

2024.07.13

HanBitN MSA Season 2 - 1

# 제로베이스에서 유능한 QA로 성장하는 소프트웨어 테스트의 모든 것

Keyword : Software Testing

발표자 : 남효진

# | 연사소개 : 남효진

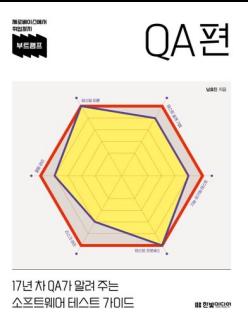


## 경력

- 전) 데브시스터즈 Platform QA & PM
- 전) 넷마블 Mobile QA
- 전) 위메프 QA Leader
- 전) NHNts QA

## 출판

<부트캠프 QA편> 저자



# | 강연소개

Session Title

## 제로베이스에서 유능한 QA로 성장하는 소프트웨어 테스트의 모든 것

Session Keyword

Software Testing

Session Additional Explanation

소프트웨어 테스팅 기본 개념과 테스트 기법을 이해하고,  
테스트가 수행되는 절차와 프로세스에서의 QA 역할과 활동 범위를 알아봅니다.  
QA에 대한 이해는 바탕으로 '보다 나은' QA가 되기 위한 업 스킬링Up-skilling과 QA의 미래까지,  
소프트웨어 테스팅의 기초부터 유능한 QA로 성장하기 위한, 실용적인 지식을 전달합니다.

# | 목차

1. 소프트웨어 테스팅 기본 개념
  1. 소프트웨어 테스팅 정의
  2. 소프트웨어 테스팅이 필요한 이유
  3. 소프트웨어 품질 범위
  4. 결함에 대한 정의 및 사례
2. 소프트웨어 테스팅 이해
  1. 테스트 단계
  2. 테스트 종류
  3. 테스트 수행 방식
  4. 테스트 케이스 정의 및 커버리지
  5. 테스트 설계 기법
3. 테스트 프로세스
  1. 소프트웨어 개발 프로세스 정의와 flow
  2. 프로세스
  3. QA 역할과 활동 범위
4. '보다 나은' QA가 되기 위한 방법
  1. QA의 요건과 자질
  2. Up-Skilling
  3. QA 산업의 미래
  4. QA를 필요로 하는 분야

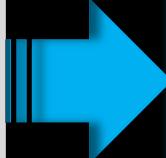
# 01

## 소프트웨어 테스팅 기본 개념

# | 소프트웨어 테스팅이란?

## 과거

제품 개발이 완료된 후  
소프트웨어를 실행하면서  
테스트를 수행하여  
제품이 잘 작동하는지 확인  
하고 결함을 찾아내는 것



## 현재

프로그램이나 시스템의 동작과 로직, 성능, 안정성이  
**기대하는 수준과 요구사항에 만족**하는지 확인  
하고, 제품의 내외부적 요인에서 발생하는 결함이 없  
는지 확인하는 활동을 통해 **제품의 품질에 대한  
확신을 증진시키는 과정**이다.

또한, 결함 **데이터를 근간**으로 개발 프로젝트의 **리  
스크**에 대한 **정량적인 판단 근거를 예측**하여 전  
달하고, 이슈가 유입되는 경로를 확인하여 **원인을  
제거하고 예방하고 방지**하는 활동이다.

# | 소프트웨어 테스팅 정의 예시

과거



# | 소프트웨어 테스팅 정의 예시

현재



사료선정

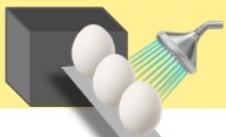
검수자의 역할과  
활동 범위



Start



① 농장에서 생산된 달걀



③ 세척, 살균 등 위생 처리



④ 크기별로 포장 및 출하



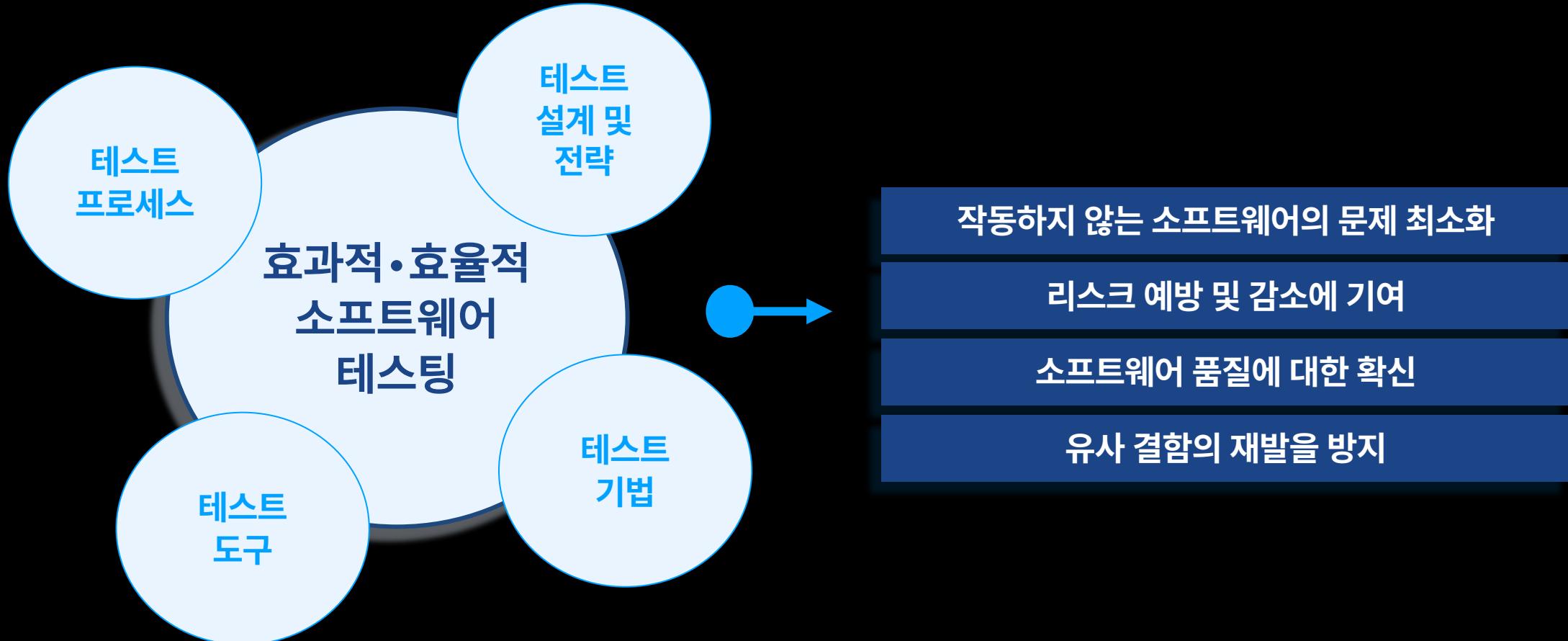
② 판매에 부적합 한  
달걀 선별



End

⑤ 소비자는 신선하고  
안전한 달걀 구매

# | 소프트웨어 테스팅이 필요한 이유



# | 소프트웨어 품질

- 고객의 기대와 주어진 요구사항을 만족하는 정도를 결정하는 소프트웨어의 기능과 특징
- 개발과정에서 표준(내부 구조, 소스 코드, 기술 수준 등)을 준수하는 소프트웨어의 능력과 속성

## 소프트웨어 품질

### 사용자 관점

품질 특성 중 기능성, 신뢰성, 안정성, 사용 용이성 등 **제품의 기능적 품질**에 해당되며 소프트웨어 테스팅 활동을 통해 품질이 측정된다.

- 고객의 기대와 요구사항에 충족하는 정도
- 제품의 내외부적 요인에서 발생하는 결함이 없어야 한다.
- 예상대로 작동되는 소프트웨어

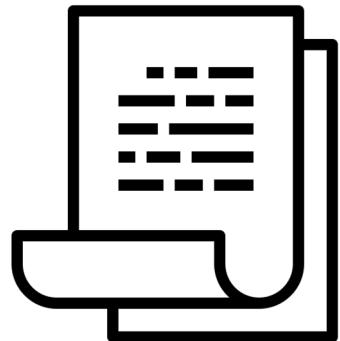
### 품질 관리자 관점

기능적 품질에 더하여 품질 특성 중 효율성, 유지보수성, 이식성 등 제품의 관리적 측면을 포함한 **문제를 예방하는 활동**

- 데이터를 활용하여 품질에 영향을 주는 요소와 리스크를 관리하고 제어하며 예방
- 제품을 개발하는 과정인 **프로세스에서 수행되는 작업의 품질을 관리 및 제어**
- 전반적인 테스팅 활동의 품질

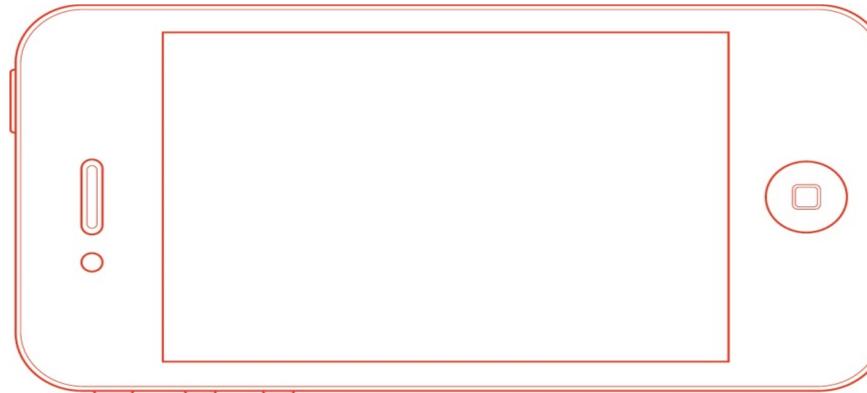
# | 소프트웨어 품질

산출물 품질



기획서/  
고객 요구사항/  
개발설계서/  
테스트 산출물  
etc.

