
품질검증 분야

Stack 통합 Test 결과보고서

[TestLink]

2014. 09.

목 차

I. Stack 통합 테스트 개요	1
1. 목적	1
II. 테스트 대상 소개	2
1. TestLink 소개	2
2. TestLink 주요기능 및 특징	3
3. TestLink 시스템 요구사항	4
III. Stack 통합 테스트	5
1. 테스트 환경	5
2. 테스트 접근 방법	6
3. 테스트 수행	8
4. 테스트 수행 결과	9
VI. 종합	10
※ 참고자료	11
[별첨1] TestLink 테스트 케이스	

I. Stack 통합 테스트 개요

공개SW Stack 통합테스트는 여러 공개SW들의 조합으로 시스템 Stack을 구성한 후 Stack을 구성하는 공개SW의 상호 운용성에 중점을 두고 기능 및 성능테스트 시나리오를 개발하여 테스트를 진행한다.

본 통합테스트를 통해 안정된 Stack 정보를 제공하여 민간 및 공공 정보시스템 도입 시 활용될 수 있도록 한다.

1. 목적

□ 공개SW Stack 통합 테스트 수행 목적

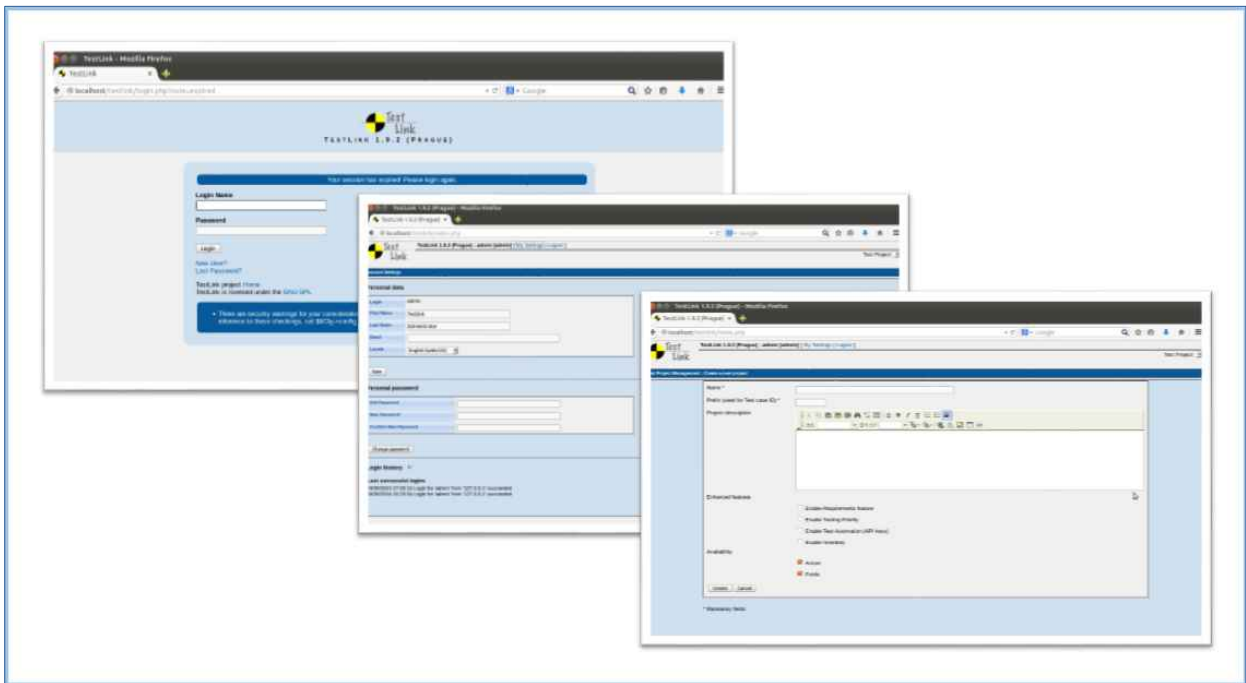
- 공개SW로 구성된 Stack이 유기적으로 잘 동작함을 확인
- 다양한 Stack 구성에 기반을 둔 테스트를 통해 안정된 Stack 조합 규명
- 공개SW 시스템 도입을 위한 Stack 참조모델의 신뢰성 정보로 활용
- 공개SW의 신뢰성과 범용성에 대한 사용자 인식 제고

II. 테스트 대상 소개

1. TestLink 소개

TestLink는 오픈소스 웹 기반으로 개발된 프로젝트 관리 시스템으로 대상 시스템의 개발 프로세스에 따른 체계적인 테스트 관리 환경을 제공한다. TestLink는 APM(apache, php, mysql) 구성 환경에서 적용 및 실행 가능하며, 서버 단에서 다양한 OS환경(Linux, Windows 및 Mac OS X 등)을 지원한다.

현재 TestLink는 GPL(GNU General Public license) 라이선스 적용을 받으며, 테스트 서버 구축을 통해 Test Case 관리 자동화 도구로 널리 사용되고 있다.



