

# 사용자 매뉴얼

TestLink version 1.7

**Translated by JIN SJ**

© 2004 - 2008 TestLink Community

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.2 published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. The license is available in "[GNU Free Documentation License](#)" homepage.

<b>1 일반 정보</b>	4
1.1 전체 구조	4
1.2 기본 용어	4
1.3 간단한 테스트링크 워크 플로우 예제	5
<b>2 테스트 프로젝트</b>	7
2.1 새 테스트 프로젝트 생성	7
2.2 테스트 프로젝트 편집 및 삭제	7
<b>3 테스트 명세</b>	8
3.1 테스트 케이스 스위트	8
3.2 테스트 케이스	8
3.3 키워드	10
<b>4 요구 기반 테스트</b>	11
4.1 활용성	11
4.2 요구사항 명세	11
4.3 요구사항	12
<b>5 테스트 플랜</b>	14
5.1 테스트 플랜 생성 및 삭제	14
5.2 빌드	14
5.3 테스트 플랜 활성화하기 - 테스트 케이스 추가하기	14
5.4 테스트 실행 할당	16
5.5 우선순위	16
5.6 마일스톤	16
<b>6 테스트 실행</b>	18
6.1 일반	18
6.2 네비게이션	18
6.3 실행	19
<b>7 커스텀 필드</b>	21
<b>8 테스트 보고서와 매트릭스</b>	22
8.1 일반 테스트 플랜 매트릭스	22
8.2 전체 빌드 현황	23
8.3 쿼리 매트릭스	23
8.4 Blocked, Failed, and Not Run Test Case Reports	24
8.5 테스트 보고	24

8.6 차트.....	24
8.7 각 테스트 케이스에서 발생한 총 버그.....	25
8.8 요구사항 기반 보고서.....	25
8.9 어떻게 새 보고서를 추가할 것인가.....	26
<b>9 사용자 관리.....</b>	<b>27</b>
9.1 계정 셋팅.....	27
9.2 역할 승인.....	27
9.3 권한 설정.....	29
9.4 테스트 플랜 할당.....	29
<b>10 데이터 가져오기와 내보내기 .....</b>	<b>31</b>
10.1 키워드 가져오기/내보내기.....	31
10.2 테스트 프로젝트 내보내기/가져오기.....	32
10.3 테스트 스위트 가져오기/내보내기.....	33
10.4 단 하나의 테스트 케이스.....	34
10.5 테스트 스위트의 모든 테스트 케이스.....	35
10.6 소프트웨어 요구사항 가져오기/내보내기.....	35
10.7 XML 을 통해 엑셀로부터 테스트 케이스 가져오기.....	36

## 1. 일반 정보

테스트링크는 웹에 기반한 테스트 관리 시스템이다. 이 매뉴얼은 사용자가 테스트링크에 사용되는 프로세스, 용어 및 동작 구조를 이해하는데 도움을 줄 자료로 활용될 수 있을 것이다. 시스템 요구사항, 설치 과정 및 구성에 관해 더 자세한 정보를 얻기 원한다면 설치 매뉴얼을 참조하라. [www.teamst.org](http://www.teamst.org) 혹은 [www.testlink.sourceforge.net](http://www.testlink.sourceforge.net) 에서 가장 최신의 문서들을 얻을 수 있다.

### 1.1 전체 구조

테스트링크에는 세 가지 기본적인 초석이 있다: **테스트 프로젝트**, **테스트 플랜** 그리고 **사용자**가 그것이다. 이 문서의 다른 모든 자료들이 이들과 연관이 있거나 이 항목들의 속성으로 볼 수 있다. 무엇보다 먼저, 우리는 다양한 문서(Documentation)와 테스트 업계에 전반적으로 사용되는 단어들을 우선 정의해볼 필요가 있다.

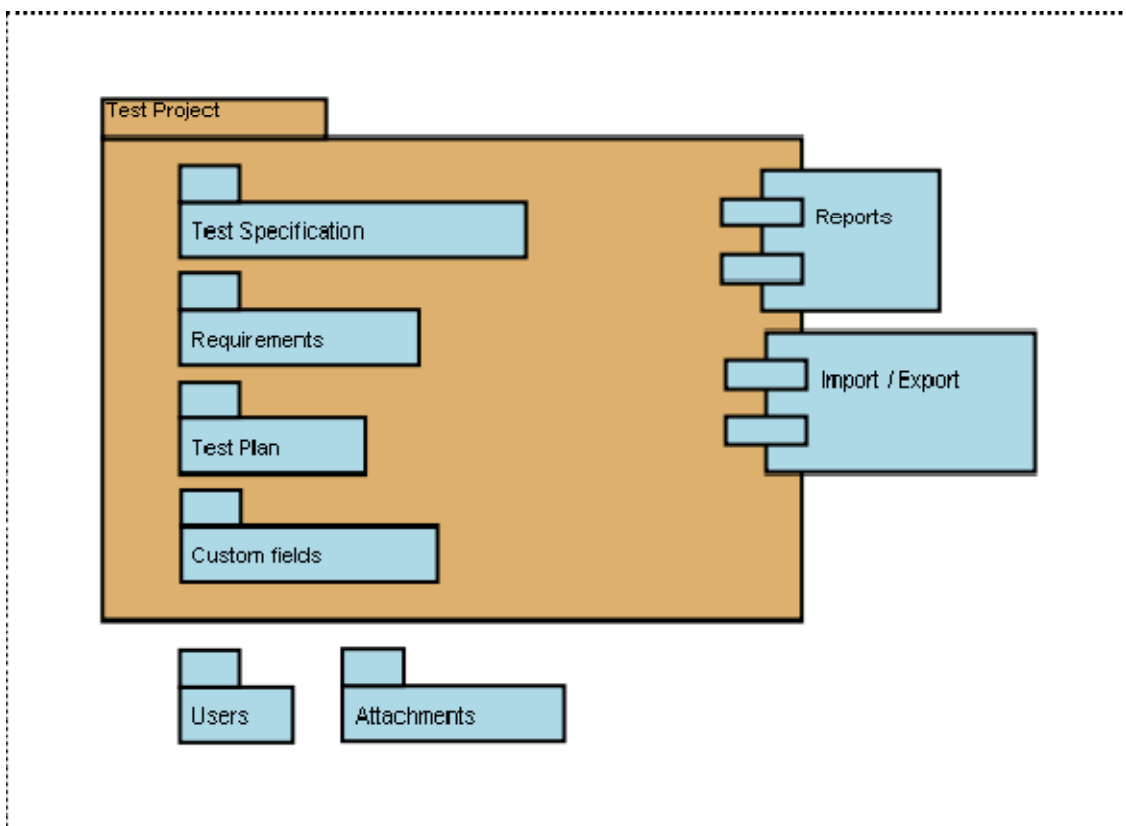


그림 1: 테스트 프로젝트는 테스트링크의 기본적인 컴포넌트이다.

