

테크놀로지 리더를 위한

Media & IT(Information Technology)

강자원

컴퓨터시스템응용기술사

/ KBS MNC(Media Network Center)팀

#1. 소프트웨어공학과 미디어



들어가며

필자가 방송국 입사를 준비할 때(2004년) 만해도 유비쿼터스(Ubiquitos)가 핵심 키워드였다. 유비쿼터스(Ubiquitous)란 ‘어디에나 있는’, ‘도처에 존재하는’이라는 뜻으로, 정보통신에서는 ‘사용자가 시간과 장소에 구애받지 않고 자유롭게 컴퓨터에 접속할 수 있는 네트워크 환경’을 의미한다. 그렇게 개념만으로 존재했던 기술의 실체가 이제는 모든 것이 소셜과 오픈으로 통하는 시장이 되었다. 그러나 모든 진화에는 사이클이 있어서 그 원동력을 근본적으로 이해하지 못하면 앞으로의 변화를 미리 준비하기 어렵다. 특히 인터넷 공간과 같이 사용자들에 의해 움직이는 변화무쌍한 미디어라면 더욱 그렇다. 이제는 방송이라는 표현보다는 미디어라는 표현을 더 많이 포괄적으로 사용하고 있다. 그 차이부터 이야기해보자. 방송과 미디어의 사전적 정의를 찾아보면 다음과 같다.

방송(放送, Broadcasting)은 다수를 상대로 하여 정보를 전달하는 매체

미디어(Media)는 정보를 주고받는 것, 한마디로 ‘매체’라는 의미로 대개는 신문, 텔레비전, 영화 등의 대중매체(정확히는 매스미디어)를 일컫는 영어 단어

일방적인 전달이 아니라 정보를 주고받는 형태가 미디어라는 것이다. 방송이 이렇게 양방향인 다양한 미디어로 진화함에 따라 콘텐츠 생산의 형태와 방식 또한 변화하고 있다. IT(Information

Technology)는 방송이 미디어로 변화하는 그 흐름 속에 필수불가결한 기술요소 및 핵심이 되었다. 먼저, 나의 IT 상식에 대해 자가진단을 해보자. 다음의 10문제 중 나는 몇 문제나 답할 수 있는가?

나의 IT 상식



스스로 체크해보세요!
나의 IT능력 자가체크!

1. 리눅스에서 할당받은 IP주소를 확인할 때 사용하는 명령어
2. 다른 특수한 목적을 갖는 마크업 언어를 만드는 데 사용하도록 권장하는 다목적 마크업 언어
3. 기존 소프트웨어를 버리지않고 기능을 개선시키거나 새로운 소프트웨어로 재활용 하는 기법
4. 소프트웨어의 개발과 운영의 합성어로 소프트웨어 개발자와 정보기술 전문가 간의 소통, 협업 및 통합을 강조하는 개발환경을 의미하는 명칭
5. 둘 이상의 프로세스가 서로 상대방의 프로세스가 끝나기만을 기다리고 있기 때문에 결과적으로 아무것도 완료되지 못하는 상태
6. 신뢰할 수 있는 발신자가 보낸 것으로 가장하는 표적화 된 이메일 공격기법
7. 리눅스 가상화 컨테이너로 사용자의 컴퓨터 안에서 가상의 리눅스 컴퓨터가 구동될 수 있도록 해주는 기술
8. 소형 네트워크들을 묶어 대규모 파이프라인을 통해 극도로 높은 대역폭으로 다른 네트워크들의 집합과 연결되는 네트워크
9. 보통 이중화나 클러스터링이 구성되어 있는 장비에서 사용되며 내가 살아있거나 상대가 살아있는 것을 점검하는 메세지
10. 접속된 서버의 대수를 늘려 처리 능력을 향상시키는 것으로 하나의 장비에서 처리하던 일을 여러 장비로 나누어 처리할 수 있게 설계를 변경하는 것

<정답은 맨 뒤에...>

각 IT 영역에서의 미디어 연재

위 문제들은 실제로 미디어와 관련된 IT 프로젝트를 진행하는 현장에서 아주 빈번하게 사용하고 있는 기술용어들이다. 아는 것보다 모르는 것이 많다고 걱정하지 말라! 이제부터 방송기술인들이 테크놀로지 리더로 성장할 수 있도록 분야별로 미디어와 관련된 IT 기술에 대해 설명하고자 한다. 이제부터 필자는 아래와 같은 주제로 매월 총 12회에 걸쳐 공학적인 관점에서 각 IT 영역에서의 미디어를 분석하고 들여다보고 소개하려 한다.

- | | | |
|-----------------|------------------|---------------|
| 1. 소프트웨어공학과 미디어 | 5. 소프트웨어개발과 미디어 | 9. 클라우드와 미디어 |
| 2. 네트워크와 미디어 | 6. 애플리케이션과 미디어 | 10. 인공지능과 미디어 |
| 3. 보안과 미디어 | 7. 시스템 아키텍처와 미디어 | 11. 블록체인과 미디어 |
| 4. 데이터와 미디어 | 8. IT 운영과 미디어 | 12. 가상현실과 미디어 |

이번 호에서는 소프트웨어공학이 무엇이며, 이는 미디어와 어떤 관계가 있는지 알아보려 한다.

