퍼 오버플로우 공격:

메모리에 할당된 버퍼의 크기를 초과하는 데이터를 보내 프로세스 흐름을 바꾸어 공격자가 의도한 코드가 실행되도록 하는 공격

강제적 접근 통제(MAC):

사용자가 가진 접근 허가 권한과 정보의 허용 등급 근거해 접근을 통제하는 기법

SSL/TLS:

4계층과 7계층에서 서버와 클라이언트 간 상호 인증, 암호화, 무결성 보장하는 프로토콜

상태전이테스트:

시스템/객체의 상태를 규정하고 이벤트에 의해 상태가 전이되는 경우의 수를 테스트하는 기법

결함 추이 분석:

결함 관리 측정 지표의 속성값을 분석하고 추후에 어떤 모듈/컴포넌트에서 결함이 발생할지 추정하는 기법

리팩토링:

외부 기능은 수정하지 않고 내부 구조/관계를 단순화해 유지보수성과 생산성을 높이는 기법

테스트 원리 中 테스트는 정황에 의존적:

테스트하는 소프트웨어의 특성/성격에 적합한 테스트를 실시해야한다는 원리

페어와이즈 테스트:

테스트하려는 값끼리 최소한 한번씩의 조합을 만들어 상대적으로 적은 양의 테스트 세트를 설계해 테스트하는 기법

회귀테스트:

오류의 수정 혹은 제거로 인해 새로이 유입된 오류가 없는지 테스트하는 일종의 반복 테스트 기법

워크스루:

회의 전에 검토 자료를 배포해 확인한 뒤 짧은 시간 회의를 진행해 문제를 식별하고 해결방안을 모색하는 기법

맥케이브 순환복잡도:

제어 흐름의 복잡한 정보를 정량적으로 표시하는 기법

분기 커버리지(=선택 커버리지, 결정 커버리지):

결정 포인트 내의 전체 조건식이 최소한 한번씩은 T/F가 되도록 하는 커버리지



시스템 테스트:

통합된 단위 시스템의 기능이 시스템 환경에서 정상적으로 수행되는지 테스트하는 단계

스파게티 코드:

정상적으로 작동하지만 소스 코드가 복잡하게 얽혀있어 사람이 이해하기는 매우 어려운 코드

운영체제(OS):

하드웨어를 편하게 사용할 수 있도록 인터페이스를 제공하는 소프트웨어

IaaS(인프라형 서비스):

스토리지 같은 하드웨어를 클라우드 형태로 제공하는 서비스

선점형 스케줄링:

하나의 프로세스가 CPU를 점유하고 있는 동안, 대기열에 우선순위가 더 높은 프로세스가 도착하면 사용하고 있는 자원을 반환 후 우선순위가 높은 프로세스를 실행하는 스케줄링 기법

네트워크:

원하는 수신자에게 원하는 데이터를 정확하게 전달할 수 있게 하는 기반 인프라

프로토콜:

서로 다른 시스템에 있는 개체간에 원활히 데이터를 주고 받기 위한 표준화된 통신 규약

유닉스:

연구시설이나 교육시설에서 주로 이용하는 범용 다중 사용자 방식의 시분할 운영체제

클라우드 컴퓨팅:

인터넷을 통해 가상화된 IT 자원을 제공 받아 데이터를 본인의 컴퓨터가 아닌 인터넷으로 연결된 다른 컴퓨터로 처리하는 기술

네트워크 프로토콜:

원거리에 있는 장치 간에 원활히 메시지를 주고 받기 위한 양식/규칙

온프레미스 방식:

외부 인터넷망을 차단한 상태에서 내부 인트라넷망만을 사용하여 개발 환경 인프라를 구성하는 방식

릴리즈 노트:

사용자에게 개발 과정에서 정리된 제품의 릴리즈 정보를 제공하기 위한 문서

모듈화:

모듈을 이용하여 성능 향상, 시스템 디버깅, 수정이 용이하도록 하는 모듈 중심의 설계 기법

클리어링 하우스:

유통업자와 소비자 간의 거래에서 디지털 저작권 라이선스를 중개하고 발급하는 정산소

SSO:

한 번의 인증으로 재인증 절차 없이 다양한 시스템에 접근할 수 있는 통합 로그인 기술

데이터 마이닝:

다량의 데이터 간의 의미있는 규칙이나 패턴을 자동적/체계적으로 찾아내는 기술

비정규화:

정규화된 엔티티/속성/관계를 성능 향상, 관리 효율성 증대를 목적으로 통합/분할/중복 시키는 기법

ROLLBACK:

트랜잭션 수행중 오류가 발생했을 때 오류가 발생하기 전인 특정 시점으로 복구하는 명령어

SQL삽입:

악의적인 SQL을 삽입/실행시켜 DB에 접근하여 정보를 탈취하거나 변조시키는 공격

정규화:

중복된 데이터를 최소화하여 이상현상을 방지하는 기법

로킹기법:

일관성 유지를 위해 하나의 트랜잭션이 액세스 중인 데이터를 다른 트랜잭션이 액세스하지 못하게 제어하는 기법

병행제어:

다수의 트랜잭션이 동시에 실행될 때 일관성 유지를 위해 트랜잭션간의 상호작용을 제어하는 기법

테일러링:

프로젝트의 필요와 특성에 맞게 소프트웨어 개발과정, 기법, 산출물 등을 요구에 맞게 최적화하는 과정 및 방법론

시맨틱 웹:

온톨로지 형태로 표현된 관계-의미 정보를 컴퓨터가 처리할 수 있도록 하는 지능형 웹

온톨로지:

현실 세계에 존재하는 개념, 개념들간의 관계, 개념의 속성을 컴퓨터가 이해할 수 있는 형태로 표현한다

파티셔닝:

보다 작은 단위인 파티션으로 분할해 효율성을 증대시키고 성능을 향상시키는 기법

테스트 오라클:

테스트 결과의 참/거짓을 확인하기 위해 사전에 정의된 참값을 대입하여 비교하는 기법

동등 분할 테스트:

입력 데이터를 비슷한 도메인별로 유효값/무효값으로 그룹핑하여 대푯값을 도출해 테스트케이스를 설계하고 테스트하는 기법

시스템 카탈로그:

DB에 저장된 모든 개체들에 대한 정보가 담겨있으며 DBMS가 자체적으로 생성/관리 하는 테이블의 집합체

스머프 공격(=스머핑):

공격자 IP 주소를 피해자 IP주소로 위조하여 네트워크 전체에 ICMP echo 패킷을 직접 브로드캐스팅해 타겟 시스템을 마비 시키는 공격

XSS:

공격자가 악의적인 스크립트를 취약한 웹 사이트에 삽입해두면 해당 페이지에 피해자가 접근했을 때 스크립트가 실행되면서 피해자의 정보를 탈취해가는 공격 기법

CSRF:

취약한 웹사이트에 악의적인 스크립트를 삽입하여 해당 페이지에 사용자가 접근했을 때 사용자의 의지와는 관계 없이 공격자가 의도한 행위를 특정 웹사이트에 요청하는 공격 기법

S-HTTP:

네트워크 트래픽을 암호화하는 주요 기술로 서버와 클라이언트 간의 메시지 전체를 암호화 하여 전송한다

타임스탬프 순서:

트랜잭션에 타임스탬프를 부여해 미리 순서를 선택하고 부여된 타임스탬프를 근거로 해 트랜잭션을 수행하는 기법

낙관적 검증:

트랜잭션 수행 중 어떠한 검증도 하지 않고 트랜잭션이 완료되면 일괄 검증을 수행해 데이터베이스에 반영하는 기법

제품 소프트웨어 패키징 도구

배포를 위한 패키징 시에 디지털 콘텐츠의 지적 재산을 보호하고 관리하는 기능을 제공하며 안전한 유통과 배포를 보장하는 도구

SOAP:

SMTP, HTTP, HTTPS등을 이용하여 네트워크 상태에서 XML기반의 메시지를 주고 받는 프로토콜

옵티마이저:

사용자가 작성한 SQL문의 최적의 실행 계획을 탐색하고 수립하는 DBMS 내부 핵심 엔진

Secure OS:

운영체제의 보안 약점으로 인해 발생할 수 있는 공격을 예방하고자 커널에 보안 기능과 참조 모니터를 추가한 OS

NFV:

네트워크 기능을 가상 기능으로 모듈화 하여 필요한 곳에 제공하는 네트워크 가상화 기술

SDN:

오픈 API 기반으로 데이터 플레인과 컨트롤 플레인을 분리하여 네트워크 트래픽을 중앙 집중적으로 관리하는 기술

BCP(Business Continuity Planning):