### 1.2 기본 용어

**테스트 케이스**는 각 단계별 행동(액션, 시나리오)과 기대 결과로 이루어지는 테스트 업무를 칭한다. 테스트 케이스는 테스트링크의 기본적인 구성요소이다.

**테스트 스위트**(테스트 케이스 스위트)는 테스트 케이스를 일정한 단위(Unit)로 묶어준다. 이것은 테스트 명세를 논리적인 부분(Logical parts)으로 구성해준다.

*테스트 스위트는 TL 1.6 과 그 이전 버전의 컴포넌트와 카테고리 개념을 대체하는 것이다.*

**테스트 플랜**은 당신이 테스트 케이스를 실행하길 원할 때 생성된다. 테스트 플랜은 하나 혹은 그 이상의 테



스트 프로젝트들에 포함되어 있는 테스트 케이스로 만들어 질 수 있다. 테스트 플랜은 빌드, 마일스톤, 사용자 할당과 테스트 보고서를 포함한다.

**테스트 프로젝트**는 테스트링크에서 거의 영구불변한 존재이다. 테스트 프로젝트는 전체 라이프타임을 걸쳐 서로 다른 버전이 각기 실행될 것이다. 테스트 프로젝트는 테스트 케이스, 요구사항 및 키워드와 함께 테스트 명세를 포함한다. 프로젝트에 포함된 인원들은 정의된 역할을 가지고 있어야 한다.

*테스트 프로젝트는 TL1.6과 그 이전 버전에서는 **Product**<sup>1</sup>이라고 불리어졌다.*

**사용자:** 각각의 테스트 링크 사용자는 테스트링크의 어떤 기능들을 활용할 수 있는지를 정의한 역할을 가지고 있다. 좀 더 자세한 사항은 “사용자 관리” 장을 참조하라. 그림 2는 각각의 사용자 역할에 따른 일반적인 행위들을 보여주고 있다.

### 1.3 간단한 테스트링크 워크 플로우 예제

1. 관리자가 **테스트 프로젝트** “Fast Food”와 두 명의 사용자, “리더” 권한을 가진 아담(Adam)과 “상급 테스터” 권한을 가진 벨라(Bela)를 생성한다.
2. 리더 아담이 **소프트웨어 요구사항**을 외부에서 가져오고(Import) 이러한 요구 사항들을 파트 별로 구현하기 위해 새로운 테스트 케이스를 생성한다. 그는 테스트 케이스를 다시 두 개의 테스트 스위트: “Fish”와 “Chips”로 재구성한다.
3. 테스터 벨라는 테스트 스위트로 구성된 **테스트 명세**를 사용하는 테스트 시나리오(새로운 테스트 케이스의 내용물을 만드는 것)를 작성한다.
4. 아담은 키워드 “리그레션 테스트”를 생성하고 이 키워드를 테스트 케이스 중 10개에 할당한다.
5. 아담이 **테스트 플랜** “Fish & Chips 1”, **빌드** “Fish 0.1”을 생성하고 테스트 스위트 “Fish”안의 모든 테스트 케이스를 이 테스트 플랜에 연결한다. 그는 스스로와 벨라를 이 테스트 플랜에 인적자원으로 할당한다.
6. 이제 개발자들이 첫 빌드를 만든다. 아담과 벨라는 테스트를 실행하고 그 결과를 기록한다: 5개가 Pass되고, 한 개가 Fail 처리되었으며, 4개가 Block 처리된다.
7. 개발자들이 새로운 빌드 “Fish 0.2”를 만들고 벨라가 Fail 처리된 테스트 케이스와 Block 처리된 테스트 케이스만을 다시 테스트한다. 이번에는 Fail 처리된 케이스와 Block 처리된 케이스가 모두 Pass 처리된다. 그들은 또한 “리그레션 테스트” 키워드가 할당된 모든 테스트 케이스를 추가로 실행한다.
8. 팀의 **매니저**가 결과를 보고 싶어 할 수도 있다. 관리자는 그 스스로 계정을 만들고 로그인 할 수 있다고 설명해준다. 매니저가 계정을 생성한다. 그는 기본적인 “게스트” 권한을 가지고 **테스트 결과**와 테스트 케이스를 열람할 수 있다. 그는 또한 모든 보고서에 걸쳐 Pass 처리된 모든 테스트 케이스를 볼 수 있고 “Fish 0.1” 빌드에서 발생한 문제들과 같은 특정 빌드에서 발생한 문제점도 보고서를 통해 볼 수 있다.<sup>2</sup>
9. 이후, 개발자들이 역시 마지막으로 “Chips” 기능을 추가한다. 아담은 **테스트 플랜** “Fish & Chips 2”를 만든다. 그는 첫 번째 테스트 플랜을 템플릿으로 활용할 수 있다. 모든 “Fish” 테스트 케이스와 역할이 자동적으로 추가된다. 그는 새로운 빌드 “Fish 1.1”을 만들고 “Chips”의 모든 테스트 케이스를 이 테스트 플랜에 연결한다.

<sup>1</sup> 일반적인 공산품이나 제품을 뜻하는 것이 아님.

<sup>2</sup> 매니저는 열람만 가능하고 아무것도 편집할 수 없다.

10. 이제 다시 평소처럼 테스트를 시작한다.

11. 이후, 관리자는 다른 제품 “Hot Dog”을 위해 새 테스트 프로젝트를 생성한다. 그러나 이것은 다른 팀의 일이고 또 다른 얘기이다.