소프트웨어 공학이란

소프트웨어공학이란 소프트웨어의 개발, 운용, 유지보수 등의 생명 주기 전반을 체계적이고 서술적이며 정량적으로 다루는 학문이다. 즉, 공학을 소프트웨어에 적용하는 것이다. 지금까지 우리가 시스템을 구축할 때 알게 모르게 진행해왔던 프로세스에는 소프트웨어공학적 접근이 모두 녹아있었다. 콘텐츠를 생산하는 과정이 디지털융합과 더불어 IT 프로세스화 됨에 따라 소프트웨어를 개발하는 일뿐만 아니라 장비구매 및 시스템의 운영, 유지보수의 단계까지 공학적 접근과 프로세스가 이미 많은 부분 적용되어 있다. 다만 우리가 몰랐을 뿐이다.



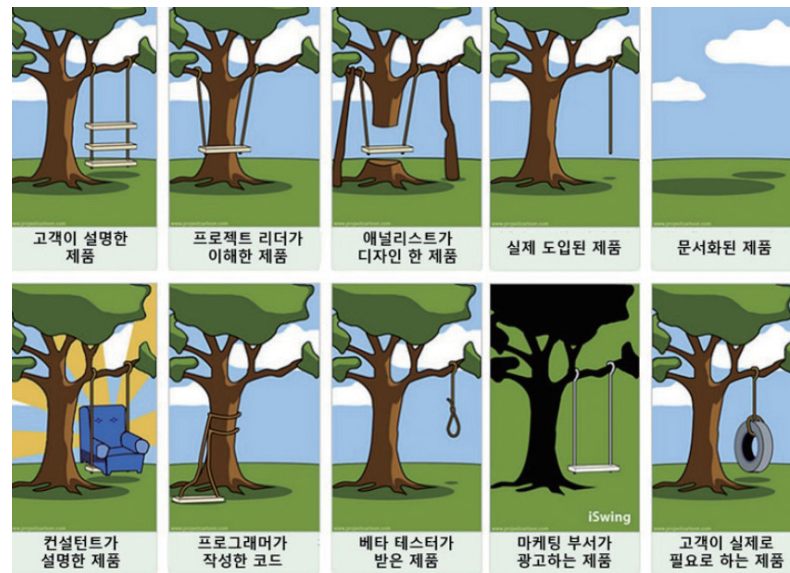
소프트웨어공학의 세부 분야

우리 각자 자사의 미디어 콘텐츠를 생산, 및 저장, 유통하는 데에 사용되는 IT 기반의 시스템 구축과정을 떠올려보자. IT 기술과 소프트웨어공학적인 마인드 셋이 없는 리더는 ‘소프트웨어 공학’ 따위의 ‘장난’은 실제 개발 과정에 필요 없다고 역설하고 이것을 당연하게 여긴다. 보통, 이렇게 만들어지는 소프트웨어의 품질은 보장할 수 없고, 이 보장할 수 없는 소프트웨어를 통해서 꾸준한 유지보수 비용과 일거리가 발생하는 사실은 아마도 ‘4대 강’처럼 한번 만들어 두면 끊임없는 유지보수 업무를 발생시키는 식의 문제 정의와 처리 방법이라고 할 수 있겠다. 당연한 것이지만, 결론적으로 이야기하자면, 시스템구축 및 개발 총괄 책임자의 대우는 형편없고 일정 조절이나 개발에 대해서 지휘할 수 있는 권리나 인사권 같은 것도 매우 부족한 상황으로 변화한다. 그래서 이런 조직일수록 소프트웨어공학은 그냥, 뜬구름 잡는 이야기가 되는 경우가 다반사이다. 일반적으로 회사에서는 기획과 경영, 회계와 관리에 해당하는 업무들이 가장 중요하므로 소프트웨어 개발의 시각화나 공정에 대해서는 그다지 관심이 없는 경우가 많다. 오히려, 제품을 선택하고 유지보수 업체를 어떻게 관리하고 운용할 것이냐에 핵심과 초점이 있기 때문에 소프트웨어 공학적인 배경은 가장 중요한 선택의 포인트가 되지 못한다.

만약, 계획과 절차 없이 소프트웨어를 개발할 경우 코딩과 수정(coding-and-fix) 과정이 반복된다. 즉흥적인 코딩과 수정이 반복되면 다음과 같은 문제에 노출된다.

- ▶ **개발 지연과 예산 초과** 계획에서 벗어난 일정은 요구사항이 명확하지 않으며 프로그램 골격이 확정되지 않아 계속 수정이 반복되므로 개발 지연과 예산을 초과하게 된다.
- ▶ **낮은 품질** 명확한 범위 산정과 일정 관리가 되지 않아 결국 소프트웨어의 품질이 저하된다.
- ▶ **유지보수 곤란** 복잡한 시스템을 구성하는 각 요소를 잘 구조화하고 관리하지 않으면 간단한 수정도 불가능하며 개발자가 변경되면 유지보수도 곤란해진다.
- ▶ **재작업** 구조화되지 않고 마구잡이로 개발한 프로그램은 수정할 수 없게 되어 재구조화는 등의 작업이 필요하고 이는 비용이 발생하게 된다.

