

SETP(Simple and Easy Test Process) 핵심 모델과 자동화 도구 Legba

박유봉⁰ 서주영 최병주
이화여자대학교 컴퓨터학과
(youbong, jyseo, bjchoi)@ewha.ac.kr

SETP(Simple and Easy Test Process) Core Model and Automation Tool Legba

Youbong Park⁰ Jooyoung Seo Byoungju Choi
Dept. of Computer Science & Engineering, Ewha Womans University

요 약

소프트웨어 개발 프로세스의 평가를 통해 소프트웨어 프로덕트의 품질을 높여려는 다양한 시도들이 행해지고 있다. 소프트웨어 테스트 프로세스는 소프트웨어 개발 프로세스 중에서도 소프트웨어 프로덕트의 품질 향상에 직접적인 영향을 끼치므로, 이에 대한 체계적인 개선이 필요하다. 본 논문에서는 프로세스의 품질과 프로덕트의 품질을 동시에 고려하는 소프트웨어 테스트 모델인 SETP 핵심 모델과 이를 지원하는 자동화 도구(Legba)를 제안한다.

1. 서론

소프트웨어 품질에 대한 요구가 늘어나면서 소프트웨어 개발 프로세스의 평가를 통해 소프트웨어 프로덕트 품질을 보증하려는 다양한 평가 방법론에 대한 관심이 높아져가고 있다. 대표적으로 CMU SEI의 CMMI[1]와 ISO의 SPICE[2]가 프로세스 개선을 통한 프로덕트 품질 향상을 목적으로 하는 평가 방법론이다.

국내의 소프트웨어 관련 산업체에서 CMMI 혹은 SPICE 에 대한 인증 획득에 적극적이며, 인증을 통해 조직 내에 소프트웨어 개발 프로세스가 정착되고 있다. 하지만 조직의 표준 소프트웨어 개발 프로세스가 직접적으로 소프트웨어 프로덕트의 품질 향상으로 연결되는 데에는 어려움을 겪고 있으며, 이는 실질적으로 프로덕트 품질과 직결되는 테스트 프로세스에 대한 체계적인 개선과 기술 지원이 부족하기 때문이다.

CMMI의 경우 Verification 프로세스 영역과 Validation 프로세스 영역에서 테스트 프로세스와 관련하여 일부 언급하고 있으며, 미국 일리노이 공대의 TMM[3]을 중심으로 TIM[4], TOM[5], TPI[6] 등의 경우 테스트 활동 자체에 중점을 둔 모델들이다.

본 논문에서는 테스트 프로세스 개선을 위해 테스트 프로세스 뿐 아니라 테스트 조직, 테스트 인프라, 결합 관리와 품질 관리와 같은 프로세스 영역을 갖는 SETP 핵심 모델과 이를 지원하는 자동화 도구인 SETP 자동화 도구(Legba)를 소개하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 테스트 프로세스 개선을 위한 SETP 핵심 모델을 기술하고 3장에서는 이를 지원하는 SETP 자동화 도구로 Legba를 기술하며 마지막으로 4장에서 결론을 맺는다.

SETP 핵심 모델은 프로세스 품질과 프로덕트 품질을 동시에 고려하는 소프트웨어 테스트를 위한 모델로써, 간결하고(Simple), 실용적이어서, 적용하기 쉽도록(Easy to Apply)함을 목적으로 하는 테스트 프로세스 개선을 위한 모델이다. 이 모델은 소프트웨어 개발의 각 단계에서 수행해야 하는 테스트 활동을 위한 지침이 된다.

일반적으로 신뢰성, 유지보수성, 사용성, 테스트 가능성, 재사용성, 정확성 등의 평가로 인증 되는 소프트웨어 프로덕트 품질은 주로 개발이 완료된 후에 측정되어 왔지만, 사실 소프트웨어 개발 생명주기의 각 단계에서 품질을 제어하기 위한 방안으로도 측정 가능하다. 소프트웨어 프로덕트를 개발하는 동안 각 개발 단계에서 산출된 작업산출물을 Verification하고 검증 Validation하는 등의 품질 확보를 위한 프로세스를 체계적으로 정의하고, 프로덕트 품질에 직접적으로 영향을 미치는 작업 산출물을 식별하고 보장할 수 있는 지침을 제공하는 것이 SETP 핵심 모델의 역할이다.

다음은 조직이 갖는 테스트 프로세스 성숙도 수준을 테스트 인식이 부재한 초기 단계부터 이상적인 V 모형으로의 성숙되는 과정으로 표현한 것이다.

(1) 성숙도 수준 1: Initial

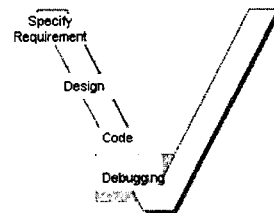


그림 1 성숙도 수준 1

테스트 프로세스나 독립된 테스트 조직이 존재하지 않으며,

2. SETP 핵심 모델 (Simple and Easy Test Process Core Model)

(2) SETP Generator

사용자의 역할과 관점, 제품의 도메인 등의 입력 정보와 SETP 핵심 모델을 바탕으로 간결하고(Simple), 적용하기 쉬운(Easy to apply) 테스트 프로세스를 생성한다.

(3) SETP Evaluator

생성된 테스트 프로세스의 각 단계별로 평가 지표를 설정한다. SETP Evaluator는 설정된 평가 지표의 수행 여부와 상태정보를 데이터베이스에 저장하고, 전체적인 테스트 프로세스 이행 상황을 모니터링한다.