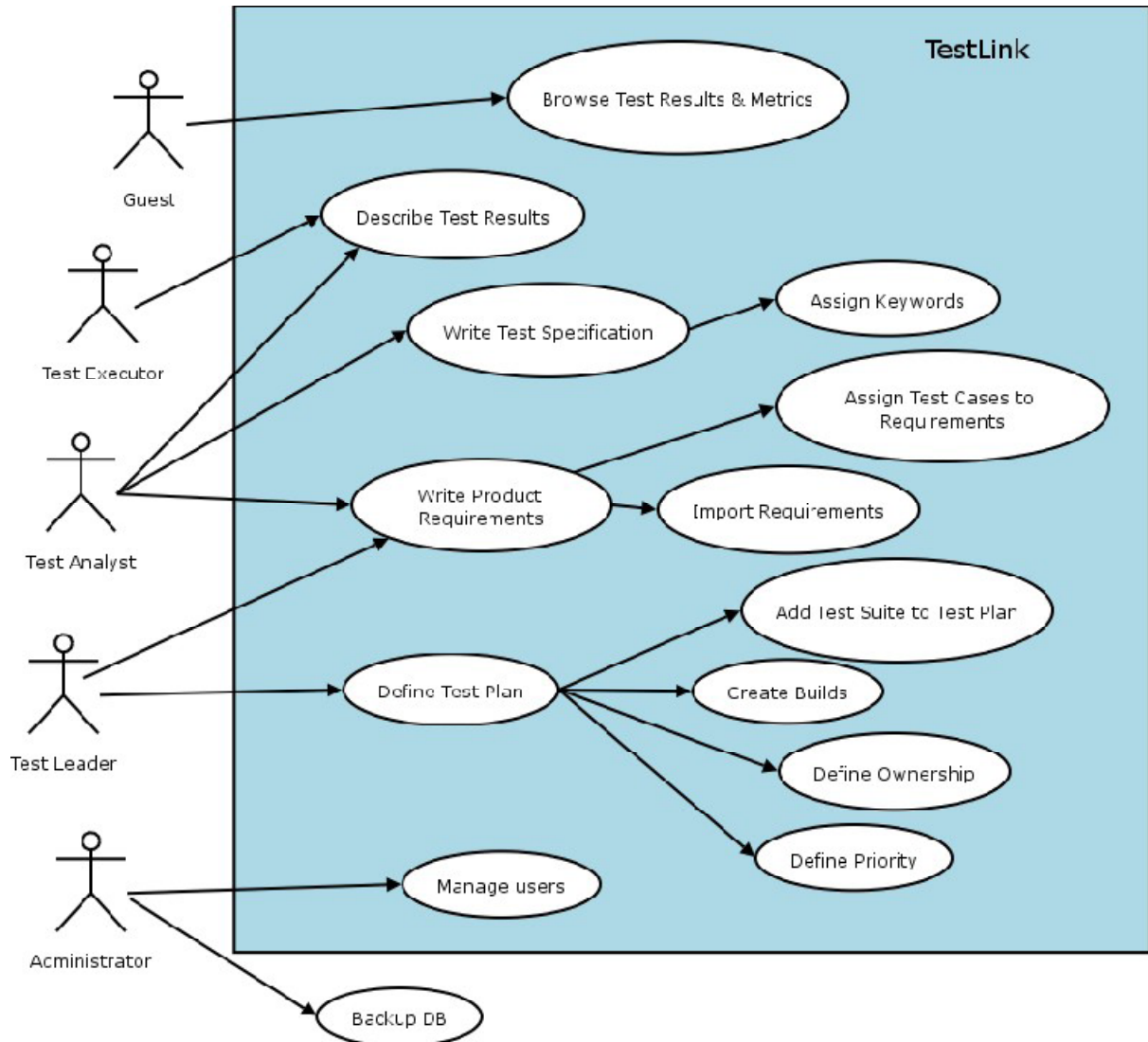


그림 2: 기능 오버뷰

## 2. 테스트 프로젝트

테스트 프로젝트는 테스트링크의 기본적인 조직 유닛이다. 테스트 프로젝트는 당신 회사의 제품을 말하는 것으로, 기능과 상관관계가 매번 바뀔 수도 있으나 대부분의 기능이 동일한 상태로 남는다. 테스트 프로젝트는 문서화, 테스트 명세, 테스트 플랜 그리고 특정한 사용자의 권한을 포함한다.

### 2.1 새 테스트 프로젝트 생성

새 테스트 프로젝트를 생성하기 위해서는 “관리자” 권한이 필요하다. 각각의 테스트 프로젝트는 고유한 이름을 가지고 있어야 한다. 이들을 시각적으로 구분하기 위해 백그라운드에 색을 넣는 기능이 테스트 프로젝트 템플릿에 할당될 수 있다. 관리자는 기능과 요구사항이 연동되도록 할 수 있다.

새 테스트 프로젝트를 생성할 때의 주의사항은 다음과 같다:

- 시스템으로부터 테스트 프로젝트를 삭제하는 것은 다수의 테스트 케이스를 미아 상태로 만들거나 시스템에서 테스트 케이스를 완전히 삭제하기 때문에 추천할 만한 일은 아니다.
- 테스트 플랜은 특정 시점 상의 테스트 프로젝트의 테스트 행위를 나타내는 것이다. 따라서, 테스트 플랜은 테스트 프로젝트의 테스트 케이스로부터 생성된다. 한 제품의 버전들을 위해 각기 다른 테스트 프로젝트 생성을 추천하지 않는다.
- 테스트링크는 XML 혹은 CSV 데이터를 테스트 프로젝트로 가져오는 것을 지원한다. 이 부분은 이후에 나올 가져오기에 관련된 장에서 더 자세하게 다루어 질 것이다.

### 2.2 테스트 프로젝트 편집 및 삭제

테스트 프로젝트를 편집하기 위해서는 “관리자” 권한이 필요하다. 사용자는 테스트 프로젝트의 이름, 백그라운드 컬러와 요구사항 기능의 사용유무를 변경시킬 수 있다.

사용자는 테스트 프로젝트가 유용하지 않게 된다면 이를 **비활성화(Inactivate)** 시킬 수 있다. 이는 곧 화면 가장 윗부분의 네비게이션 바에 위치한 리스트에서 해당 프로젝트가 보이지 않는다는 것을 의미한다(관리자는 그 프로젝트가 리스트에서 ‘\*’표시된 것으로 볼 수 있다).

사용자는 또한 테스트 프로젝트를 **삭제**할 수도 있다. 이 행동은 데이터베이스로부터 관련된 모든 데이터 역시 삭제한다. **이 행동은 되돌릴 수 없다.** 우리는 삭제 대신 비활성화 할 것을 아주 강력하게 추천한다.

### 3. 테스트 명세

테스트링크는 테스트 명세를 분석해 테스트 스위트와 테스트 케이스로 재구성한다. 이러한 레벨은 애플리케이션 전체에 걸쳐서 지속된다.

#### 3.1 테스트 스위트

사용자는 테스트 케이스를 테스트(케이스) 스위트로 구성할 수 있다. 각각의 테스트 스위트는 타이틀, 정형화된 테스트 케이스 기술, 그리고 다른 테스트 스위트로도 구성될 수 있다. 테스트링크는 테스트 스위트를 위해 트리 구조를 사용한다. 설명 부분에 대부분의 데이터에 유용한 정보를 포함하고 있는 것이 일반적이다. 예를 들어, 다음과 같은 것들이 명세될 수 있다: 초점(Scope), 구성(Configuration), 사전조건(Precondition), 관련된 문서(Related documentation), 툴(Tools), 인프라스트럭처(Infrastructure) 등.

하나 혹은 그 이상의 테스트 스위트 생성하는 것은 테스트 프로젝트를 만들었을 때 해야 하는 첫 단계의 작업 중 하나이다. 사용자(편집 권한을 가진)는 테스트 스위트와 테스트 케이스 모두에 걸쳐 생성, 삭제, 복사, 이동, 내보내기 및 가져오기를 할 수 있다. 제목과 설명 내용 역시 변경이 가능하다.

외부 문서나 그림 등이 특정한 테스트 스위트에 첨부될 수 있다. 이러한 기능은 반드시 테스트링크 환경 설정을 통해 설정되어야 한다.

#### 3.2 테스트 케이스

테스트 케이스는 특정한 프로그램 경로를 실행하거나 혹은 특정한 요구사항이 반영되어 있는지 검증하는 것과 같은 특정한 목표를 위해 개발된 입력값, 실행 사전 조건, 그리고 기대 결과값(출력값)의 일련의 조합이다.

