

2013. 11. 11. [제73호]

테스트 자동화 실패를 극복할 수 있는 4가지 요인

소프트웨어공학센터 경영지원TF팀

C o n t e n t s

- ▶ 서론
- ▶ 1. 테스트 자동화 프로젝트는 소프트웨어 개발
- ▶ 2. 먼저 수동 테스트에 성공하기
- ▶ 3. 단순히 자동화가 추가된 테스트를 뜻하지 않음
- ▶ 4. 오토파일럿(autopilot)을 피하라

Key Message

테스트자동화 툴은 비용과 시간을 절감하여 SW품질향상에 기여하는 것으로 평가되어 왔으나 테스트 자동화가 모든 문제를 해결하는 만능 열쇠는 아님. 오히려 이를 과실했을 경우 품질관리 측면에서 프로젝트가 실패하는 경우가 발생하기도 함에 따라 이에 대한 요인을 분석하여 제공하고자 함.

▶ **서론**

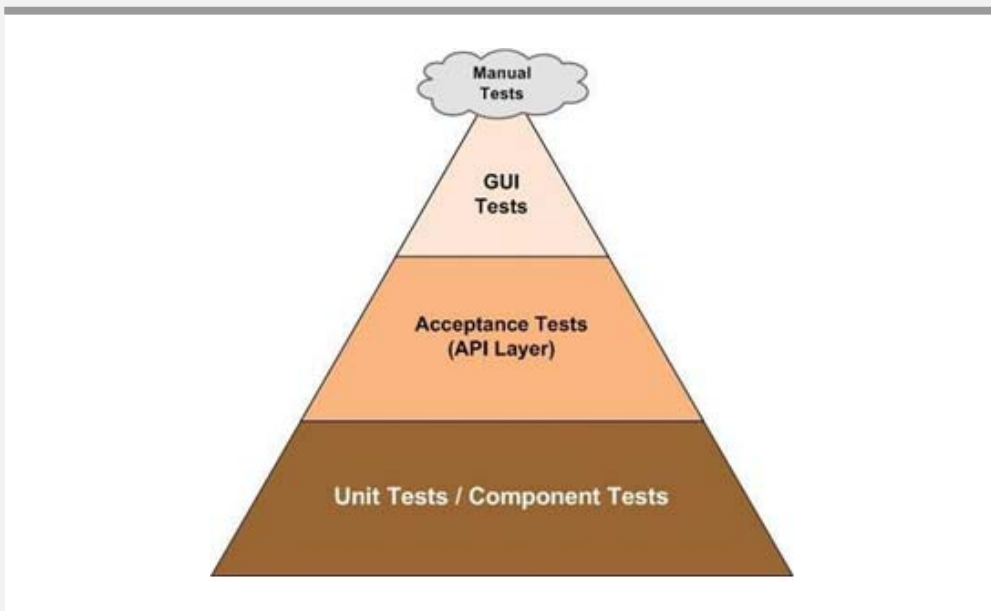
- 프로젝트 개발 조직이 테스트 자동화 툴을 도입하면 수작업으로 품질관리를 하는 것보다 미리 구성된 스크립트를 사용하기 때문에 당연히 시간과 비용이 절약된다고 생각함
- 단순히 테스트 자동화를 도입한다고 시간과 돈이 절약되고 소프트웨어 품질이 높아지는 것은 아님
- 테스트 자동화는 프로젝트 초기에 툴 선정과 테스트 스크립트를 만드는 단계가 필요하고, 테스트 수행자들은 코딩 기술을 보유하고 있어야 함
- 자동화의 도입은 피할 수 없는 흐름으로 특정 테스트들은 사람이 수동으로 하는 것보다 자동화테스팅이 더욱 효율적임
- 테스트 자동화를 사용하는 QA 관리자들의 실수를 줄이고, 그들의 준비상태를 평가하고, 현실적인 기대치를 설정하는 데 도움이 될 4가지 요인을 살펴봄

▶ **1. 테스트 자동화 프로젝트는 곧 소프트웨어 개발 프로젝트**

- 테스트 자동화 프로젝트는 곧 소프트웨어 개발 프로젝트라는 것일 인식하는 데 실패한다는 것은 곧 테스트 자동화에 실패한다는 것을 의미함
- 개발 프로젝트에 코딩이 필수적인 것과 같이 테스트 자동화 툴은 코드를 작성할 수 있는 팀 구성원 없이는 불가능함
 - 이처럼 테스트자동화는 코드 작성에 소프트웨어 개발과 같은 수준의 사고와 노력이 필요함
 - 마이크 코언(Mike Cohn)의 테스트 자동화 피라미드에 따르면, Unit Test와 Component Test가 코드 설계와 아키텍처를 검증할 수 있는 가장 기반이 되는 테