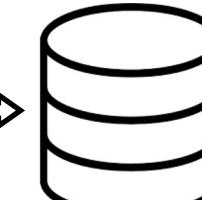
프로덕트 품질



Code

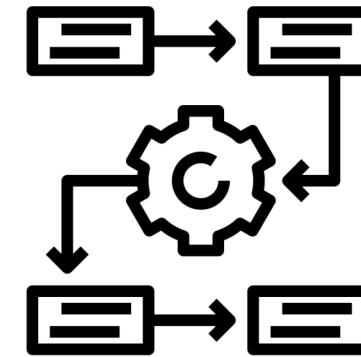


Server



DB

프로세스 품질



# | 소프트웨어 결함 예시 ① - {설계 오류}

[이슈] 쿠키런: 킹덤, 이번엔 '무한 뽑기' 터졌다

게임뉴스 | 박광석 기자 | 댓글: 31개 | 01-29

인벤 공식 앱으로 보기



이례적인 24시간 점검 이슈로 출시 초부터 난항을 겪었던 데브시스터즈의 신작 '쿠키런: 킹덤'에 이번엔 개발 단계에서의 마일리지 설계 오류로 인한 '무한 뽑기' 이슈가 불거졌다.

쿠키런 킹덤에서는 유료 재화인 크리스탈로 뽑기를 할 경우 일정량의 '마일리지'가 포인트 명목으로 지급된다. 이렇게 쌓인 마일리지로는 쿠키 영 훈석, 보물 뽑기, 별사탕 등 다양한 품목을 직접 구매할 수 있었는데, 그중에는 쿠키 뽑기를 1회 진행할 수 있는 '마법의 쿠키터' 아이템도 포함됐다. 가격은 80 마일리지다.

쿠키런 킹덤의 뽑기에서 중복 쿠키가 나오면 쿠키를 승급시킬 수 있는데, 이미 5성을 달성했다면 그 후부터 등장하는 중복 쿠키는 모두 마일리지로 환산된다. 이때 제공되는 마일리지가 1회 뽑기에 필요한 80 마일리지보다 더 많았다는 점이 문제가 됐다.

결국 이미 게임 속에 등장하는 모든 쿠키의 승급을 마친 상태의 유저라면, 뽑기를 반복하면 할수록 사용되는 유료 재화보다 더 많은 양의 마일리지가 계속 쌓이게 되는 셈이다.

이러한 구조적 허점을 활용하여 수십만 포인트의 마일리지를 쌓고, 이를 활용하여 '무한 뽑기'를 진행했다는 이들과 이에 대해 문제를 제기하는 유저들의 제보가 계속 이어지자, 데브시스터즈는 지난 28일, 별다른 공지 없이 급하게 마일리지 상점에서 '마법의 쿠키터'를 삭제하는 방식으로 대처했다.

개발 설계 오류

품질 검증 활동 부족

기업과 제품의 신뢰 하락

비즈니스 이미지 손상

금전적 손실

# | 소프트웨어 결함 예시 ② - {프로세스 오류}

## 정의선 “갈 길 멀다” 이유 있었다…

### 현대차 무상수리 절반이 SW 오류

업데이트 2024.01.19. 10:53



지난해 [현대차\(265,000원 ▼ 500 -0.19%\)](#)의 소프트웨어 오류가 빈번했던 것으로 나타났다. [기아\(122,400원 ▼ 100 -0.08%\)](#) 또한 1년 내내 소프트웨어 문제가 끊이지 않았다. SDV(소프트웨어 중심 차) 전환을 노리는 정의선 현대차그룹 회장은 최근 기술 경쟁력에 대해 여러 번 비판했다. 소프트웨어 관련 잦은 문제 발생과 무관치 않은 것으로 보인다. 현대차·기아는 관련 조직을 재편하기로 했다.

현대차·기아의 소프트웨어 오류는 대부분 신차와 전기차에서 발생했다. 이들은 SDV 전환 과정에서 새로 개발된 소프트웨어를 적용하고 있는데, 신뢰성 확보가 충분하지 않은 것으로 보인다. 수년간 테스트를 반복하며 안정성을 높이는 기존의 기계 부품과 달리 소프트웨어는 개발에서 적용까지의 시간이 짧아 예기치 않은 문제가 나타나는 빈도도 많다.

현대차 중 가장 많은 소프트웨어 오류가 나타난 차는 2022년 11월 출시한 그랜저다. 지난해 12건의 오류가 있었다. 이어 제네시스 GV60 7건, 전기차 아이오닉6 6건, 아이오닉5 4건 순이었다. GV70·GV80 전동화 모델도 각 4건을 기록했다. 기아는 스포티지 4건, 전기차 EV6·EV9·K5·니로·K8 등에서 각 3건의 소프트웨어 오류가 있었다.

소프트웨어 오류 중에는 시트·트렁크·경고등 점등 등 비교적 간단한 문제도 있었고, 배터리 충전 불가·주행 중 정지·시동 꺼짐 등 안전과 밀접한 것들도 있었다.

잦은 오류는 현대차·기아의 SDV 전략에 흠집을 냈다. 그蘭저의 경우 앞에 아무것도 없는 데도 센서가 오작동을 일으켜 긴급 정지하는 오류가 발생 ‘유령 보는 차’라는 오명을 얻었고, EV9도 뒷좌석에 아무도 타지 않았는데 후석승객알림이 울려 ‘유령 타는 차’라는 지적을 받았다.

정 회장은 최근 SDV 전환의 어려움을 여러 차례 밝혔다. 정 회장은 그룹 신년회와 CES 2024에서 “소프트웨어 경쟁에서 뒤처진 면이 있다”.”(소프트웨어) 경쟁력과 품질에서 모두 최고가 되어야 한다”.”우리가 (SDV 전환이) 좀 늦었다“.”갈 길이 멀다”라고 말했다.

현대차그룹은 현대차·기아 외에도 [현대모비스\(219,000원 ▼ 2,000 -0.9%\)](#), [현대오토에버\(152,100원 ▼ 400 -0.26%\)](#) 등 그룹 내 여러 주체가 소프트웨어를 각자 개발해 종 구난방이다. 잦은 오류도 이런 개발 환경 탓으로 여겨진다. 정 회장은 난잡한 소프트웨어 개발 방식에 문제가 있다고 진단, 최근 관련 조직을 통합하는 조직 재편을 단행했다.

기대한 동작을  
수행하지 않음

동작하지 말아야  
함에도 기능 동작

금전적 손실

비즈니스 이미지  
손상

부상 또는 사망

프로세스 오류

# | 소프트웨어 결함 예시 ③ - {환경적 오류}

## 워싱턴 지하철 추돌사고…80여 명 사상

입력 2009.06.23 (07:13) | 수정 2009.06.23 (16:08)



미국 수도인 워싱턴 D.C.에서 지하철 추돌사고가 일어나 지금까지 9명이 숨지고 70여명이 다쳤습니다.

워싱턴 DC 에드리엔 펜티 시장은 우리 시각 오늘 아침 6시쯤 워싱턴 북부 도심의 지상구간에서 발생한 이 사고로 지금까지 최소 9명이 숨졌다고 밝혔습니다.

이와관련해 소방당국은 현장에서 70여 명을 응급치료한 뒤 병원으로 이송했으며 이중 일부는 부상 정도가 심각하다고 말했습니다.

오늘 사고는 지하철역 근처에 정차해있던 열차를 뒤따르던 열차가 추돌하면서 발생했으며, 정확한 사고원인이 나오는데는 며칠이 걸릴 것으로 전해졌습니다.

이 사고로 일부 객차가 맞은 편 열차 위로 퉁겨져 올라갔으며, 구조대원들은 인명 구조에 어려움을 겪었던 것으로 전해졌습니다.

교통 당국이 사고 원인을 조사하고 있는 가운데, 연방교통안전위원회는 열차의 속도 등을 기록한 장치를 확보했지만, 사고 열차의 과속 여부에 대해서는 밝히지 않았습니다.

주미 한국 대사관은 한인 희생자나 부상자는 현재까지 없는 것으로 파악됐다고 밝혔습니다.

워싱턴에서는 2년 전에도 다운타운에서 지하철 객차가 탈선해 20명이 다치고 터널에 갇힌 승객 60명이 구조된 적이 있습니다.

출처 : /news.kbs.co.kr/

동작하지 말아야  
함에도 기능 동작

제품의 외부적  
요인에서 발생한  
결함

금전적 손실

부상 또는 사망

품질 예방 활동  
부족

뉴스

## 듀크 포인트 페리터미널 충돌사고로 폐쇄

권민수 기자 ms@vanchosun.com ▶ 기자의 다른 기사보기



최종수정 : 2011-12-22 14:18



### 나나이모행 디파처베이 터미널로 우회

밴쿠버 아일랜드 나나이모의 듀크 포인트 BC페리(BC Ferries) 터미널이 폐리 충돌사고로 21일부터 임시 폐쇄됐다.

이에 따라 당일부터 밴쿠버 남서쪽의 트와슨 터미널을 출발해 나나이모의 듀크 포인트로 가던 폐리는 나나이모의 디파처베이 터미널로 운행하고 있다.

디파처베이와 듀크포인트는 둘 다 나나이모 인근 폐리 터미널이지만, 디파처베이가 듀크포인트보다 수 킬로미터 북쪽에 있는 관계로 트와슨에서 출항해 디파처베이로 가면 전보다 운행 시간이 더 걸린다.

밴쿠버에서 폐리를 이용하는 승객들도 이 점을 알고, 22일 신설된 트와슨-디파처베이 폐리보다 기존의 밴쿠버 북서쪽의 홀슈베이에서 출항해 디파처베이로 가는 폐리에 몰리고 있다. 듀크 포인트 폐쇄에 따라 나나이모 인근에서 유일하게 가동 중인 폐리항구가 된 디파처베이에도 차량과 승객이 몰렸다.

한편 충돌사는 20일 오후 3시경 코스털 인스파레이션 호가 접항하던 중에 발생했다. 이 사고로 뱃머리 부분과 항구의 폐리 도킹시설이 부서졌다. 선박 수리는 1개월 가량, 도킹 시설 수리는 수 개월이 소요될 전망이다.

21일 BC페리 대변인은 “사고원인은 상세한 조사 후에 나오겠지만, 초기 조사 결과 전자기기의 고장이 충돌의 원인으로 보인다”고 밝혔다. BC페리는 듀크 포인트-트와슨 탑승을 예약한 이들은 디파처베이-트와슨행 폐리를 탑승할 수 있다고 밝혔다. 또한 듀크 포인트와 디파처 베이 사이에는 임시 셔틀버스를 운행할 예정이다.