테스트 케이스는 다음과 같은 파트로 구성된다:

- **제목(Title):** 짧은 설명이나 약자를 포함할 수 있다(e.g. TL-USER-LOGIN)
- **요약(Summary):** 개관(概觀)을 위해 아주 짧게 쓴다.
- **단계(Steps):** 테스트 시나리오(입력 행위)를 서술한다; 역시 사전조건과 명확한 정보를 여기에 포함한다.
- **기대 결과(Expected Results):** 체크포인트와 테스트되는 제품 혹은 시스템의 기대 행위를 서술한다.
- **ID:** 테스트 케이스에 숫자로 부여되는 ID는 테스트링크에 의해 자동으로 할당되며, 사용자에게 의해 변경될 수 없다. 이 ID는 시스템에 걸쳐 널리 활용되며, 테스트 케이스가 생성될 때 글로벌 카운터(Global Counter)를 사용하므로 이는 곧 테스트 케이스 생성이 테스트 프로젝트의 독립성을 의미하는 것이다.
- **첨부 파일:** 환경 설정이 허락하는 한 첨부파일이 추가될 수 있다.

#### 테스트 케이스 - 액티브 속성

만약 다양한 버전의 테스트 케이스가 존재한다면, 새로운 속성, 즉 활성화/비활성을 사용하는 것이 효과적일 수 있는데, 다음과 같은 방법으로 사용할 수 있다:

- 모든 테스트 케이스 버전은 활성화된 상태로 생성된다.
- 비활성화된 테스트 케이스 버전은 “테스트 플랜에 테스트 케이스 추가하기” 항목에서 선택이 불가능하다. 이것은 테스트 디자이너에게 매우 유용하다. 그들은 테스트 케이스의 버전을 편집하거나 변경할 수 있으며 단지 그가 작성이 완료되었다고 판단했을 때, 테스트 케이스의 상태를 활성화시

커 테스트 플랜에서 사용할 수 있도록 할 수 있다.

- 테스트 케이스가 한 번 테스트 플랜에 링크되고 결과를 가지고 되면, 테스트 케이스는 비활성화될 수 없다.

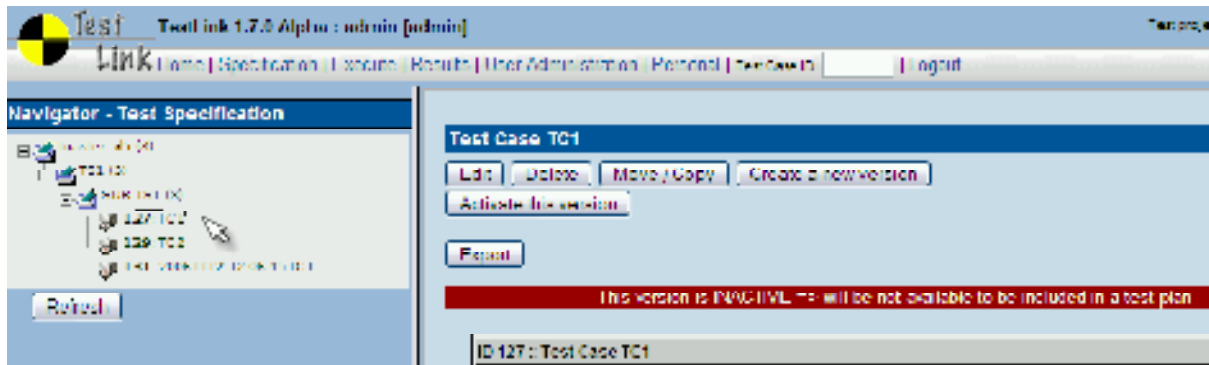


그림 3: 테스트 케이스 명세에서 볼 수 있는 항목들

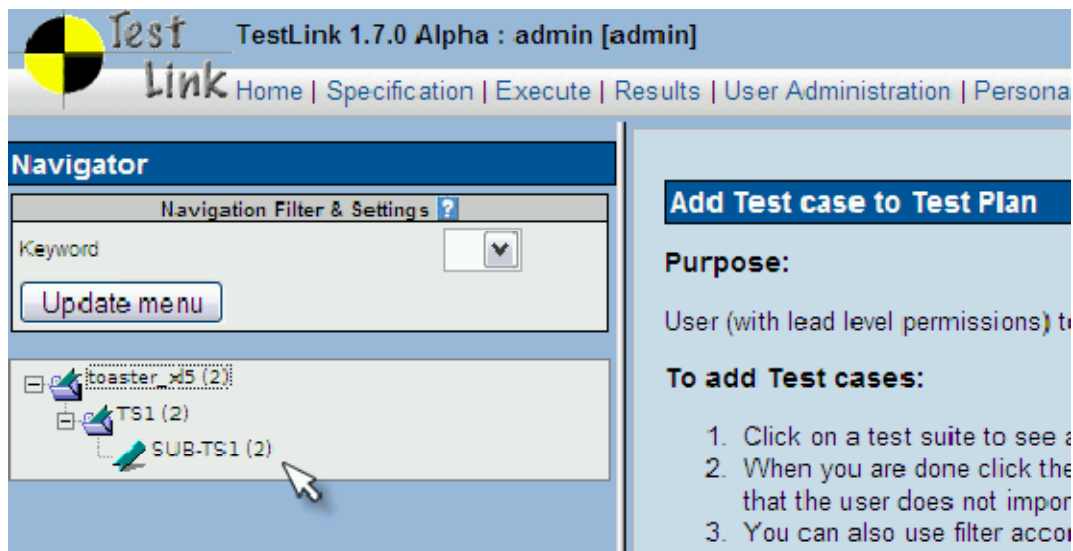


그림 4: 테스트 플랜에 테스트 케이스를 추가하는 화면

주지하듯이, 테스트 프로젝트 이름 근처의 숫자(이 예에서는 : toaster\_x15)는 2인데, 사실 테스트 프로젝트는 3개의 테스트 케이스를 가지고 있다. 테스트 케이스 TC1이 계산되지 않았기 때문인데, 이는 이 케이스가 비활성화되어 있기 때문이다.

### 테스트 케이스 삭제

테스트 케이스와 테스트 스위트는 “리더” 권한을 가진 사용자에게 의해 테스트 플랜으로부터 삭제될 수 있다. 이 행동은 테스트 플랜을 처음 수립해 어떤 결과도 발생하지 않은 동안은 매우 유용할 수 있다. 그러나, 테스트 케이스 삭제는 이와 관련된 모든 결과들을 손실할 수 있는 원인이 될 수도 있다. 그러므로, 이 기능을 사용할 때에는 세심한 주의를 기울일 필요가 있다.

### 요구사항 관계

테스트 케이스는 소프트웨어/시스템의 요구사항과 다수 대 다수로 연관될 수 있다. 이 기능은 테스트 프로젝트에서 미리 활성화되어야 사용이 가능하다. 사용자는 테스트 케이스와 요구사항을 메인 화면의 요구사항 할당하기 항목을 통해 할당할 수 있다.