재난/재해시에 업무 복구, 업무 재개, 비상 계획 등을 통해 비즈니스 연속성을 보장하기 위한 계획

RTO(Recovery Time Object):

업무 중단 시점부터 업무 복구까지 걸리는 시간

RPO(Recovery Point Object):

데이터를 복구해 다시 정상 가동 시킬 때 데이터 손실 허용 시점

DRS(Disaster Recovery System):

원활한 재해 복구 계획의 수행을 위해 평상시 모아놓은 자원과 관리체계를 통합한 재해 복구 센터

데이터 흐름도(DFD):

데이터가 프로세스의 흐름을 따라 어떻게 변화하는지 표현한 다이어그램

데이터 사전(DD):

자료 요소, 자료요소들의 집합, 자료 흐름등을 구체적으로 명시한 사전

HIPO(Hierarchy Input Process Output):

시스템 설계, 분석, 문서화에 사용되며 하향식 소프트웨어 개발에 사용되는 문서화 도구

고가용성(HA):

시스템이 장애에 대응해 오랫동안 지속적으로 정상 운영이 가능한 성질

tpmC:

1분당 최대 처리 횟수로 하드웨어 성능을 나타내는 객관적 지표

단일 책임의 원칙:

하나의 클래스는 하나의 책임만을 수행한다는 원칙

CRUD 매트릭스:

프로세스와 데이터 간의 관계 의존성을 Create, Read, Update, Delete 관점에서 표현한 표

INDEX:

검색 능력의 향상을 위해 컬럼에 대한 정보를 구성한 자료구조

소프트웨어 아키텍처:

소프트웨어의 구성요소, 외부로 드러난 특징, 구성요소간의 관계를 표현한 구조

IGMP:

멀티캐스트 그룹 멤버십을 구성하는 3계층 프로토콜

슈퍼키:

모든 튜플에 대해 유일성은 만족하나 최소성은 만족하지 못하는 키

관계 대수:

필요한 데이터와 데이터를 어떻게 유도해낼 것인가를 기술한 절차적 언어

그룹함수:

테이블의 모든 행을 하나 이상의 컬럼을 기준으로 값에 따라 그룹화 하여 그룹별 결과를 출력하는 함수

LoC:

각 기능의 원시 코드 수를 중간치, 비관치, 낙관치를 이용해 예측치를 구하여 비용을 산정하는 모형

은행가 알고리즘(Banker's Algorithm):

필요한 자원의 수를 미리 제시해 안정 상태일 때만 프로세스에 자원을 할당하는 교착 상태 회피 기법

링크 상태 알고리즘:

링크 상태 정보를 모든 라우터에게 전달해 최적의 경로를 구성하며 다익스트라 알고리즘을 사용하는 라우팅 프로토콜 알고리즘

와이어 프레임:

이해관계자와 UI 화면 구성을 협의하고 간략한 서비스의 흐름을 공유하기 위한 화면 단위 레이아웃 설계 작업



드라이브 바이 다운로드:

공격자가 불특정 웹사이트나 웹서버에 악성 스크립트를 설치하면 불특정 사용자가 접속시 동의 없이 프로그램이 실행돼 의도된 멀웨어 서버에 연결되어 사용자를 감염시키는 공격 기법

멀티캐스트 프로토콜:

동일한 데이터를 특정 그룹 내에 있는 다수의 사용자에게 동시에 전달할 수 있는 프로토콜

제어 흐름 테스트:

프로그램의 제어 구조를 그래프로 표현하여 프로그램 내부 구조와 동작을 테스트하는 기법

UDDI:

WSDL을 등록하고 검색할 수 있는 저장소로 공개적으로 접근/검색이 가능한 레지스트리이자 표준

애자일 방법론:

절차보다 사람이 중심이 되어 변화에 신속/유연하게 대응할 수 있는 신속 적응적 경량 개발 방법론

유스케이스 다이어그램:

프로그램 기능 및 외부 요소를 액터, 유스케이스, 시스템을 이용해 사용자 관점에서 표현한 다이어그램

방화벽:

미리 정의된 규칙에 근거하여 불법적인 외부 침입과 내부 정보 유출을 차단하는 보안 시스템

스레드:

프로세스의 실행/제어 부분만을 가져온 작은 실행 단위로 독립적으로 수행된다.