출처 : /www.vanchosun.com/news

# | 소프트웨어 결함 예시 ④ - {환경적 오류}



기대한 동작을  
수행하지 않음

오작동

환경적 요인

품질 예방 활동  
부족

## Y2K 넘겨도 인식오류 계속

제2 Y2K로 떠오르고 있는 컴퓨터 시간 인식오류는 유닉스의 2038년 인식문제. 대형 컴퓨터 운영체제로 널리 쓰이는 유닉스는 70년 1월 1일부터 초단위로 시간을 계산해왔는데 다음 세기 38년 1월 19일 오전 3시 14분 7초가 되면 더 이상 시간을 읽을 수 있는 공간이 없어져 문제가 된다는 것이다.

유닉스도 이를 의식해 신제품에는 최소한 2000억년 이상 시간 인식오류 가능성을 배제했다고 주장하고 있다.

소프트웨어를 만드는 대표적 프로그램 언어인 '비주얼 C++' 역시 비슷한 문제를 안고 있는데 2036년이면 시간을 '0'으로 잘못 인식하는 것으로 전해졌다.

앞으로 수십년 뒤의 일을 지금부터 걱정해야 할 이유는 여러가지다. 먼저 저당권 등 장기금융상품의 경우 문제가 언제 터질지 모를 만큼 심각하다. 이보다 더 큰 문제는 새로운 소프트웨어라 하더라도 그 바탕은 오래된 소프트웨어에 두고 있는 경우가 많다는 데 있다.

요컨대 기존 소프트웨어의 공간은 그대로 둔 채 기능적 향상을 꾀한 경우가 대부분이라 어디서부터 손을 써야 할지 모를 정도로 문제가 광범위하다는 것이다.

출처 : [www.mk.co.kr/news](http://www.mk.co.kr/news)

# | 소프트웨어 결함 예시 ⑤ – {인적 오류}

## 교토 대학 데이터 손실 사고로 살펴보는 ‘내부자 위협으로서의 기기’

Christopher Burgess | CSO © 2022.01.13

작년 12월 28일, 일본 교토 대학 컴퓨팅 및 미디어 연구팀 소장 오카베 토시오가 슈퍼컴퓨팅 오류로 77TB 상당의 사용자 데이터가 손실된 것에 대해 사과했다. 잃은 데이터는 교토 대학 내 연구 기관 14곳의 데이터 파일 3,400만 개를 포함한다.



© Getty Images Bank

12월 16일 교토 대학 정보 관리 및 통신 연구소는 공식 홈페이지를 통해 사용자에게 이번 사건의 개요를 포함한 사과문을 개별적으로 전송하겠다고 밝혔다. 연구소에 따르면, 작년 12월 14일~12월 16일에 /LARGE0 디렉토리를 지원하는 슈퍼컴퓨터 시스템 백업 스토리지 프로그램에 결함이 발생해 뜻하지 않게 데이터가 삭제됐다.

### 데이터 손실이 발생한 경위

HPE가 설명한 것처럼, 백업 스크립트를 실행하는 기기는 가시성과 가독성을 개선하도록 조정됐다. 하지만 스크립트가 수정됐고, 이전 스크립트를 덮어쓰는 방식으로 재배치됐다. 그 결과 하나의 작은 결함이 발생했으며, 결과 코드가 중간부터 새로고침됐다. 이로 인해 삭제 지정된 파일 디렉토리가 아닌, /LARGE0 디렉토리가 삭제된 것이다.

이번 문제의 재발 가능성을 낮추기 위해 HPE가 권고한 사항은 다음과 같다.

- 슈퍼컴퓨터 시스템에 적용하기 전, 코드를 꼼꼼하게 확인한다.
- 이번 사건의 영향과 관련해 검사를 실시하고, 재발 방지를 위한 개선점을 제시한다.
- 휴먼 에러가 반복되지 않도록 엔지니어를 재교육한다.

교토 대학은 1월 말까지 백업 프로세스를 전면 중단했다.

기대한 동작을  
수행하지 않음

금전적 손실

비즈니스 이미지  
손상 및 신뢰 하락

프로세스 및 인적  
오류

# | 품질 관리자 관점의 소프트웨어 결함

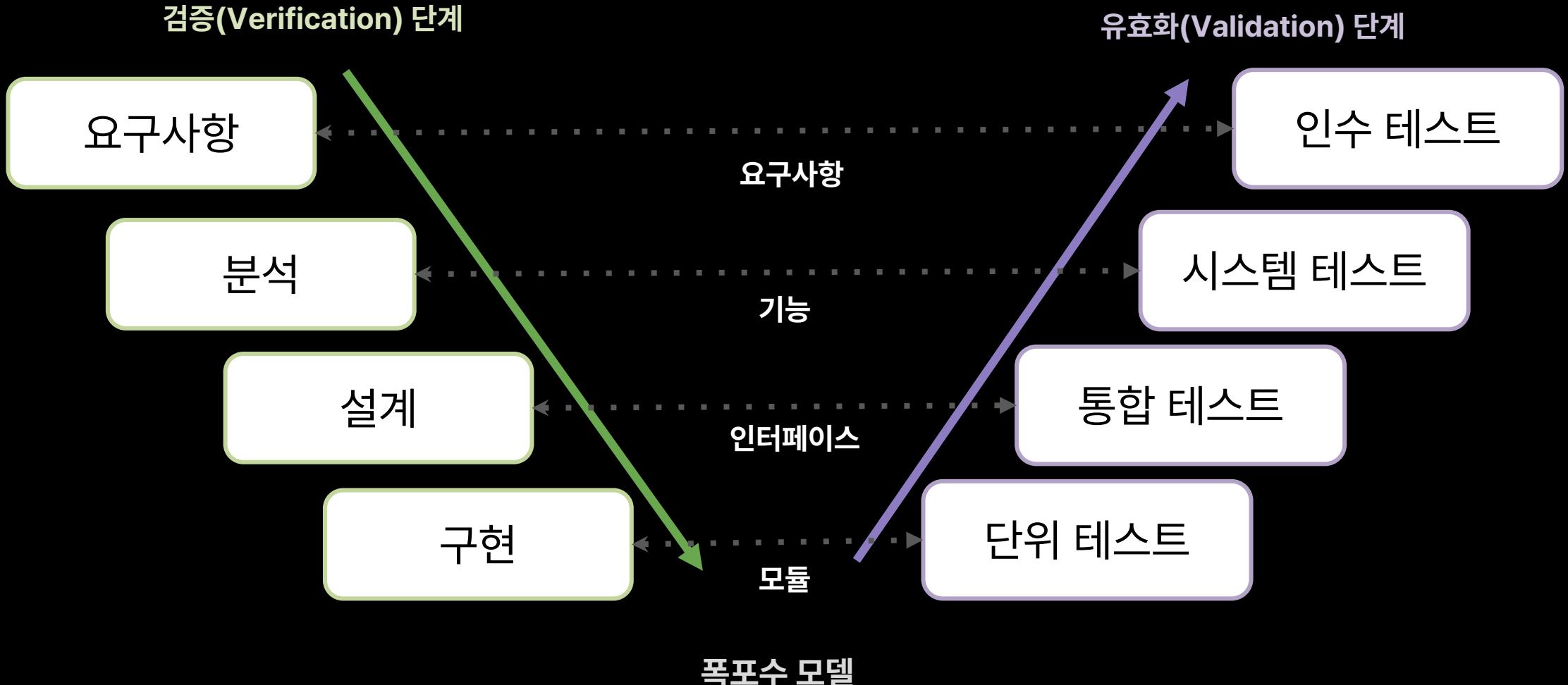
- **기능적 오류**(코드에 존재하는 오류)
  - 소프트웨어 또는 시스템이 요구사항과 다르게 기능이 동작하거나 동작하지 않아야 함에도 동작
  - 개발자 실수, 알고리즘 오류, 설계 결함, 계산 오류
  - **데이터** 입력/저장/조회 및 누락 등 처리 문제, 데이터 손실
- **문서 결함**
  - 기획서, 개발 설계서, 데이터 테이블과 같은 문서 내 요구사항 누락 또는 문서 내 존재하는 오류
  - 유저 시나리오 또는 테스트 케이스 누락 및 사용성 문제
- **설계 오류**
  - 복잡한 코드, 개발 설계의 복잡도로 발생되는 문제
  - 구조, 기능, 영향도 파악의 어려움, 수정/개선/최적화 등 유지보수가 어려운 코드와 **아키텍처**
  - 시스템, 모듈 간 인터페이스가 안되는 문제
  - 소프트웨어 **테스팅 설계 오류**
- **성능 문제**
  - 클라이언트, 서버, 데이터가 적절한 성능을 제공하지 못하는 문제
  - 소프트웨어 반응 속도 및 처리 능력 저하, 메모리 부족/누수/비효율적 사용, 시스템 충돌
  - 데이터 베이스 쿼리 지연

# | 품질 관리자 관점의 소프트웨어 결함

- **사용성 오류**
  - 사용자 인터페이스의 부적절함(직관적이지 않은 UI, 불필요하게 복잡한 화면 흐름), 사용자의 기대를 만족시키지 못하는 기능, 사용자의 자기 주도 학습의 어려움
- **보안 취약성**
  - 취약한 암호화, 접근 우회 취약성
  - 해커의 공격, 어뷰징, 신원 도용, 사용자 데이터 유출
- **이식성 문제**
  - 소프트웨어가 다양한 환경, 시스템, 하드웨어로 이식되지 못하거나 통합되지 못하는 문제
  - 다양한 언어, 문화, 국가에 맞지 않는 기능 및 UI 문제
  - 사용자의 접근성 제한, 시스템 간 정보 교환 실패, 텍스트 인코딩 오류, 문화적 민감성을 고려하지 못함
- **환경에 의한 오류**
  - 외부 물리적 요인의 영향으로 인한 개발 환경의 변경, 기술이나 시스템의 변경, 하드웨어 조건 변경
- **기타**
  - 시간적 압박, 작업 담당자의 역량 및 기술 부족, 커뮤니케이션 오류
  - 기능/성능상 **예외처리 부재**, 체계적인 **프로세스 부재**