[그림 1. TestLink 기본 화면 구성]

2. TestLink 주요기능 및 특징

□ TestCase 관리 자동화

TestLink는 테스트 케이스와 요구 사항의 맵핑, 테스트 케이스의 시나리오 작성, 기대결과 (Expected Result) 및 테스트 결과 관리, Version 별 릴리즈에 따른 테스트 계획과 결과를 통합 관리할 수 있는 웹 관리 환경을 제공한다.

□ Test Link 기능 카테고리

- Test Project 생성
- Test Specification 작성
- Test Plan 작성
- Build Version 생성
- Test Suite/Case Assign
- Test Execution 및 Reporting

□ Test Link 특징

- Web 환경의 인터페이스를 통해 사용자는 인스톨 없이 어디서든 접근 가능
- 테스트의 우선순위 부여 기능 및 사용자의 활동영역 정의 (리더, 테스터 등)
- 다양한 종류의 DB를 지원 (MySQL, PostgreSQL, MS-SQL 등)
- 트리 구조를 이용한 테스트 케이스 작성 기능 제공
- 다양한 언어 지원 (English, French, German, Italian, Spanish 등)
- 다양한 버그 시스템들과 직접 연동 가능 (Bugzilla, Mantis, Jira, TrackPlus, Eventum, Trac, Seapine)
- 다양한 형식의 보고서 제공 (HTML, MS-word, MS-Excel) 및

E-mail에 직접 보고서를 보낼 수 있는 기능 제공

3. TestLink 시스템 요구사항

□ TestLink 시스템 요구사항

항목	정보
OS	- 모든 OS 지원
Java JDK	- 4.1.1 이상
PHP5	- 4.3.0 이상
Web-Server	- Apache 1.3.x or 2.0.x and higher, IIS 3 and higher, etc.
DB	- MySQL 4.0.x 이상

[표 1. TestLink 시스템 요구사항]

※ 추가적인 자세한 정보는 아래의 링크 정보 참조

[1] SW배포

- <http://sourceforge.net/projects/testlink/files/>

[2] 설치가이드

- <http://testlink.sourceforge.net/docs/documents/installation.html>

<http://informatiandocr.blogspot.kr/2013/01/how-to-install-and-configure-testlink.html>

III. Stack 통합 테스트

1. 테스트 환경

TestLink SW 환경

SW	Version
TestLink	1.2.10
MySQL	6.2
PHP	5.3.10-1ubuntu3.13
Apache	2.2.22

[표 2-1. 테스트 SW 환경]

Stack 환경

Stack	OS	네트워크 정보 (IP)
A (Server)	Ubuntu 12.04 LTS	121.162.249.93

[표 2-2. Stack 환경]

HW 환경

제조사	모델명	CPU	MEM	Disk	NIC
HP	dc7900 CMT	Quad-Core 2.66Ghz~4P	2.8GiB	265GB	Gigabit 1Port

[표 2-3. HW 환경]

※ 참고 - 동일 HW 사양의 PC 2대에서 Sever 및 User 사용 환경을 구축함.

2. 테스트 접근 방법

TestLink는 테스트 서버 구성을 통해 소프트웨어 개발 프로세스에 따른 검증 영역에 대해 테스트 관리 환경을 제공한다. 따라서 실제 작업 환경인 웹 기반으로 구현된 사용자 환경의 기능 신뢰성이 중요하므로 제공되는 각 주요 기능 카테고리 별 기능 동작을 확인하는데 중점을 두고 테스트를 진행하였다.

본 테스트에서는 리스크 분석을 통해 TestLink의 기능 및 비기능 항목들에 대한 기능적/ 기술적 아이템을 정의하였다. 또한, 각 정의된 아이템 별 사용자 시나리오를 바탕으로 테스트 아이디어를 도출하였다.

□ 탐색적 테스트(Exploratory Testing)

탐색적 테스트는 테스트 엔지니어의 지적 능력을 최대한 공유, 활용하는 것을 목적으로 하는 테스트 접근법으로 테스트를 수행할 대상을 실행시켜 사용함과 동시에 사용 측면에서 문제가 되는 부분에 집중하여 테스트를 설계 및 계획한다.

이러한 과정은 효율적 진행을 위한 Time Boxing을 통해 수행되므로 테스트 케이스 작성을 최소화할 수 있고, 상대적으로 적은 시간에 집중적인 테스트를 가능하게 한다.

□ 리스크 기반 테스트(Risk based Testing)

리스크 기반 테스트 기법은 테스트 대상에 비해서 테스트 자원이 부족한 경우 효과적이고, 효율적인 테스트 수행을 위해 적용 될 수 있다. 해당 기법은 크게 리스크 식별과 리스크 분석, 그리고 리스크 계획의 세 단계로 구분 진행된다.

리스크 식별 단계에서는 제품 품질관점에서 테스트 대상이 될 항목을

식별하고, 프로젝트나 제품에 대한 리스크 요소를 식별한다.

리스크 분석 단계에서는 장애 발생가능성과 장애로 인한 영향을 식별하고 리스크 우선순위를 결정한다.

