Agile Software Development

- 애자일 소프트웨어 디벨롭먼트
- 신속하고 변화에 유연하며 적응적인 소프트웨어 개발을 목표로 하는 다양한 경량 개발 방법론 전체를 일컫는 총칭
- 개발 대상을 다수의 작은 기능으로 분할하여 하나의 기능을 하나의 반복 주기 내에 개발하는 개발 방법
- 하나의 반복 기간을 프로젝트마다 다르지만 일반적으로 1주에서 4주 정도인 경우가 많다. 이 반복 주기를 계속해 나가면 하나씩 기능을 추가 개발하는 것이다.
- ref 애자일 개발방법론

Boilerplate code

- boilerplate code(보일러 플레이트) 또는 boilerplate(보일러플레이트)는 변경하는 것 없이 많은 곳에서 포함되어야만 하는 코드의 부분을 의미한다.

클라우드 네이티브 [Cloud Native]

- 단일 처리에 특화된 최소한의 서비스 단위로 개발을 하고, 이를 컨테이너 환경에서 실행하는 애플리케이션 개발 방식
- 수많은 컨테이너가 오케스트레이션을 통해 자동으로 관리, 운영하는 등 클라우드 네이티브는 클라우드 컴퓨팅 모델의 이점을 최대한 살린 개발방식

Dead Link

- **깨진 링크(broken link)** 또는 **죽은 링크(dead link)**는 영구적으로 이용할 수 없는 웹 페이지나 서버를 가리키고 있는 월드 와이드 웹 상의 링크를 말한다. 깨진 링크에 들어갈 때의 가장 일반적인 결과는 바로 웹 서버가 응답하지만 특정한 페이지를 찾을 수 없을 때에 나타나는 404오류이다. 브라우저는 웹 서버가 도메인 이름에서 찾을 수 없음을 가리키는 DNS오류를 반환하기도 한다ㅏ. 또, 콘텐츠 필터나 방화벽과 같은 차단 방식의 까닭으로 링크가 깨졌다고 보고하기도 한다
- 출처 깨진 링크

Docker

- Container-based Virtualization System [컨테이너 베이스드 버츄얼리제이션 시스템]
- 컨테이너 기반의 가상화 시스템

IEEE 754

- IEEE에서 개발한 컴퓨터에서 부동소수점을 표현하는 가장 널리 쓰이는 표준이다. $+0$, -0 등의 수와 무한(Infinity), NaN (Not a Number)등의 기호를 표시하는 법과 이러하나 수에 대한 연산을 정의하고 있다.

log [로그, 운용 기록]

1. 기계 작동에 관한 기록, 기계 상황의 변경, 스위치 선택, 입출력 장치에 대한 사항, 제어 탁자에서 입력한 자료, 기계 정지 시의 상태나 원인 등에 관한 사항의 기록 등 컴퓨터 운용에 관계되는 모든 기록, 컴퓨터에 있는 로그 파일에 자동적으로 기록되게 할 수 있다.
2. 어떤 트랜잭션이 데이터베이스의 어떤 값을 갱신할 때 갱신 전의 값과 후의 값을 로그에 먼저 기록하는 것. 트랜잭션의 시작과 완료, 철회 등의 정보도 로그에 기록된다.

logging [로깅]

컴퓨터 동적 상태에 따른 변화를 시간에 따라 기록하는 것

Interpreter [인터프리터]

파이썬 코드를 입력한 후 엔터 키를 누르면 결과가 바로 나온다. 이처럼 코드를 한 줄 한 줄 실행하여 결과를 얻는 방식을 인터프리터(interpreter)방식이라고 부른다.

meme

- 인터넷 밈 (Internet Meme) 또는 줄여서 밈 (Meme)
- 인터넷에서 유행하는 특정 콘텐츠와 문화 요소를 이르는 망

Proxy Server [프록시 서버]