## 02 소프트웨어 테스팅 이해

# | 소프트웨어 테스트 단계



# | 소프트웨어 테스트 단계

## 단위 테스트

- 하나의 모듈, 프로그램, 클래스, 메서드와 같은 가장 작은 단위를 대상으로 테스트
- 소스 코드를 중심으로 개발자 테스트 수행

## 통합 테스트

- 통합 후 모듈 간 인터페이스, 즉 상호 작용이 발생하는 기능과 동작을 테스트
- 통합 구간의 코드를 중심으로 개발자 테스트 수행(경우에 따라 테스트 조직 수행)

## 시스템 테스트

- 소프트웨어와 하드웨어 결합 후 기능·성능적 요구사항과 인터페이스 전체를 테스트
- 테스트 환경에서 테스트 전문가에 의해 테스트 수행

## 인수 테스트

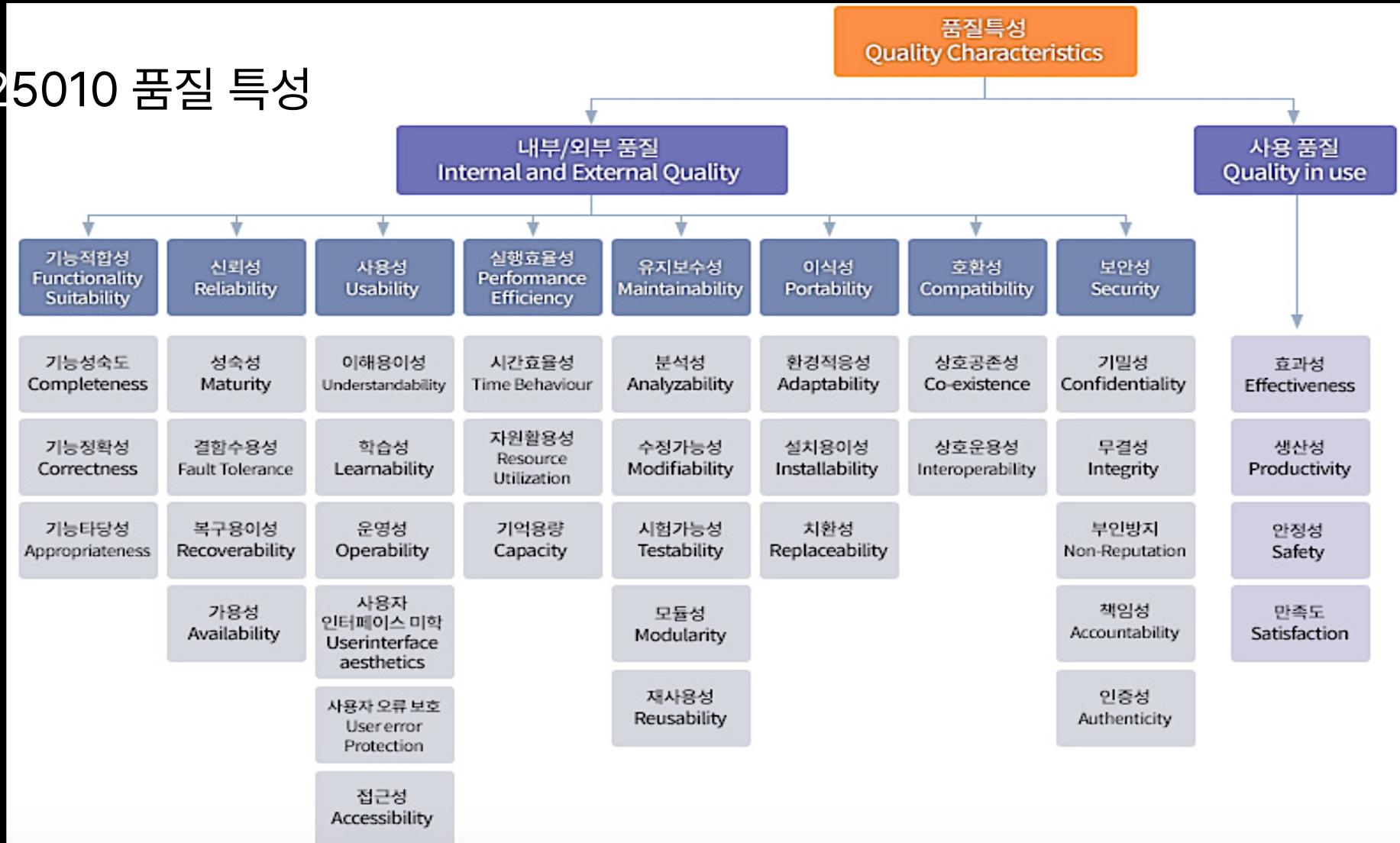
- 실 운영 환경에서 사용할 준비가 되었는지 확신을 얻기 위한 과정
- 요구사항 충족 여부, 제품 완성도를 테스트
- 라이브 환경에서 제품을 사용하게 될 고객에 의해 테스트 수행

# | 품질 특성과 상응하는 소프트웨어 테스트 종류

테스트 단계	테스트 종류	설명
단위, 통합, 시스템, 인수	<b>기능 테스트</b>	시스템의 기능과 특징 그리고 시스템 간 상호 작용을 확인하는 테스트
통합, 시스템	<b>비기능 테스트</b>	소프트웨어의 사용성, 성능, 유용성, 안전성, 확장성 등 비기능적인 측면에서 시스템의 동작을 확인하는 테스트 -사용성 테스트 : 소프트웨어를 쉽게 사용할 수 있는 정도를 측정하고 검증 -서버 부하 테스트 : 시스템에 부하를 주면서 응답시간, 처리량, 속도 등 성능 지표를 측정하고 검증 -그 외 클라이언트 성능 테스트, 네트워크 테스트, 호환성 테스트, 어뷰징 테스트, API 테스트, 데이터 테스트 등 포함
시스템, 인수	<b>회복 테스트</b>	문제나 장애가 발생했을 때 소프트웨어나 시스템의 복귀 능력을 검증하는 테스트
시스템	<b>보안 테스트</b>	외부의 침입이나 해킹에 대응하는 능력을 검증하는 테스트
시스템	<b>회귀 테스트</b>	이미 테스트된 시스템을 반복해서 확인하는 것으로 결함 수정 이후 변경한 결과로 이전에 제대로 작동하던 기능에 문제가 발생하지 않는지 확인하는 테스트

# | 소프트웨어 테스트 종류

## ISO/IEC 25010 품질 특성



# | 소프트웨어 테스트 종류

## 프로덕트 품질 특성별 테스트 종류 및 수행 목적

품질특성	테스트 종류	테스트 목적
기능성	기능 테스트	테스트 대상의 주요 요소 및 동작, 절차를 확인한다.
	정책 검수	기업 내·외부 정책 적용을 확인한다(예: 구글, 애플 플랫폼 정책).
	API 테스트	클라이언트의 요청에 대한 서버의 응답을 검증하고 부하를 발생시켜 성능을 확인한다.
	데이터 테스트	데이터 품질 및 데이터 정합성을 검증한다.
신뢰성	보안, 어뷰징 테스트	예외 사항에 대한 보안, 어뷰징 케이스를 확인한다.
	네트워크 테스트	네트워크 전환이나 단절 상황 시 데이터 손실이 없도록 구현되었는지, 네트워크 레이턴시 부하로 발생하는 이슈가 없는지 확인한다.
이식성	디바이스 기능 연동	디바이스 하드웨어 및 소프트웨어 기능에 따른 영향도를 확인한다.
	공존성 테스트	다른 애플리케이션과의 상호 작용 능력을 확인한다.
	호환성 테스트	다양한 디바이스의 해상도와 운영체제 버전을 고려했는지 확인한다.
효율성	클라이언트 성능	하드웨어 환경상에서 제품을 원활하게 이용 가능한지 확인한다(예: 해상도, 메모리, cpu 등).
	서버 부하 테스트	서버 부하·성능을 검수한다.
유지보수성	운영 테스트	점검, 업데이트 등 운영 중 소프트웨어 변경에 따른 시나리오 기반 테스트를 진행한다.