### 3.3 키워드

키워드는 사용자에게 테스트 케이스를 분류하기 위한 다른 방식을 제공하기 위해 생성된다. 키워드는 테스트 명세에 포함되어 있는 몇 가지 항목을 기준으로 테스트 케이스를 묶는 방식으로 사용된다. 예를 들어, 당신은 아래와 같이 정의할 수 있다:

- 리그레션 혹은 완료(Sanity) 셋
- 리뷰된 테스트 케이스
- 한 플랫폼에서만 유효한 테스트 케이스 묶음

#### 키워드 생성

키워드는 *mgt\_modify\_key* 권한을 가진 사용자에게 의해서만 생성이 가능하다. 이 권한은 현재 리더에게만 부여되어 있다. 하나의 키워드 혹은 키워드 그룹이 생성되고 사용자가 그들을 테스트 케이스에 할당할 수 있다.

#### 키워드 할당

키워드는 키워드 할당 화면(메인 화면)이나 혹은 테스트 케이스 관리화면(개별적인)에서 테스트 케이스에 할당 가능하다.

#### 키워드에 의한 필터

사용자는 키워드를 기준으로 필터링 할 수 있는 기능을 사용할 수 있다:

- 테스트 명세 안에서 테스트 케이스 찾기
- 테스트 케이스 스위트(테스트 플랜) 안에 테스트 케이스 그룹 추가하기
- 테스트 실행 화면

## 4 요구사항 기반 테스트

시스템이 명세된 대로 만들어졌다는 것을 증명하기 위해, 사용자는 요구사항 기반 테스트를 수행한다. 각각의 요구사항들은, 하나 혹은 그 이상의 테스트 케이스로 디자인된다. 테스트 실행의 마지막에 테스트 매니저는 수행된 테스트와 요구사항이 충족되었는지에 대한 보고를 한다. 이 정보의 의거해 고객이나 다양한 이해 당사자들이 시스템이 다음 테스트 단계로 넘어갈지 혹은 바로 출시할 지를 결정하게 된다. 테스트 매니저는 고객과 이해 당사자들의 시각이 반영된 명세대로 시스템을 만들었다는 것을 확인하기 위해 위험과 요구사항 기반의 테스트를 조합해서 사용한다. 그 결과로, 이 완벽한 테스트는 다음과 같은 장점을 가져다 준다:

- 위험과 요구사항의 조합은 모호하거나 빠뜨린 요구사항을 드러내준다. 이 점은 높은 우선순위를 가진 위험 요소들에 있어 특히나 흥미로운 요소이다.
- 정보 시스템의 가장 중요한 부분의 테스트에 먼저 초점을 맞출 수 있다: 가장 높은 우선순위를 가진 위험을 커버할 수 있다.
- 고객 및 이해 당사자와 동일한 언어로 커뮤니케이션 할 수 있다. 이것은 테스트 프로젝트의 상황을 보고하는 것을 좀 더 쉽게 만들어 준다. 이후 테스트에 좀 더 많은 투자를 할지, 아니면 위험을 감수할지 좀 더 나은 근거를 가지고 결정할 수 있다.
- 위험과 그 우선순위는 시간의 압박을 받고 있는 테스트 프로젝트에 대해 좀 더 쉽게 협상할 수 있는 여지를 준다. 어떤 위험 요소들은 테스트 프로젝트 기간 안에 커버되어야 하고, 반면 나머지 일부는 연기될 수도 있다. 위험과 요구사항에 기반한 테스트는 좀 더 잘 컨트롤된 테스트 프로젝트를 그 결과로 남길 수 있다. 고객 및 이해 당사자와의 커뮤니케이션이 좀 더 향상될 수도 있다. 테스트 매니저는 가장 높은 우선순위를 가진 위험부터 테스트를 시작해야 한다. 프로세스는 더 능률적이 되고 최종결과는 더 높은 품질을 보장받을 것이다.

### 4.1 활용성

이 기능은 테스트 프로젝트 레벨에서 사용 가능하도록 할 수 있다. 즉, 관리자가 특정한 테스트 프로젝트에 대해 이를 활성화 시킬 수 있다는 것이다(메인 화면에서 테스트 프로젝트 링크를 편집). 그렇지 않으면 링크가 보이지 않는다.

이 기능을 위한 두 가지 사용자 레벨이 존재한다. 대부분의 역할은 요구사항을 볼 수 있지만, 그들을 수정할 수는 없다. 더 자세한 내용은 사용자 섹션을 참조하라.

### 4.2 요구사항 명세

요구사항은 하나 혹은 그 이상의 시스템/소프트웨어/사용자의 요구사항 명세로 묶일 수 있다.

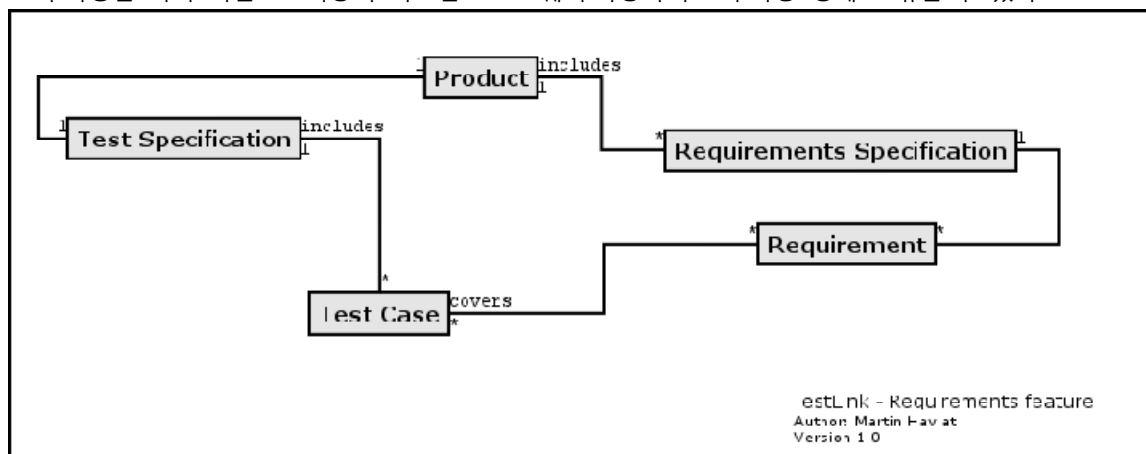


그림 5: 오브젝트와 관련된 요구사항 간의 관계

요구사항으로 문서를 만드는 방법은 다음과 같다:

1. 메인 화면에서 요구사항 명세(Requirements > Requirement Specification)를 클릭하라. 요구사항 명세의 목록이 보일 것이다.
2. 문서 생성을 위해 생성하기(Create) 버튼을 클릭하라.
3. 제목(Title), 초점(Scope)을 결정하고 최종적으로 테스트 케이스의 개수를 결정하라. 테스트 케이스의 개수는 추후 통계를 위해 활용된다. 당신이 유용한 요구사항 문서를 가졌을 때만 사용하라. 그렇다고 모든 요구사항이 테스트링크에서 모든 순간에 바로 활용 가능한 것은 아니다.
4. 데이터베이스에 데이터를 추가하기 위해 생성하기(Create) 버튼을 클릭하라. 당신은 요구사항 명세 목록의 테이블에 새로 생성된 문서의 제목을 확인할 수 있을 것이다.
5. 다음 작업을 위해 생성된 문서의 제목을 클릭하라. 요구사항 명세 화면이 나타날 것이다.