- 시스템에 방화벽을 가지고 있는 경우 외부와의 통신을 위해 만들어 놓은 서버
- 프록시 (Proxy)란 원래 '대리'라는 의미를 가지고 있음
- 클라이언트가 프록시 서버를 통해서 다른 네트워크 서비스에 간접적으로 접속할 수 있게 해주는 컴퓨터
- 시스템에 방화벽을 가지고 있는 경우 외부와의 통신을위해 만들어 놓은 서버
- 프록시 서버는 프록시 서버에 요청된 내용들을 캐시를 이용하여 저장
- 프록시 서버 (Proxy Server)

pseudo-code[수도코드]

- 의사(疑 비교할 의 / 似 비슷할 사 | Pseudo : 가짜의)
- 컴퓨터 프로그램이나 알고리즘이 수행해야 할 내용을 우리가 사용하는 언어(한국어 또는 영어 등)로 간략히 서술해 놓은 것을 말한다.
- 특정 프로그래밍 언어로 구현하기 이전에 일반 언어로 작성한 간단한 버전의 프로그래밍 코드

QD, Quantum Dot [퀀텀 닷]

- 양자점[量子點]은 wlfmadl 2~10nm(나노미터: 10억분의 1m의 길이]에 불과한 반도체 입자로 특이한 전기적, 광학적 성질을 지닌 입자
- 퀀텀닷 기본원리

리팩터링 [refactoring]

- 결과 변경 없이 코드 구조를 재조정하는 방법.
- 버그를 없애거나 새로운 기능을 추가하는 것이 아니라 코드 가독성과 유지보수성을 높이는 행위를 말한다.

Sandbox [샌드박스]

- 보호된 영역 안에서 프로그램을 작동시키는 보안 소프트웨어
- ITWorld 용어풀이 | 샌드박스(SandBox)

SOLID 원칙

- 로버트 마틴이 정립한 객체 지향 프로그래밍 및 설계의 다섯 가지 원칙을 마이클 페더스가 새로 정리한 원칙

1. S : SRP [Single Responsible Principle, 단일 책임 원칙]
    - 한 클래스는 하나의 책임만 가져야 한다.
2. O : OCP [Open/Closed Principle, 개방-폐쇄 원칙]
    - 소프트웨어 요소는 확장에는 열려 있으나 변경에는 닫혀 있어야 한다.
3. L : LSP [Liskov-substitution Principle, 리스코프 치환 원칙]
    - 객체는 프로그램의 정확성을 깨뜨리지 않으면서 하위 타입의 인스턴스로 바꿀 수 있어야 한다.
4. I : ISP [Interface Segregation Principle, 인터페이스 분리 원칙]
    - 특정 클라이언트를 위한 인터페이스 여러 개가 범용 인터페이스 하나보다 낫다.
5. D : DIP [Dependency Inversion Principle, 의존관계 역전 원칙]
    - “ 추상화에 의존해야지 구체화에 의존하면 안 된다 ” 는 원칙을 따르는 방법의 하나

tokenizer [토크나이저]

- tokenization [토크너제이션] : 토큰화
- 문자열을 여러 개의 조각, 즉 여러 개의 Token(토큰)들로 쪼개는 것을 말한다.
- Token은 문자열의 한 조각으로 하나의 단어가 하나의 토큰이라고 할 수 있다.

Use Case

- 유즈 케이스
- 시스템의 동작을 사용자의 입장에서 표현한 시나리오이며, 시스템에 관련한 요구사항을 알아내는 과정이며, 소프트웨어 개발 프로세스 중 개발을 위한 소프트웨어의 기능을 개략적으로 설명이 가능한 방법이다.
- 유즈 케이스는 시스템을 분석하는 사람과 사용할 사람이 함께 시스템의 사용 방법을 결정하는데에 도움을 준다. 작성 시 주의할 점은, 시스템을 개발하는 사람보다는 사용자의 입장에서 이해하기 쉽고, 간단하게 작성되어야 한다.
- ref 유즈케이스(Use Case)에 대하여

Wireframe

- 와이어 프레임
- 레이아웃과 UI 요소들에 대한 뼈대를 제공하는 그림
- PENCIL PROJECT

오션, IT, 용어

From:
http://rwiki.repia.com/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://rwiki.repia.com/doku.php?id=wiki:user:emblim98:terms&rev=1611618483

Last update: **2022/03/10 19:52**