# | 소프트웨어 테스트 수행 방식

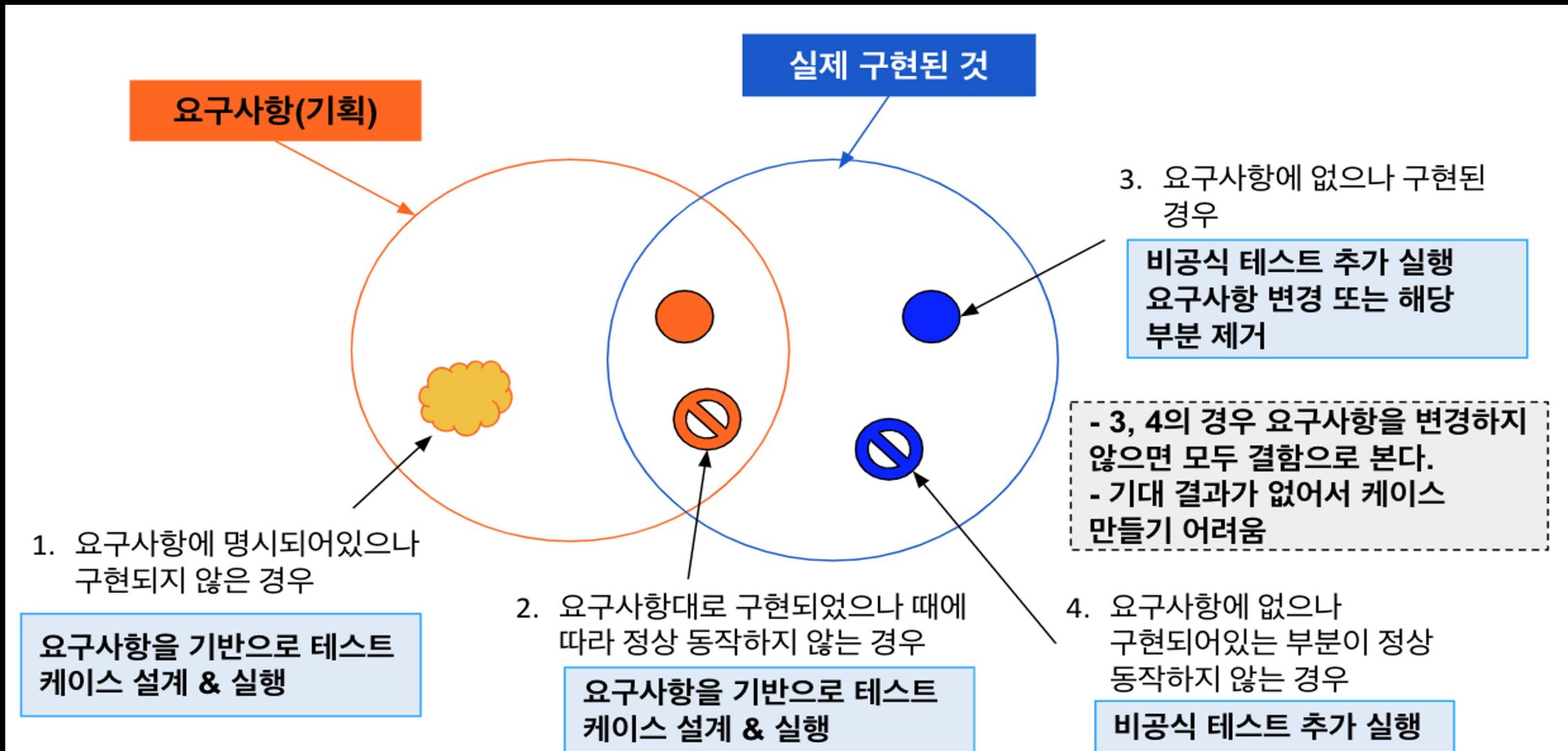
정적 테스트 (산출물 리뷰, 정적 분석)		동적 테스트 (명세기반, 구조기반, 경험기반)	
리뷰	<p>소프트웨어를 사용하지 않고 개발 중간 산출물의 리뷰나 분석을 통해 결함을 발견하는 방법.</p> <ul style="list-style-type: none"><li>✓ 리뷰 종류 : 비공식적 리뷰, 기술적 리뷰, 워크쓰루, 인스펙션</li><li>✓ 결함 종류 : 요구사항 누락, 코딩 표준 위반, 개발 설계 결함, 부정확한 인터페이스 명세 등</li></ul>	<p>테스트 대상이 되는 소프트웨어를 테스트 환경이나 실 운영 환경에서 실행하여 결함을 발견하는 방법.</p>	<p>명세기반</p> <p>고객의 요구사항 명세서나 기획서를 기반으로 테스트 케이스를 도출하고 이를 수행하여 중대한 결함이 없음을 보장하는 활동</p> <ul style="list-style-type: none"><li>✓ 기법 종류 : 동등 분할, 경계값 분석, 결정 테이블, 상태 전이, 유즈 케이스, 조합 테스팅</li></ul>
정적 분석	<p>소프트웨어를 실행하지 않은 상태에서 도구를 활용하여 테스트를 수행하는 방법으로 소스코드와 모델에서 결함을 발견하는 것.</p> <ul style="list-style-type: none"><li>✓ 결함 종류 : 제어 흐름, 데이터 흐름과 같은 프로그램 코드나 HTML, XML과 같은 산출물 분석을 통한 결함 발견, 코드와 개발 설계 복잡도 경고, 정의되지 않은 변수, 일관성이 없는 인터페이스, 보안 취약성 등</li></ul>	<p>구조기반</p> <p>구현된 소프트웨어나 시스템의 구조와 소스코드를 중심으로 테스트를 수행.</p> <ul style="list-style-type: none"><li>✓ 컴포넌트 레벨 : 구문, 결정, 분기문 등 코드 자체</li><li>✓ 통합 레벨 : 모듈 간 호출하는 관계</li><li>✓ 시스템 레벨 : 메뉴, 비즈니스 프로세스, 웹페이지 구조</li><li>✓ 기법 종류 : 구문, 결정, 조건, 조건/결정, 다중 조건 등</li></ul>	<p>경험기반</p> <p>테스터의 경험, 직관, 기술 능력을 바탕으로 테스트 케이스를 도출하고 이를 수행하여 결함을 발견하는 방법.</p> <ul style="list-style-type: none"><li>✓ 기법 종류 : 오류 추정, 탐색적 테스팅 등</li></ul>

# | 테스트 케이스

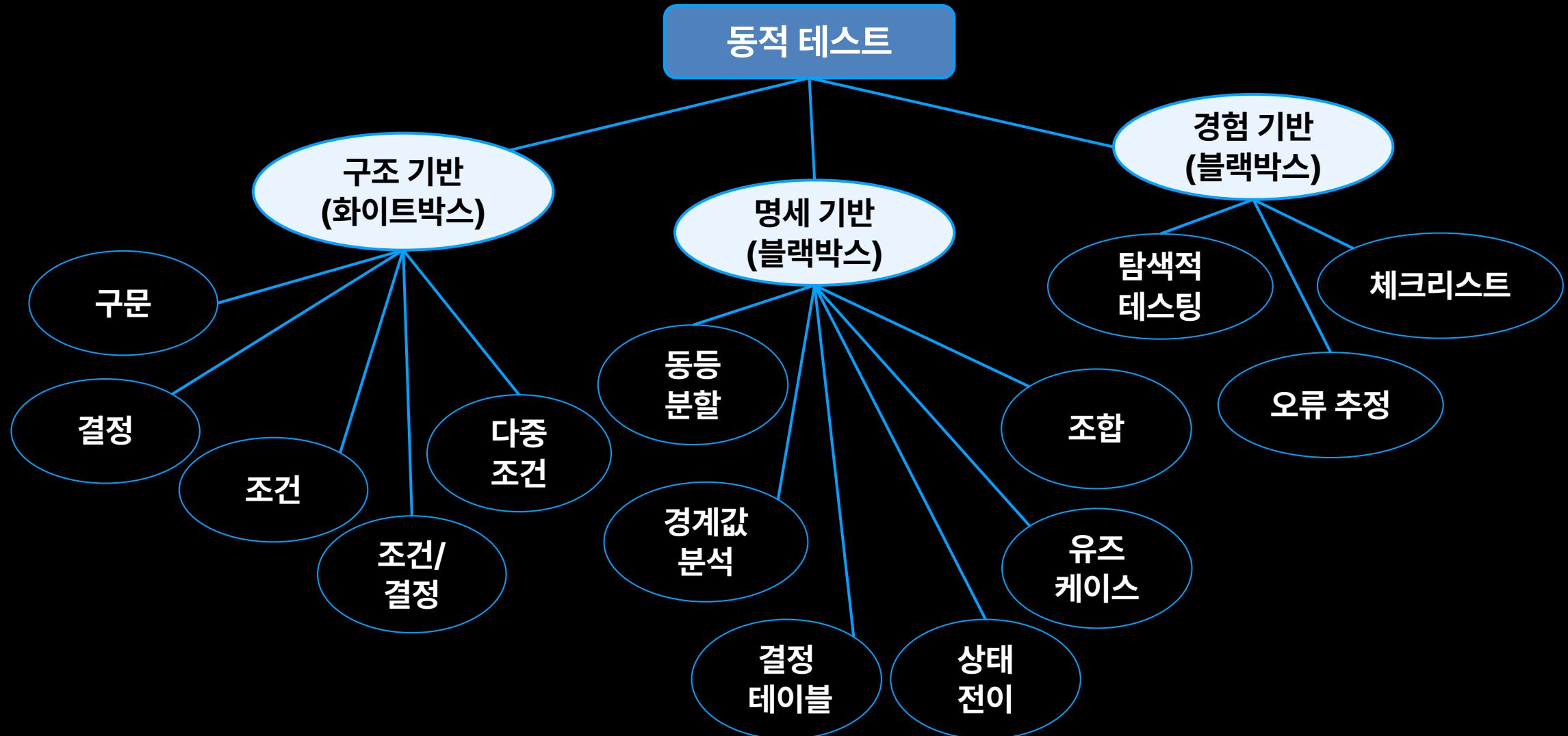
- 테스트 케이스란,  
소프트웨어가 요구사항에 준수하는지 그리고 예상된 결과에 만족하는지 확인하기 위한 목적으로 테스트를 실행하는 절차를 구체적으로 기술해 둔 문서로 테스트 목적, 입력값, 실행 조건, 수행 절차, 예상 결과 등의 집합으로 구성되어 있다.
- 목적  
최적의 테스트 케이스로 가능한 많은 결함을 발견할 수 있도록 하는 것으로, 테스트 대상을 테스트하여 일정 수준의 테스트 보장성을 확보하는 것을 목표로 한다.

ID	카테고리	사전조건	수행 절차	기대 결과	확인결과	중요도	비고
001	테스트 할 기능이 위치한 경로	테스트 실행을 위한 특정 조건	예상 결과를 확인하기 위한 프로그램의 특정 경로를 실행하는 절차	기대하는 기능 동작의 결과	Pass Fail Block N/I	Critical Major Minor Trivial	