객체지향에서의 상속:

상위 클래스의 메서드와 필드를 하위 클래스에서도 사용할 수 있도록 하는 기법

객체지향에서의 다형성:

하위 클래스 타입이 상위 클래스 타입으로 변환될 수 있다는 개념

브로드캐스트 프로토콜:

하나의 송신자가 같은 서브 네트워크에 속한 모든 수신자에게 동시에 동일한 데이터를 전송할 수 있는 프로토콜

스토리보드:

UI화면 설계를 위해 와이어프레임 등 구축하는 서비스를 위한 대부분의 정보를 담은 문서

트로이 목마:

겉으로 보기엔 정상적인 프로그램으로 보이지만 악성코드가 숨어있어 실행하면 악성 코드를 실행 시킨다.

DHCP:

IP주소와 설정 정보를 동적으로 할당해주는 프로토콜

Dos공격:

피해 서버에 다수의 접속 시도를 통해 자원을 고갈시켜 정상적인 사용자가 서비스를 원활히 사용할 수 없도록 하는 공격

JSON:

키-값 쌍, 속성-값 쌍으로 이루어진 데이터 오브젝트를 전달하기 위해 사람이 읽을 수 있는 텍스트를 사용한 개방형 표준 포맷

자료사전의 목적:

특정 자료 용어의 의미가 무엇인지 알려주기 위해 데이터의 정의를 취합/정리한 사전이다.

개체 정의서:

개념 모델링 단계에서 도출한 개체의 속성, 타입 등의 정보를 개괄적으로 명세화한 정의서

테이블 정의서:

논리 및 물리 모델링 단계에서 테이블의 컬럼, 자료 타입등을 명세화한 정의서

EAI:

기업 내의 서로 다른 애플리케이션 간 정보를 연계/통합/전달할 수 있게 하는 솔루션

ESB:

미들웨어를 중심으로 서로 다른 프로토콜을 사용하는 애플리케이션이 느슨한 결합을 통해 연계될 수 있게 하는 기술

XML:

SGML과 HTML의 단점을 보완한 특수 목적의 마크업 언어

AJAX:

자바 스크립트를 이용해 웹서버와 클라이언트 간에 비동기적으로 XML 데이터를 교환/조작하기 위한 기술

DB암호화 방식 中 TDE 방식:

DBMS 커널에서 자체적으로 내장된 암호화 기능을 수행하는 방식

인터페이스 구현 검증 도구 中 Selenium:

다양한 개발언어와 브라우저를 지원하며 플레이백 도구를 지원하는 인터페이스 구현 검증 도구

프로시저:

일련의 SQL문을 하나의 함수처럼 사용하기 위한 절차형 SQL

사용자 정의 함수:

일련의 SQL을 절차적으로 수행하고 단일 결과값을 반환하는 절차형 SQL

트리거

데이터의 삽입, 삭제, 갱신등의 이벤트가 발생하면 관련 작업이 자동으로 실행되는 절차형 SQL

힌트:

SQL문을 실행하기 전에 미리 정보를 알려주고 효율적인 수행을 가능하게 한다.

MyBatis:

JAVA를 이용한 관계형 데이터베이스 프로그래밍을 쉽게 해주는 프레임워크

메타데이터:

데이터 타입, 데이터 구조 등 데이터와 관련된 정보를 나타낸 상위 데이터.

FP방식:

개발을 어렵게 만드는 요소에 가중치를 부여하고 해당 가중치를 기반으로 비용을 산정하는 방식

코드 난독화:

역공학을 이용한 공격에 대비하여 코드를 알아보기 어렵게 바꾸는 기술

디지털 포렌식:

사법기관에 제출하기 위해서 디지털 증거 자료를 수집/기록/보관 등을 하는 일련의 활동

IPv6:

IPv4의 주소 고갈, 이동성 및 보안 기능 결여 등을 보완하고 128bit주소 체계를 사용하는 IPv4를 대체하기 위한 차세대 인터넷 프로토콜

데이터베이스의 고립화 수준:

다른 트랜잭션이 현재 데이터에 대한 무결성을 해치지 않기 위해 잠금을 설정하는 정도

ESP:

IPSec에서 암호화 알고리즘을 이용해 캡슐화 기반 페이로드 기밀성을 제공하는 프로토콜

NAT(Network Address Translation):

사설 네트워크에 속한 ip 주소를 공인 ip 주소로 변경하여 주는 네트워크 주소 변환 기술