각각의 요구사항 명세는 포함된 데이터와 관련된 자체적인 통계와 보고서를 가지고 있다.

모든 명세는 “요구사항 명세” 창의 “출력하기(Print)” 버튼을 사용해 출력이 가능하다. 관리자는 환경 설정 파일<sup>3</sup>을 통해 회사, 저작권 및 포함되는 중요한 내용을 결정할 수 있다.

### 4.3 요구사항

각각의 요구사항은 제목(Title), 초점(Scope / HTML format), 그리고 상태(Status) 값을 가진다. 제목은 반드시 고유한 것으로 식별 가능해야 하며 최대 100자까지 작성이 가능하다. 초점 값은 HTML 포맷의 텍스트이다. 상태는 VALID 혹은 NOT\_TESTABLE 값을 가질 수 있다. NOT\_TESTABLE 값을 가진 요구사항의 경우 메트릭스에서 계산되지 않는다.

요구사항은 테스트링크 인터페이스를 통해, 혹은 CSV 파일을 가져오는 방식을 통해 수동으로 생성/편집 및 삭제가 가능하다.

#### 요구사항 가져오기<sup>4</sup>

테스트링크는 두 가지 타입의 CSV를 지원한다. 첫 번째 ‘간단한’ 방식은 각 열이 제목 및 초점으로만 구성되어 있다. 두 번째 ‘Export from Doors’ 방식은 헤더를 검색하고 대응하는 필드를 선택한다. 적절한 방식을 선택해 문제를 해결하라. 여기에는 세 가지 방법이 있다: 업데이트, 같은 제목으로 동일한 요구사항 생성하기, 그리고 문제를 일으키는 요구사항 추가를 중단하는 것이다.

#### 테스트 케이스에 관한 요구사항의 관계

테스트 케이스는 소프트웨어/시스템의 요구사항과 \* 대 \* 관계를 가진다. 즉, 당신은 하나 혹은 다수의 테스트 케이스를 하나의 요구사항, 그리고 하나의 테스트 케이스로 더 많이 커버될 수 있는 하나 혹은 그 이상의 요구사항에 할당할 수 있다. 사용자는 메인 화면의 요구사항 할당하기(Assign Requirements) 링크를 통해 테스트 케이스에 요구사항을 할당할 수 있다.

테스트 명세의 커버리지는 요구사항 명세 화면의 분석(Analyze) 버튼을 통해서 살펴 볼 수 있다.

#### 요구사항 기반 보고서

보고서와 메트릭스 메뉴를 검색해 보라. 거기에 요구사항 기반 보고서라는 링크가 있을 것이다. 현재 요구사

<sup>3</sup> 관리자가 직접 테스트링크가 설치된 시스템에 접속해서 config.inc.php 파일의 내용을 편집해야 한다.

<sup>4</sup> 요구사항을 외부 파일에서 가져오려면 미리 Requirement Specification이 생성되어 있어야 하며 메인 화면에서 Requirements > Requirements Specification > 해당 문서의 제목을 클릭 > List of requirements > import 버튼을 선택한다.



항 명세의 요구사항과 테스트 플랜이 이 보고서를 위해서 분석된다. 테스트 케이스(테스트 플랜에서 활용 가능한)의 가장 최근 결과가 각 요구사항에 대해 처리된다. 가장 높은 우선순위를 가진 결과가 요구사항에 반영된다. 하향식 우선순위는 다음과 같다: Failed, Blocked, Not Run and Passed.

#### 요구사항 커버리지의 예

요구사항은 세 가지 테스트 케이스에 의해 커버된다. 그 중 두 가지는 현재의 테스트 스위트에 포함되어 있다. 빌드 1에서 하나는 Pass 처리되고 하나는 테스트 되지 않았다. 이제 요구사항은 실행되지 않은 것에 대한 전체 결과를 가진다. 두 번째 테스트 케이스는 빌드 2에서 시험되고 Pass 처리되었다. 따라서 요구사항도 역시 Pass 처리되는 것이다.

## 5 테스트 플랜

테스트 플랜 프로세스의 기록은 테스트의 참여 정도, 테스트 환경, 테스트 케이스 디자인 기법 및 사용되는 테스트 측정 기법, 그리고 그들의 선택에 대한 근본적인 이유 등을 자세하게 기술한다.

테스트 플랜은 테스트 케이스 실행의 근간이다. 테스트 플랜은 이름, 설명, 선택된 테스트 케이스의 집합, 빌드, 테스트 결과, 마일스톤, 테스트 할당 및 우선순위 정의 등을 포함한다.

### 5.1 테스트 플랜 생성 및 삭제

테스트 플랜은 현재의 테스트 프로젝트에 대해 리더 권한을 가진 사용자에게 의해 “테스트 플랜 관리(Test Plan Management)” 페이지로부터 생성이 가능하다. “생성(Create)” 버튼을 누르고 데이터를 입력하라.

테스트 플랜은 특정한 시점에 테스트 명세로부터 가져온 테스트 케이스로 구성된다. 테스트 플랜은 아마도 다른 테스트 플랜으로부터 생성될 것이다. 이것은 원하는 특정 시점에 사용자가 테스트 케이스로부터 테스트 플랜을 생성할 수 있게 한다. 이것은 패치에 대한 테스트 플랜을 생성할 때 필요할 것이다. 사용자가 테스트 플랜을 열람하기 위해서는 적합한 권한을 가지고 있어야 한다. 권한은 (리더에 의해) 사용자/프로젝트 권한 정의 섹션에서 할당될 것이다. 사용자들이 당신에게 그들이 작업하고 있는 프로젝트를 볼 수 없다고 얘기할 때 이것이 아주 요긴하게 사용될 것이므로 이 사실을 기억하고 있어야 한다.

테스트 플랜은 리더 권한을 가진 사용자에게 의해 삭제가 가능하다. **테스트 플랜 삭제는 테스트 플랜과 이에 대응하는 데이터, 포함된 테스트 케이스(테스트 명세에 포함된 것이 아닌), 결과와 기타 대응하는 다른 것들 모두를 영원히 삭제하는 것이다.** 이것은 아주 특별한 경우를 대비한 것이다. 아니면, 같은 페이지에서 테스트 플랜을 비활성화 시킬 수 있는데, 이는 “메인” 화면의 선택 메뉴에서는 보이지 않는다.

### 5.2 빌드

리더 권한을 가진 사용자는 메인 페이지의 “**빌드 관리**” 링크를 사용할 수 있다.