# | 테스트 커버리지



# | 소프트웨어 테스트 설계 기법



# | 소프트웨어 테스트 설계 기법

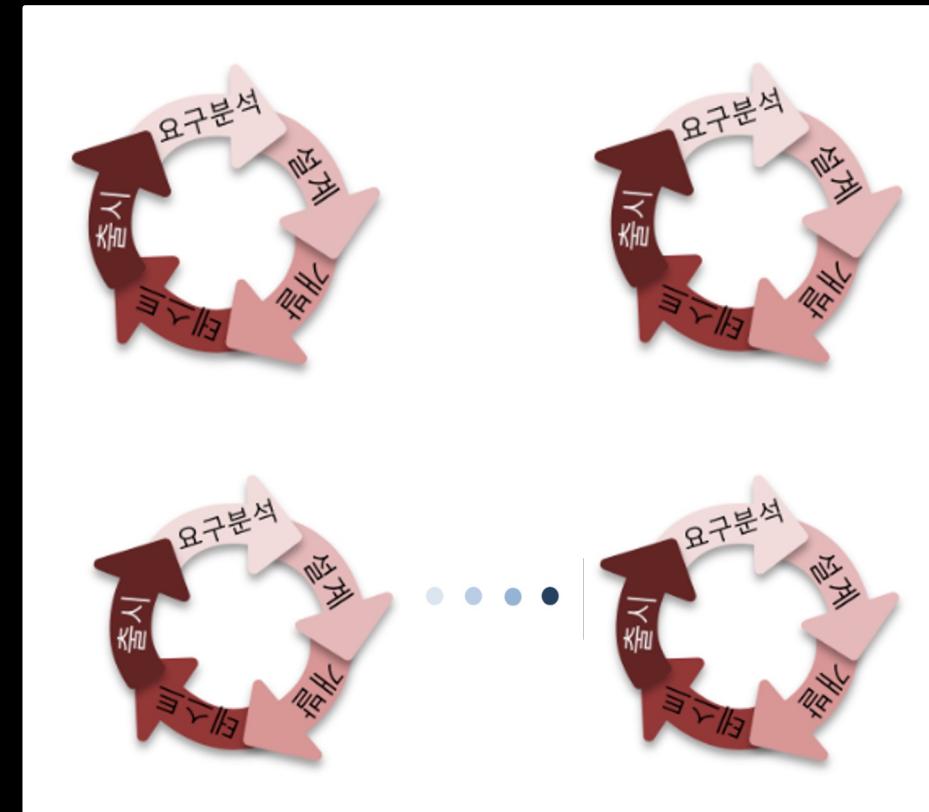
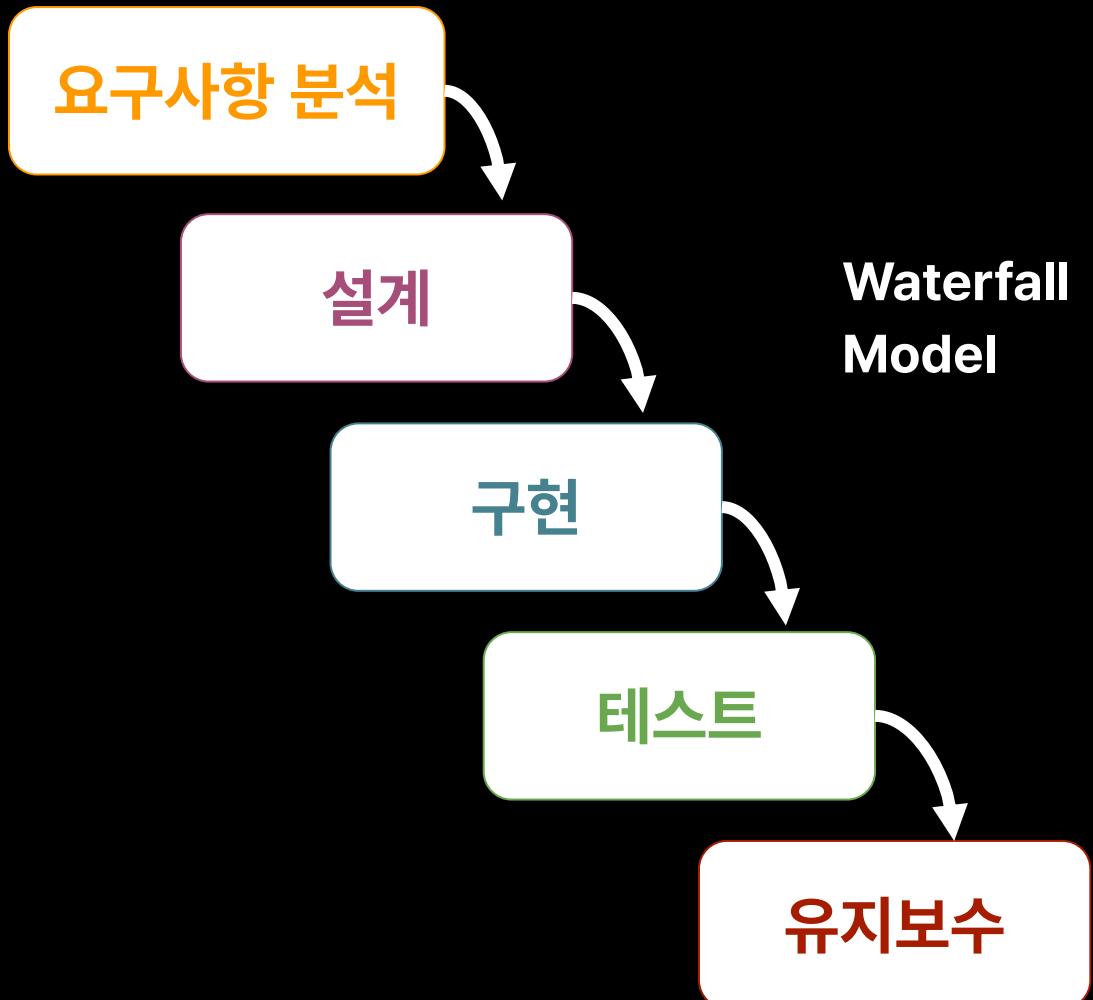
- 테스트 설계 기법
  - 테스트 베이스와 소프트웨어 제품을 분석하여 테스트 케이스를 도출하기 위해 사용하는 방법
  - 계획된 테스트 전략을 구체화하는 방법
  - 테스트 수행 방법, 테스트 수준 결정, 테스트 조건 식별
- 설계 기법이 필요한 이유
  - 프로젝트 상황에 맞는 목적 달성과 전략을 실현하기 위해
  - 모든 경우의 수를 테스트하는 것이 불가능하기 때문
  - 테스트 개수는 줄이면서 충분한 커버리지를 만족하기 위해
  - 결함이 발생될 가능성이 높은 범위를 식별하기 위해

# | 소프트웨어 테스트 설계 기법

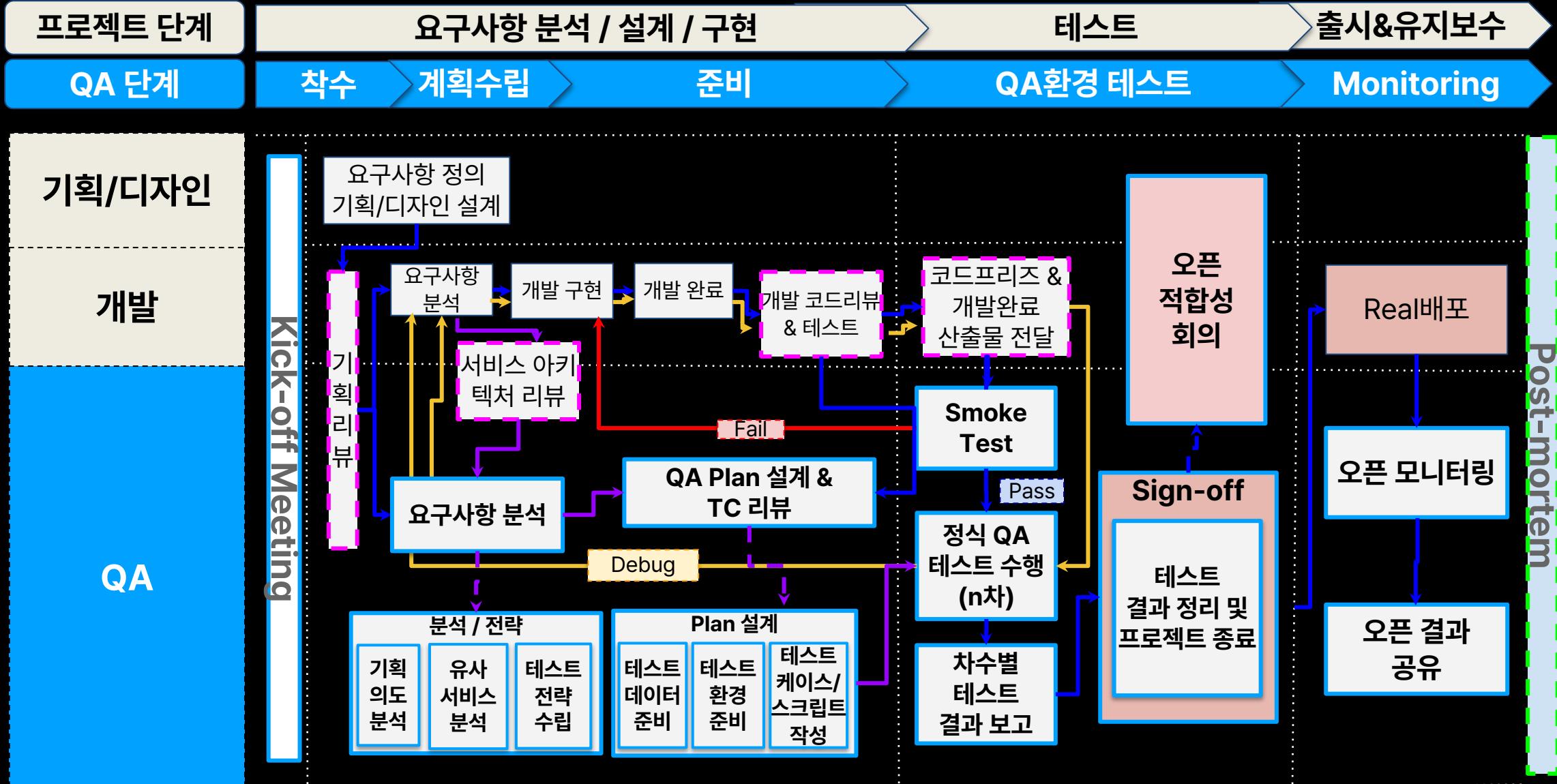
- 테스트 설계 기법의 종류
  - 명세 기반 기법
  - 경험 기반 기법
  - 구조 기반 기법
- 테스트 설계 기법 선택에 영향을 주는 요소
  - 테스트 대상 시스템의 종류와 복잡도
  - 리스크 수준
  - 고객 요구사항 기준
  - 테스터의 역량과 지식 수준
  - 테스트 수행 일정의 시간적 제약