마지막으로 리스크 계획 단계에서는 리스크의 우선순위에 따른 대처 방안 및 완화 정책을 수립하며, 이후 테스트 수행 시 커버리지를 고려하여 선택과 집중을 통해 테스트를 수행하게 된다.

□ 시나리오 테스트

시나리오 테스트 기법은 단일 기능에 대한 결함 여부를 확인하는 것이 아니라, 서로 다른 컴포넌트 사이의 상호작용과 간섭으로 발생할 수 있는 결함을 발견하기 위한 기법이다.

3. 테스트 수행

기능 테스트 수행 관련 세부 케이스는 별첨 「TestLink 테스트케이스」 문서를 참고한다.

□ 탐색적 테스트 현황

이하, 테스트 현황의 차터 항목 구분은 각 기능 아이템 항목 별 세부 구분 항목으로서 사용자 시나리오를 기반으로 정의되었다.

기능 아이템	기본 차터	테스트 아이디어
Test Project 관리	6	21
User 관리	2	7
Requirement Specification 관리	6	19
Test Specification 관리	3	20
Test Plan 관리	1	11
Test Execution 관리	1	4
Test Plan contents 관리	2	5
합 계	21	87

[표 3. 테스트 아이디어 현황]

- Test Project 관리 카테고리에서는 Project 정보 관리 및 User 별 역할 할당과 Platform, Inventory 설정 관련 기능을 중심으로 테스트 진행.
- User 관리 카테고리에서는 사용자 정보의 등록 및 등록된 정보의 관리 기능을 중심으로 테스트 진행.
- Requirement Specification 관리 카테고리에서는 요구 명세 정보의 생성 및 수정과 설정된 정보의 조회관리 기능을 중심으로 테스트 진행.
- Test Specification 관리 카테고리에서는 Test Case 정보의 생성 및 등록된 Test Case 정보의 관리 기능을 중심으로 테스트 진행.
- Test Plan 관리 카테고리에서는 Test Plan 수립 및 빌드된 Plan 정보의

- 사용자 별 할당과 MileStone 관리 기능을 중심으로 테스트 진행.
- Test Execution 관리 카테고리에서는 테스트 수행 정보의 등록 및 변경 처리와 User 별 할당된 TestCase 정보의 모니터링 기능을 중심으로 테스트 진행.
 - Platform 정보 관리 카테고리에서는 등록된 해당 정보의 Test Plan 반영처리와 Test Suit, Test Case, Test Plan 정보 별 모니터링 관리 및 업데이트 기능을 중심으로 테스트 진행.

4. 테스트 수행 결과

TestLink 테스트를 수행한 결과 기능 카테고리 별 정리된 내용은 아래와 같다.

분류		PASS	FAIL	N/T	N/A
기능	테스트 아이디어				
Test Project 관리	21	21	0	0	0
User 관리	7	7	0	0	0
Requirement Specification 관리	19	19	0	0	0
Test Specification 관리	20	20	0	0	0
Test Plan 관리	11	11	0	0	0
Test Execution 관리	4	4	0	0	0
Test Plan contents 관리	5	5	0	0	0

[표 4. 테스트 결과]

- TestLink로 테스트 서버 환경을 구축하고 서버와 구분된 PC 환경에서 웹 브라우저 UI 접속을 하여 테스트 수행한 결과, 기능 검증 대상인 Test Project, Test Specification, Test Plan, Test Execution 카테고리 별 항목들에 대하여 주요 기능 관련한 심각한 오류 및 치명적 장애 이슈는 발견되지 않음.

VI. 종합

- TestLink 테스트 수행 결과 공개 SW로 구성된 Stack 환경에서 치명적 결함 이슈 발생 없이 Test Case 관리 자동화 시스템의 제반 기능들이 각 기능 resource들과 유기적으로 동작함을 확인하였다.
- TestLink는 Selenium, SOAP UI와 같은 테스트 도구뿐만 아니라 JUnit과 같은 다양한 테스트 프레임워크와 연동이 가능하다. 기존 테스트 도구만 사용했을 때와 달리 전체 테스트 현황이나 History, 상세한 테스트 케이스 특히 JUnit과 같은 코드가 아니라 Human readable한 형태로 테스트 케이스를 관리할 수 있게 해준다.
또한, 요구사항에서부터 Test Case 및 결과까지의 추적성을 보장해 준다.
- TestLink는 Test Case Management System으로서 프로젝트 별 역할에 따른 User의 권한설정이 가능하고, 테스트 명세서 관리와 계획, 리포팅, 요구사항 트래킹 기능을 제공하여 테스트 서버를 구성하는 데 유용한 솔루션이다.

※ 참고 자료

- [1] TestLink 공식 사이트
 - <http://www.TestLink.org/>
- [2] 포럼/ 커뮤니티
 - <http://forum.testlink.org/>
 - <https://twitter.com/TLOpenSource>
- [3] OSI 프로젝트 라이선스 정보
 - <http://testlinkjavaapi.sourceforge.net/license.html>
- [4] 배경 기술/ 도메인 정보
 - <http://softwaretestingstandard.org/part2.php>