스트 프로세스로, 만약 팀이 테스트 자동화에 대한 준비가 없다면, 테스트 자동화를 학습하는 것으로부터 시작해야 함

그림 1_ 마이크 코언(Mike Cohn)의 테스트 자동화 피라미드



- 이것은 코딩 스킬이 부족한 테스터들에게는 중요한 장애로 부각되는데, 이를 극복하기 위해 테스터들은 Ruby와 같은 스크립트 언어를 마스터하려는 노력을 수반해야 함

▶ 2. 먼저 수동 테스트에 성공하기

- 테스트 프로젝트의 성공은 그 방법이 수동이나 자동의 여부보다 사용자에게 높은 품질의 소프트웨어를 전달하기 위한 사항들을 제대로 테스트하는 것에 달려 있음
- 무엇보다도 먼저 테스트에 전문성을 보유하지 않고서 자동화 테스트 툴을 활용한다는 것은 좋은 방법은 아님
- 테스트 전문가 Mike Kelly는 많은 조직들이 비용절감차원에서 테스트 자동화에 집중하는 경향이 짙는데, 이보다는 목적에 부합하는 테스트 프로젝트를 수행하기 위해서는 테스트 리스크와 범위 등에 대한 테크닉을 익히고 익숙해진 다음 자동화 테스트로 전환하는 것이 필요하다고 밝힘

▶ 3. 단순히 자동화가 추가된 테스트를 뜻하지 않음

- 일반적으로 테스트 자동화는 쉽다는 오해를 하지만 테스트 자동화는 수동테스트 보다 어렵다는 것을 우선 인식해야함
- 기존 테스트 프로세스에 단순히 자동화 테스트를 추가하는 것이 아니라 전반적으로 테스트 프로세스를 재설계한다고 생각해야 함
- 어떤 테스트가 자동화에 적합한지, 어떤 테스트를 수동으로 남겨둘 것인지에 대해 창의적으로 생각(creative thinking)이 요구됨
- 마이크로소프트의 SW공학전문가 존 오버버그(John Overbaugh)는에 의하면 테스트 자동화를 고려하는 측면을 크게 두 가지로 제시함
 - 첫 번째는 , 성능 테스트 수행시 수백 수천건의 사용자 접속 시뮬레이션을 수행하여 다양한 테스트 케이스를 생성해야하는 등 수동적으로 테스트하는 것이 매우 복잡하고 시간소모적일 경우임
 - 두 번째는 지속통합개발(CI)시 하루종일 또는 계속적으로 반복테스트가 요구되는 빌드검증테스트(BVT) 등의 경우 테스트 자동화가 우선적으로 요구됨
- 포춘 100대 기업의 SW개발 책임자인 켈빈 헨니(Kevlin Henney)에 따르면, 테스트 자동화를 고려할 수밖에 없는 부분이 존재한다고 밝힘
 - 우선적으로는 애플리케이션 품질 전체를 아우르는 거대 규모의 부하 테스트와 테스터가 일일이 대응하기 어려운(또는 귀찮은) 작은 규모의 빈번한 테스트 등이 자동화테스팅으로 우선적으로 고려됨

▶ 4. 오토파일럿(autopilot)을 피하라

- 테스트 팀이 테스트 자동화의 도입 초기를 성공적으로 마치면, 같은 테스트를 반복적으로 수행하는 경향이 있는데 이는 큰 실수라 할 수 있음
- 많은 테스트를 빨리 수행한다고 소프트웨어의 품질이 높아지는 것은 아님
- 품질수준이 높은 소프트웨어를 얻기 위해서는 적절한 테스트들을 수행하고 이 테스트들이 적절한 것인지를 지속적으로 재평가해야 함
- 최고의 소프트웨어를 만드는데 필요한 테스트가 무엇인지를 계속 생각하고 그에 적

합한 스크립트를 만들어야 함

- 위와 같은 방법을 통해 테스트 자동화 툴을 성공적으로 도입할 수 있음
- 자동화방식으로 과거의 테스트를 반복적으로 수행하기만 한다면 향후에 문제가 생길 가능성이 많음으로 인해 바람직한 방식이 아님을 다시 한번 밝힘

참고 자료

1. <http://www.base36.com/2013/03/automated-vs-manual-testing-the-pros-and-cons-of-each/>
2. <http://blogpost110.wordpress.com/2013/07/25/software-testing-automation/>
3. <http://www.softwaretestingclub.com/forum/topics/why-how-when-to-do-test-automation?commentId=751045%3AComment%3A180020>