## 03 테스트 프로세스

# | 소프트웨어 개발 프로세스



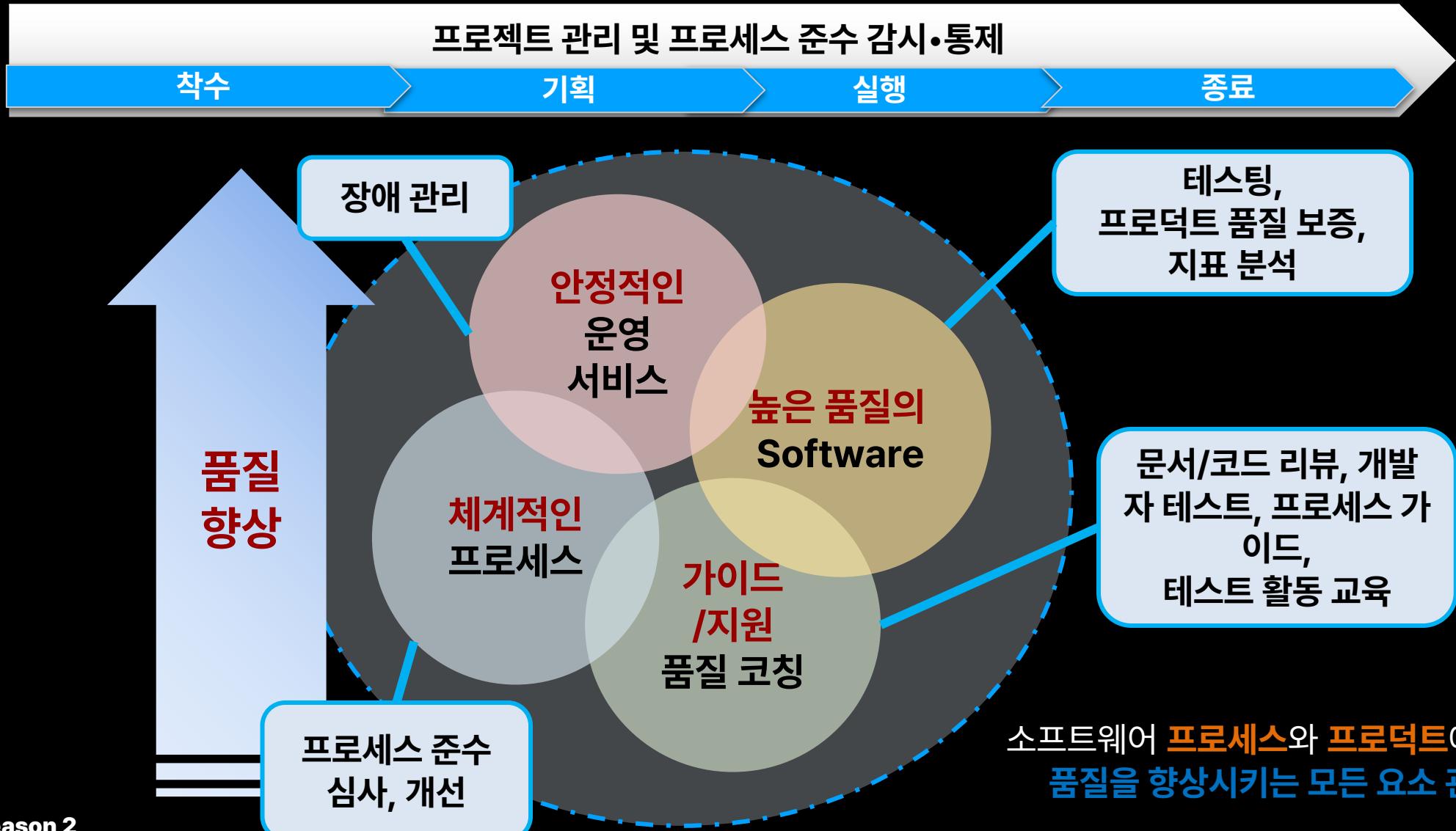
# 프로세스 Flow



# | 소프트웨어 테스트 프로세스(절차)

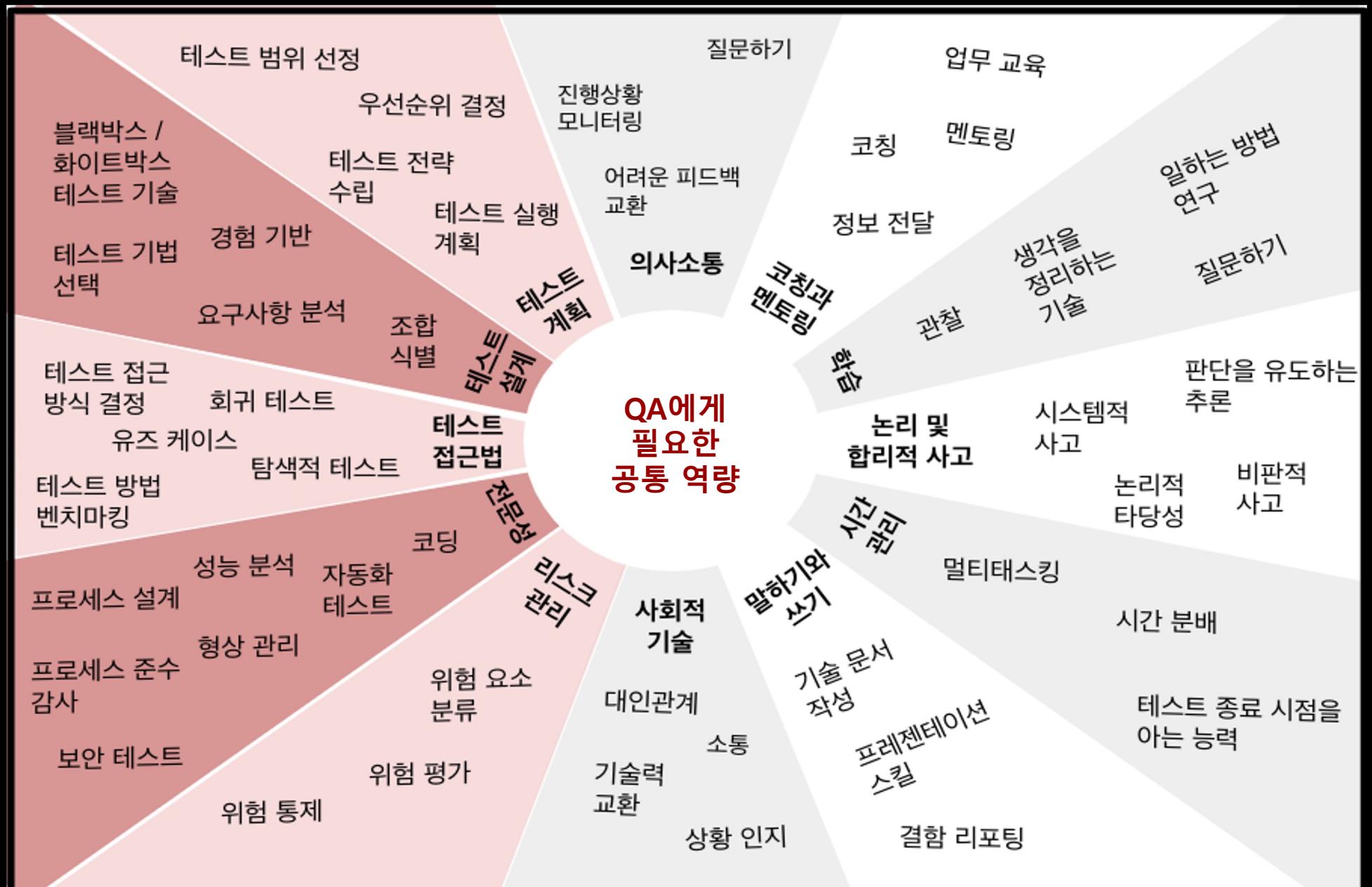


# | QA 역할과 활동 범위



## 04 ‘보다 나은’ QA가 되는 방법

# | QA의 요건과 자질



# | Up-Skilling

초급자

- **소프트웨어 공학의 기본 개념 이해**
  - 개발 언어 학습, 플랫폼 운영체제에 대한 이해, 개발 설계 구조 및 생명주기에 대한 이해
- **소프트웨어 테스팅 기본 지식 학습**
  - 테스팅 기본 원리 이해, 테스트 케이스 작성, 버그 추정 및 리포팅
- **테스팅 지원 도구 사용 방법 습득**
  - 네트워크, 클라이언트 성능, API 등 도구를 활용한 테스팅 방법을 학습하고 실습을 통한 연습

중급자

- **소프트웨어 공학의 개념 이해 및 테스팅 적용 기술 보유**
  - 하나 이상의 개발 언어 습득, 서버·DB·API 구조 이해, 개발 생명주기 단계별 테스팅 기술 지원
- **소프트웨어 테스팅 전문 지식 및 기술 보유**
  - 테스트 계획 수립, 프로세스 개선 사항 도출, 테스트 산출물 점검, 테스트 활동 관리
- **테스팅 지원 도구 사용 및 분석 기술 보유**
  - 원활한 테스팅 도구 사용 및 테스트 수행, 자동화 스크립트 작성, 데이터 분석을 통한 품질 평가

중급자

- **소프트웨어 공학의 개념 이해 및 테스팅 적용 기술 보유**
  - 하나 이상의 개발 언어 습득, 서버·DB·API 구조 이해, 개발 생명주기 단계별 테스팅 기술 지원
- **소프트웨어 테스팅 전문 지식 및 기술 보유**
  - 테스트 계획 수립, 프로세스 개선 사항 도출, 테스트 산출물 점검, 테스트 활동 관리
- **테스팅 지원 도구 사용 및 분석 기술 보유**
  - 원활한 테스팅 도구 사용 및 테스트 수행, 자동화 스크립트 작성, 데이터 분석을 통한 품질 평가

# | QA의 미래

## 해외게임사 역차별에 AI로 일감 줄어...게임업계 '경고등'

입력 2024.05.14. 오후 3:37

기사원문



김한준 기자



1



8



가가



### | 국내 규제 적용 어려운 해외게임사...AI 기술 발전에 따라 QA-로컬라이징 인력 긴장

"해외게임사랑 경쟁하는 게 매년 더 어려워지는데 이젠 AI도 신경써야 되네요"

국내 중소게임사의 힘겨운 행보가 2분기에도 이어지고 있다. 해외게임사가 국내 시장을 공략함에 따라 경쟁이 더욱 치열해진 것은 이미 오래 전부터 이어지는 일. 최근에는 이에 더해 해외 게임사 역차별 문제와 AI 기술 대두로 인한 일감 감소까지 더해져 시름이 깊어지고 있다.

출처 : /v.daum.net

# | QA의 미래

brunch story

이와같이 QA는 품질에 영향을 주는 모든 요소를 관리하고 제어하는 역할입니다.

품질 관리자의 역할 참고) 과연 이런 범위와 역할까지 AI가 대체할 수 있을까요?