빌드는 특정한 소프트웨어 릴리즈를 뜻한다. 회사의 각 프로젝트는 그 대부분이 서로 다른 빌드들로 구성된다. 테스트링크에서는, 실행이 빌드와 테스트 케이스 두 개로 구성된다. 만약 프로젝트를 위한 어떤 빌드도 생성되지 않았다면 실행 화면은 당신이 아무것도 실행할 수 없게 만들 것이다. 매트릭스 스크린 역시 완벽하게 공란으로 채워질 것이다.

각 빌드는 제목을 통해 구별된다. 제목은 설명(HTML format)과 두 개의 상태 항목을 가진다:

- Active / Inactive – 빌드가 테스트링크 상에서 유효한지 아닌지를 정의한다. 비활성화(Inactive) 빌드는 실행 혹은 보고서 페이지에서 보이지 않는다.
- Opened / Closed – 테스트 결과가 빌드를 위해 수정될 수 있는지를 정의한다.

빌드는 (빌드 타이틀 아래의 링크를 통해) 편집이 가능하고 존재하는 빌드의 테이블에서 삭제(적합한 “bin” 아이콘을 클릭함으로써)가 가능하다.

### 5.3 테스트 플랜 활성화하기 – 테스트 케이스 추가하기

다양한 테스트 프로젝트로부터 추출된 데이터는 하나의 테스트 플랜에 추가될 수 있다. 테스트 명세 데이터는 키워드에 의해 필터링이 가능하다(네비게이션 창에서 조절이 가능).

한 번 데이터가 테스트 플랜에 링크가 되면 체크 마크로 표시될 것이다. 만약 테스트 케이스가 이미 가져온

것이라면 다시 가져온다고 해도 무시될 것이다.

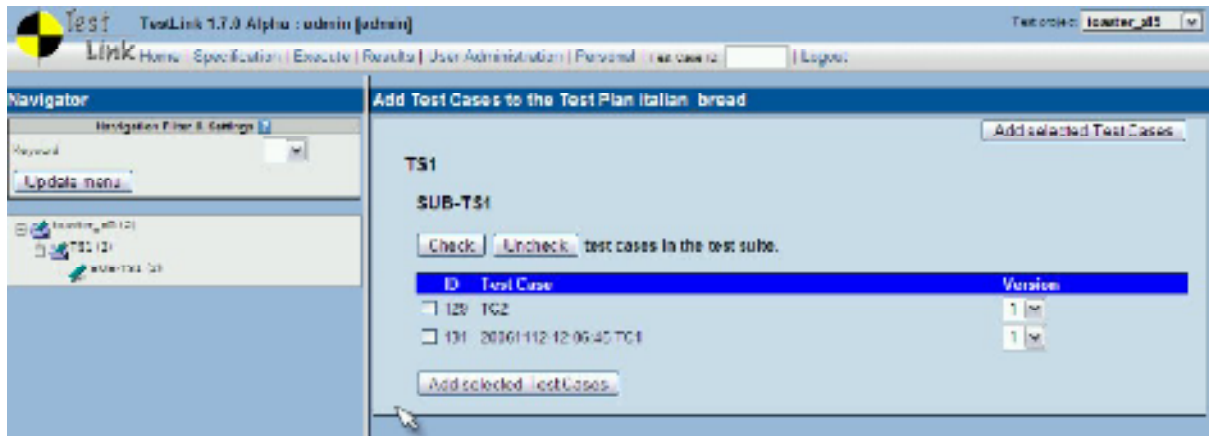


그림 6: 테스트 플랜에 테스트 케이스를 추가하기 위한 프레임

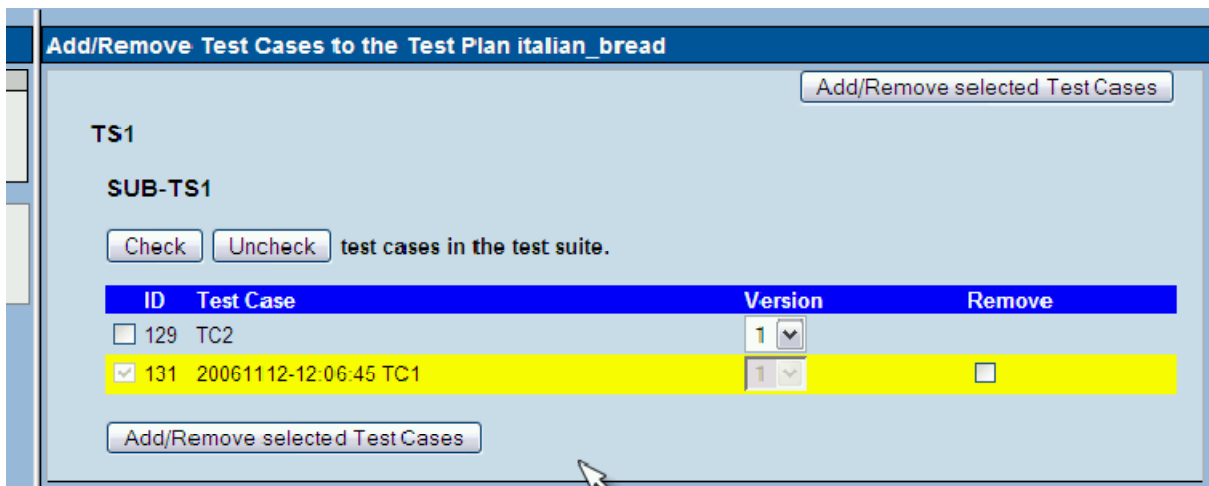
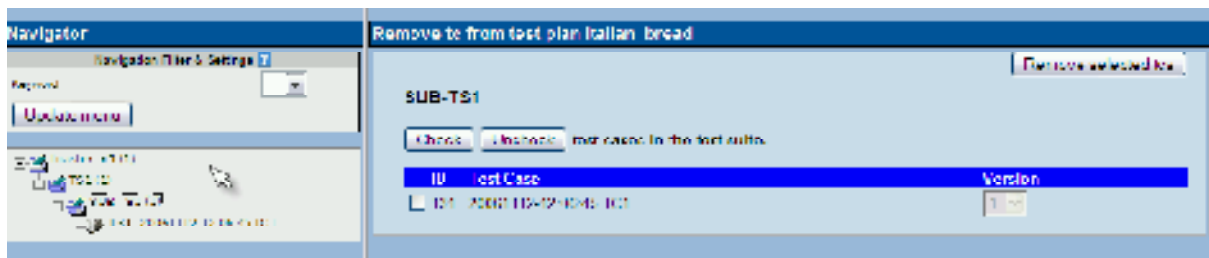


그림 7: 테스트 플랜에 포함된 테스트 케이스 내용을 수정하기 위한 프레임

### 테스트 플랜으로부터 테스트 케이스 제거하기

테스트 케이스와 테스트 스위트는 리더 권한을 가진 사용자에게 의해 “테스트 케이스 제거(Remove Test Cases)” 페이지를 통해 테스트 플랜에서부터 제거될 수 있다. 데이터를 제거하는 것은 처음 테스트 플랜을 생성하고 나서 어떤 결과도 나오지 않은 동안에는 매우 유용할 수도 있다. 그러나, 테스트 케이스를 제거하는 것은 이와 관련된 모든 결과를 소실하는 원인이 될 수도 있다. 그러므로, 이 기능을 사용할 때에는 세심한 주의가 요구된다.