(4) Quality Predictor (구현 예정)

품질변수를 바탕으로 최종 산출물의 품질을 예측한다.

3.2 SETP 생성

사용자의 역할과 대상 제품에 최적인 테스트 프로세스를 생성하기 위해서, 사용자는 SETP Viewer에 프로세스 개선 그룹, 테스트 전문가 그룹, 소프트웨어 개발 그룹, 테스트 기술 그룹 등의 해당 역할과 관점을 입력한다. SETP Viewer는 사용자 정보와 SETP 핵심 모델에 기반하여 적합한 테스트 프로세스를 추천한다. 사용자는 자신이 목표로 하는 성숙도 수준에 해당하는 테스트 프로세스 영역과 해당 영역에서의 활동들에 대한 정보를 인지하게 된다. 또한 개발 생명 주기 상에서 사용자가 수행해야 할 테스트 프로세스 영역과 활동들을 그림 7과 같이 제시한다.

사용자는 전체 개발 주기 내에서 테스트 활동을 수행해야 하는 시점과 방법에 관한 정보를 제공 받는다. SETP Generator는 이러한 정보를 바탕으로 개발 생명주기 내에서의 테스트 프로세스를 제안한다. 사용자가 역할과 관점 외에 도메인 정보를 추가로 입력하면 SETP Generator는 개발 생명 주기의 각 단계별 테스트 활동과 산출물들을 제공하는 테스트 프로세스를 생성하고 데이터베이스에 저장한다.

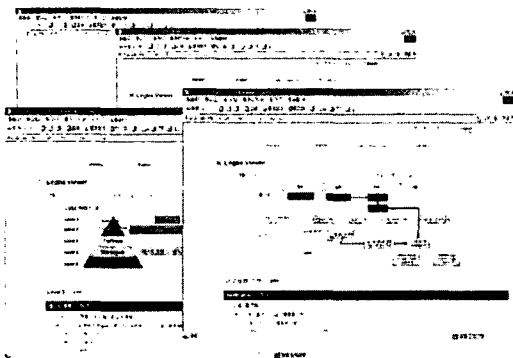


그림 7 SETP의 생성

3.3 SETP 평가

SETP Evaluator는 SETP Generator를 통해 생성된 테스트 프로세스 내에 각 개발 단계에서 Verification하고 Validation 해야 할 평가 지표들을 설정한다. 평가 지표들은 해당 개발 단계에서 필수적으로 수행해야 하는 테스트 활동들과 산출물들을 의미한다. 개발자들은 이를 통해 미리 자신이 수행

한 테스트 활동들을 각 개발 단계의 종료 시 마다 평가함으로써, 마지막 개발 단계의 시스템 테스트에만 지나치게 집중하거나 일정에 쫓겨 테스트 수행을 소홀히 하기 쉬운 기존 체계의 문제점을 해결할 수 있다는 의의를 갖는다.

SETP Evaluator는 소프트웨어 프로덕트의 전 개발 주기에 걸쳐 설정된 평가 지표들의 이행 여부와 진행 상태를 지속적으로 모니터링하여 결과적으로 우수한 품질의 프로덕트를 생산하는데 기여하게 된다.

4. 결론

본 논문에서는 실질적으로 프로덕트 품질과 직결되는 테스트 프로세스에 대한 체계적인 개선을 위해 SETP 핵심 모델과 SETP 자동화 도구인 Legba를 제안하였다.

SETP 핵심 모델은 프로세스 품질과 프로덕트 품질을 동시에 고려하는 소프트웨어 테스팅을 위한 모델로써, 5단계의 테스트 프로세스 성숙도 수준에 따른 테스트 프로세스 영역으로 구성되고 소프트웨어 개발의 각 단계에서 수행해야 하는 테스팅 활동을 위한 지침이 된다. 본 모델은 삼성전자 8개 사업부의 테스트 수행능력 실태조사에 평가모델로써 적용되었다.

Legba는 SETP 핵심모델의 실질적인 적용을 위해 구현한 자동화 도구이며, 다양한 역할을 가진 사용자들의 관점과 도메인 정보 별로 차별화된 테스트 프로세스의 생성과 생성된 테스트 프로세스의 각 개발 단계 내에서 필수적인 테스트 활동에 대한 평가 지표의 정의와 추적을 지원한다.

향후 연구 과제로써 SETP 핵심 모델의 실제적인 적용을 바탕으로 하는 모델의 검증 및 정제와 Legba의 평가 기능 보완 및 평가를 통해 품질을 예측할 수 있는 Quality Predictor의 구현을 수행하여야 할 것이다.

6. 참고문헌

[1] CMMI-SE/SW/IPPD/SS, V1.1
 [2] ISO/IEC TR 15504
 [3] I. Burnstein, A. Homyen, T. Suwannasart, G. Saxena, and R.Grom, "A Testing Maturity Model for Software Test Process Assessment and Improvement", Software Quality Professional, 1999.
 [4] T. Ericson, A. Subotic, S. Ursing, "TIM - A Test Improvement Model", Software Testing Verification and Reliability 7(4), pp.220-246
 [5] <http://www.evolutif.co.uk/tom/oeverview.html>
 [6] Kooman, Tim and Pol, Martin, "Test Process Improvement", Addison-Wesley, ISBN 0-201-59624-5