소프트웨어가 데이터를 분석하지만 분석한 데이터를 이해하고 활용하는 역할은 AI에게는 어렵습니다. AI가 이해하는 범위와 역할을 확장하려면 어떻게 사용되고 어떤 방향으로 갈지 사람에게 의존하는 경우가 많습니다. 예전에는 문제, 공정성, 윤리적 판단과 같이 사람의 개입으로만 해결할 수 있는 문제를 해결하는 역할이었지만,

AI로 일부는 대체가 가능하겠지만, QA 역할 범위 전부를 대체하는 것은 (아직은) 불가능하다고 생각합니다. 물론, 사람의 개입이 필요한 영역과 십수년간 경험으로 쌓인 정보와 노하우가 모두 데이터화 된다면 모르겠지만.. 이 또한 일정 시간이 필요할 것이고 어떤 범위까지 데이터화 할 것인지에 대한 논의가 필요하다.

AI가 QA 업무를 빼앗고 대체될 것을 우려하며 걱정하고 설레발 치기 보다, QA 영역에서 AI를 잘 활용할 수 있는 방안을 마련하는 것이 더 필요하지 않을까 한다.

기술을 사람을 돋고 유지하기 위해 개발되는 것이라는 도구적 가치라는 관점으로 바라보고 품질을 위한 업무와 비용, 시간을 효율적으로 개선하는데 적절히 활용한다면 사람의 개입이 필요한 영역(문제를 예방하고 방지하기 위한 활동, 좋은 프로세스를 설계하고 평가하는 활동, 프로젝트를 컨트롤 하는 역할 등)에 대한 고유한 가치도 인정될 것이다.

## ⭐ AI의 도움과 함께하는 QA 산업

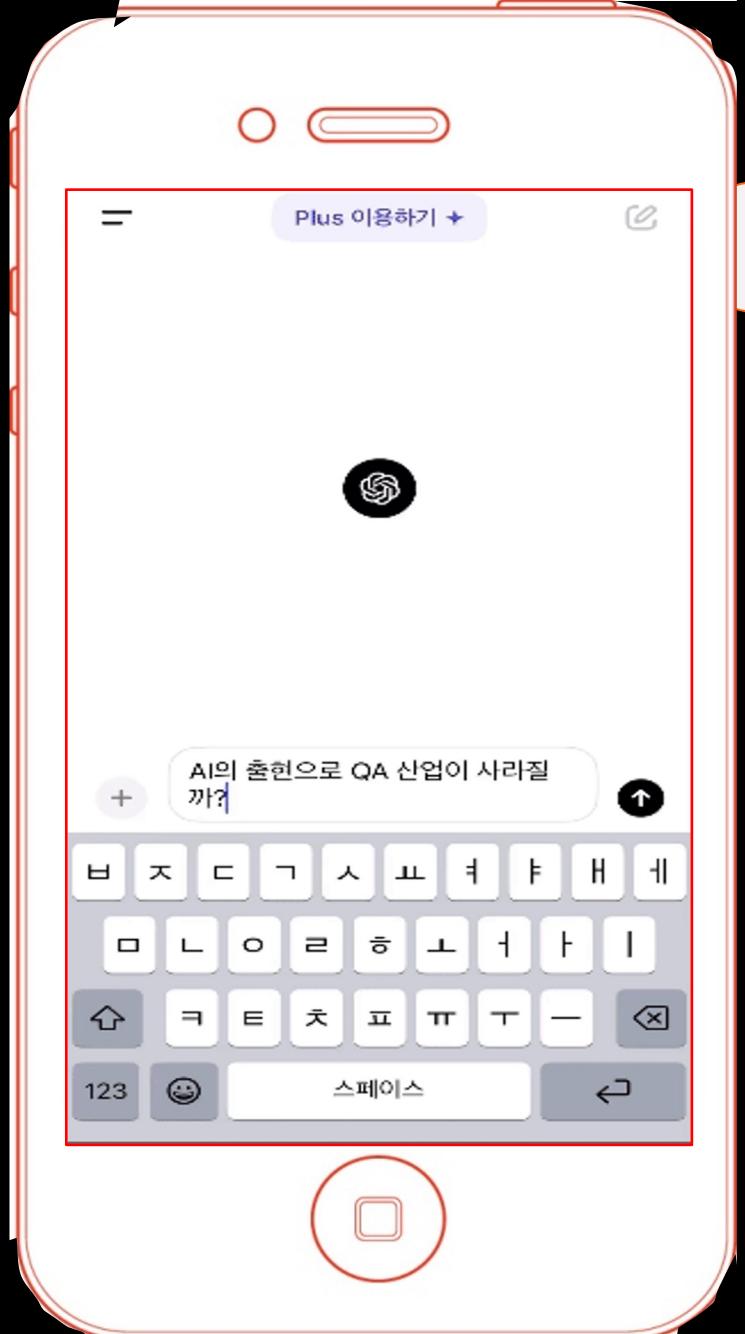
AI의 도움으로 QA 산업은 더욱 발전할 것으로 기대됩니다. AI 기술을 활용하는 QA 전문가들은 테스트 케이스와 시나리오의 생성, 코드 작성 등에 AI와 협력하여 다양한 솔루션을 구현할 수 있습니다. 이로 인해 QA 산업에서는 더 많은 일자리가 생성될 것입니다.

그러나 AI의 도움만으로는 충분하지 않습니다. AI의 결과물을 검증하고 내용의 적절성을 확인하는 작업은 프로그램 엔지니어들에게 필요한 역할입니다. AI가 생산하는 텍스트의 품질을 확보하기 위해서는 인간의 개입과 통찰력이 필요한 것입니다.

## QA 전문가들에게 주는 축복과 도전

QA 산업의 미래는 매우 희망적입니다. AI의 도움으로 더 많은 일자리와 기회를 창출할 것입니다. 하지만 동시에 AI와의 협업과 경쟁에 대한 도전을 안고 있습니다. QA 전문가들은 계속해서 역량을 강화하고 지속적인 학습에 투자해야 합니다. AI 기술과 함께 발전하는 QA 전문가들은 가치 있는 일을 할 수 있을 것입니다.

출처 : [www.toolify.ai](http://www.toolify.ai)



소프트웨어의 복잡성이 고도화됨으로 인해  
AI가 전체 시스템의 품질을 보장할 수 없고,  
반복적이고 규칙적인 작업을 자동화하는 데는  
AI가 뛰어나지만 창의적이고 복잡한 문제를  
해결하는 데는 한계가 있다.

결국  
**품질을 보장하기 위해서는  
여전히 인간의 판단이 필요하고,  
또한 도메인에 대한 이해가 필요한  
전문 지식과 경험, 작업은  
인간을 대체할 수 없다.**

# | QA를 필요로 하는 분야



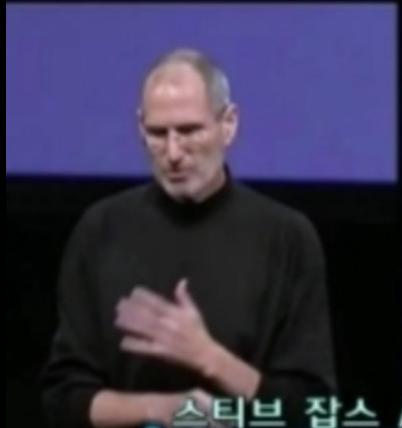
Hyundai Card



Google



# | 인문학과 연결된 품질



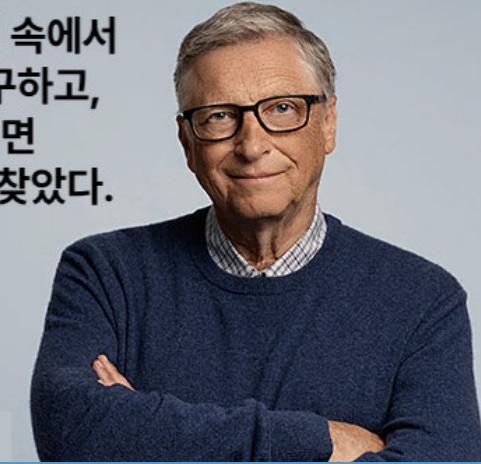
스티브 잡스 / 애플 CEO

우리가 창의적인 제품을 만든 비결은 우리는 항상  
기술과 인문학의 교차점에 있고자 했습니다

만약 과학을 어둠 속에서 무언가를 찾는 것에 비유하자면 인문학은 우리가  
그동안 지나온 길과 앞으로 닥쳐올 위험을 알려주는 촛불과 같습니다.  
**세상을 바꾸는 것은 사람 중심의 기술입니다.**

그래서 여러분이 하는 일의 중심에 사람을 먼저 둘 때 비로소 영향력을 만들  
어 낼 수 있습니다. 가치가 녹아있는 기술이 우리를 진보할 수 있게 합니다.  
(스티브 잡스 2017.6.9)

빌 게이츠는 인문 고전 속에서  
회사 경영의 해답을 구하고,  
모르는 것이 생기면  
도서관에서 해결책을 찾았다.



컴퓨터는 사람이 지시한 모든 작업을 할 수 있어요.  
만약 (공부를) 다시 시작한다면 가장 복잡한 분야가  
생물학이라는 점을 깨달았을 것 같아요.

**The End.**

# QnA

# | QA 지망생을 위한 TIP.

## TIP. QA 직군 지원자가 알아 두면 좋을 채용 면접 프레임

### ■ 커뮤니케이션

- 적극적으로 커뮤니케이션하고 업무를 리딩 할 수 있는 역량
- 열린 커뮤니케이션 : 문제 해결 능력, 원활한 협업

### ■ 역할

- 소프트웨어 테스팅과 담당하는 업무에 대한 기술적/경험적 지식
- 자발적인 자기주도성, 적극성, 자율성, 도전정신, 협업, 주체적 태도
- 논리적 사고와 문제 분석 및 해결 능력

### ■ 역량(스페셜리스트)

- 보유기술 & 강점
- 프로세스나 새로운 기술 등을 도입하거나 개선한 경험

### ■ 신뢰

- 지원하는 분야나 서비스에 대한 이해 수준
- 자신의 역할에 대한 이해 수준

### ■ 모티베이션 (개인과 조직의 목표달성을 위한 계획과 자발적 행동)

- 소프트웨어 품질과 품질을 관리하는 담당자로서 업무 가치관
- 향후 커리어를 위한 준비 및 계획

E.O.D  
**[brunch.co.kr/@swtestrecipe](https://brunch.co.kr/@swtestrecipe)**  
**ruth2460@naver.com**