시스템구축 및 개발 프로젝트에서 위와 같은 문제를 줄이기 위해서 가장 중요한 방법의 하나는 요구사항의 명확화이다. 프로젝트의 성패는 요구분석에 달려있다고 해도 과언이 아니다. 요구사항을 체계적으로 수집, 분석, 명세화, 검증하고 추적, 변경하는 요구사항 도출 및 관리 기법을 소프트웨어공학에서는 ‘요구공학(Requirements engineering)’이라 한다. 만약, 요구사항이 명확하지 않으면 각자의 이해관계자들은 아래 그림과 같이 동상이몽을 하고 있을 것이다. 또, 실제로 이러한 사례는 다반사로 일어난다. 그래서 프로젝트 기간 안에 개발이 완료되지 못하고 지연되는 등 실제로 프로젝트 및 소프트웨어의 품질을 저하하는 결과를 초래한다.



요구공학의 중요성을 설명하는 사례

요구공학이 중요한 이유

- 소프트웨어 품질을 판단하는 기초가 된다.
- 테스트 케이스를 만드는 기초가 된다.
- 이해관계자의 입장에 따라(운영자 / 사용자 / 개발자) 필요한 문서가 다를 수도 있다.
- 요구사항은 대개 자연어로 서술하는데 그러면 같은 얘기도 다르게 해석하는 경우가 있다.
- 기준 없이 여러 명이 요구사항을 작성한다면 추상화나 자세함의 정도가 다르므로 전혀 다른 아웃풋이 만들어질 수 있다.
- 시간이나 보안 등과 같이 놓치기 쉬운 비기능적인 요구사항을 파악하고 예외적인 이벤트에 대해 적절히 대응할 수 있어야 한다.

유튜브 및 SNS로 콘텐츠 생방송 서비스를 위한 스트리밍 시스템을 구축한다고 가정해보자. 아래와 같은 비기능 사항들은 이렇게 명시하지 않으면 프로젝트의 일정 지연 및 실패를 야기시킬 수 있다.

구분	요구사항	속성	상세
효율성 (Efficiency)	성능 (Performance)	빠른응답	모든 미디어 콘텐츠는 트랜스코딩이 완료되자마자 0.1초 내로 응답하여야 한다.
	자원효율성 (Resource Efficiency)	낮은 메모리점유율	프로세스가 진행되는 동안 메모리 점유율은 최대 60%를 넘지 않도록 설계되어야 한다.
기능성 (Functionality)	보안성 (Security)	고유성 (Unique)	스트리밍 서버는 허용된 장비(Key를 가지고 있는)에서만 트랜스코딩된 콘텐츠에 한하여 허용되어야 한다.
		Fault Tolerance	스트리밍 서비스는 이중화되어 서버가 장애가 발생했을 때 동기화되어 지연이 1ms 이내로 동작해야 한다.
		암호화통신	스트리밍 서버와 통신하는 서버의 모듈 간 메시지 통신방식은 256bit(SHA-2) 알고리즘으로 암호화되어야 한다.
		접근통제	스트리밍 서버에 저장된 콘텐츠목록 및 통신기록 등은 조회(열람) 시 허용된 사용자 및 IP만 가능해야 한다.
		인증 및 권한	등록되지 않은 서버 및 사용자 이외의 요청에 응답하지 않도록 보안 레벨을 설정할 수 있게 한다.
	안정성 (Safety)	트랜잭션처리	시스템을 보호하기 위해 처리하는 명령을 수행하는 동안에는 다른 명령은 수행할 수 없다.

스트리밍 서비스 구축 시 요구정의되어야 하는 비기능 요구사항 예시

마치며

미디어프로젝트 수행 시 공학적 접근에 대해 지면의 한계상 아주 간략하게만 소개를 했지만, 이 정도만 보더라도 얼마만큼 중요한지 알 수 있었을 것이다. 소프트웨어 유지보수에 드는 비용은 개발에 드는 비용의 두 배 이상이다. 개발과 구현 자체가 목표가 아니다. 우리가 무언가 새로운 솔루션을 도입하거나 시스템을 구축할 때 운영과 유지보수는 고려하지 않은 채 기능 구현과 초기 도입비용에만 초점이 맞춰져 마치 혁신적이고 비용절감인듯한 상술에 속지 않아야 한다. 테크놀로지 리더로 다가가기 위한 그 첫걸음을 소프트웨어공학에 대한 이야기로 시작했다. 방송기술인들의 IT에 대한 관점의 확장에 도움이 됐길 바란다. 📖



스스로 체크해보세요! 정답확인!!!

나의 IT능력 자가체크!

1. ifconfig
2. XML
3. 재공학
4. 데브옵스(DevOps)
5. 교착상태(Deadlock)
6. 스피어피싱(Spear phishing)
7. 도커(Docker)
8. 백본(Backbone)
9. Heartbeat
10. 스케일아웃(Scale-out)