왼쪽 창의 트리가 테스트 플랜에 포함된 테스트 케이스만을 보여준다.

## 5.4 테스트 실행 할당

테스트 실행 할당은 시행 및 메트릭스 화면 모두에 영향을 끼친다. 실행 화면에서 사용자는 그들에게 할당된 테스트 케이스를 확인하기 위해 실행 가능한 테스트 케이스를 분류할 수 있는 능력을 가진다. 주 메트릭스 스크린에서 사용자에 의해 남겨진 테스트 케이스 테이블이 보여진다. 만약 어떤 테스트 케이스도 테스트터에게 할당되지 않았다면 아무것도 없는 것이 기본값이다.

Assigning Test Case execution tasks for test plan TP1

TS1

Bulk user assignment

<input checked="" type="checkbox"/> ID	Test Case	Version	User
<input type="checkbox"/> 3	TC1	1	havlatm
<input type="checkbox"/> 5	TC2	1	admin

테스터는 만약 이들 메트릭스들이 활성화되어 있는 상태라면(설치 매뉴얼 참조) 메인 페이지에서 그들이 이미 실행한 메트릭스를 확인할 수도 있다.

## 5.5 우선순위

*이 기능은 1.7버전에서 잠정적으로 사용이 불가능하다. 특정한 테스트 케이스에 중요도를 할당하는 기능이 업데이트될 필요가 있다.*

테스트링크는 사용자에게 테스트 케이스에 **중요도(Importance)**를 할당할 수 있는 기능을 제공한다. 일반적인 **위험요소(Risk)**는 특정한 테스트 플랜 안에 포함되어 있는 테스트 스위트 레벨에서 결정된다.

테스트링크는 이러한 속성들을 **우선순위(Priority)**로 조합한다. 이러한 속성들(위험요소, 중요도와 우선순위)을 세 가지 레벨(낮음, 중간, 높음)을 가진다. 중간이 기본값이다.

위험요소, 중요도 그리고 우선순위를 할당하는 것은 모두 선택이 가능하다. 이것은 관리자에 의해 선택된 테스트 프로젝트에서만 가능하다.

## 5.6 마일스톤

*주의 : 버전 1.7은 마일스톤에 대한 기본 보고서를 포함하고 있지 않다.*

테스터 리더는 미리 정해진 시일 안에 얼마나 많은 테스트를 완료했는지의 비율을 계산할 수 있다. 현재의 마일스톤 종료 기준은 정의된 우선순위이다.

Milestones for Test Plan TP1

Milestone was successfully added

**New Milestone**

*Milestones must be created at today's date or greater*

Name:

Target Date:

% A Priority:

% B priority:

% C priority:

**Existing Milestones**

Name	Target Date	% A Priority	% B prior
<a href="#">One circle more</a>	13/09/2007	50	50

그림 8: 테스트 리더는 테스트 플랜을 위한 하나 혹은 그 이상의 마일스톤을 결정할 수 있다.

## 6 테스트 실행

### 6.1 일반

테스트 실행은 이와 같은 전제 조건이 갖춰진 이후에 실행이 가능하다:

1. 테스트 명세가 작성되어야 한다.
2. 테스트 플랜이 생성되어야 한다.
3. 테스트 케이스가 테스트 플랜에 추가되어야 한다.
4. 빌드가 생성되어야 한다.
5. 테스트 플랜이 테스터들에게 할당되어야 한다(그렇지 않다면 그들은 이 테스트 플랜을 검색할 수 없다).

메인 페이지에서 테스트 플랜을 선택하고 '테스트 실행(Execute tests)' 링크를 검색해보라. 트리 메뉴, 키워드 필터링, 결과, 빌드 그리고 테스터 별로 테스트 케이스 스위트 검색을 왼쪽 창에서 할 수 있다.

### 6.2 네비게이션

네비게이션 창은 '필터 & 세팅' 박스와 테스트 케이스 스위트를 가진 트리 메뉴로 구성되어 있다.

#### 필터링 테스트 케이스

이 테이블은 사용자에게 테스트 케이스를 실행하기 이전 스마트 네비게이션을 위한 테스트 케이스 필터 기능을 제공한다.

- 테스터: 사용자는 테스터 별로 테스트 케이스를 필터링 할 수 있다.
- 키워드: 사용자는 키워드를 사용해 테스트 케이스를 필터링 할 수 있다. 키워드는 테스트 케이스를 생성/편집/삭제하는데 사용될 수 있고, 다양한 케이스에 동시에 할당될 수도 있다. 키워드는 리더 권한을 가진 사용자에게 의해서만 생성되고, 편집되고, 혹은 삭제될 수 있지만 테스터에 의해 테스트 케이스에 할당이 가능하다.
- 결과: 사용자는 결과로 테스트 케이스를 필터링 할 수 있다. 결과는 특정한 빌드에서 테스트 케이스 실행 이후 일어난 일들을 말한다. 테스트 케이스는 Pass, Fail, 혹은 Block 처리되거나 그렇지 않으면 실행되지 않을 수도 있다.

#### 테스트 빌드 정의

사용자는 빌드에 의해 테스트 케이스를 필터링 할 수 있다. 빌드는 테스트 케이스를 추적하기 위한 기본적인 요소이다. 각각의 테스트 케이스는 한 빌드에서 아마도 한 번만 실행될 것이다. 빌드는 리더 권한을 가진 사용자에게 의해 새로운 빌드 페이지를 사용해 생성이 가능하다.

#### 트리 메뉴

네비게이션 창의 트리 메뉴는 결과 별로 색칠된 테스트 케이스 스위트를 포함하고 있다.

메뉴 색깔: 기본적으로 트리는 드롭다운 박스로부터 선택되어 결정된 빌드의 결과에 의해 분류된다.

#### 빌드에 의해 색칠된 TC 예제

사용자가 드롭다운 박스로부터 빌드 2를 고르고 "가장 최근(Most Current)"의 체크박스에는 체크하지 않는다. 빌드 2의 모든 테스트 케이스가 보여질 것이다. 따라서, 만약 테스트 케이스 1이 빌드 2에서 Pass 처리되었다면, 아마도 이것은 녹색으로 보일 것이다.

두 번째로 가능한 경우는 가장 최근의 테스트 결과에 따라 메뉴의 색이 결정되는 것이다.

### 가장 최근 결과에 의해 색칠된 TC 예제

사용자가 드롭다운 박스로부터 빌드 2를 고르고 이번에는 “가장 최근(Most Current)”의 체크박스에 체크한다. 빌드 2의 모든 테스트 케이스가 가장 최근의 결과와 함께 보여질 것이다. 따라서, 만약 테스트 케이스 1이 빌드 3에서 Pass 처리되었다면, 비록 이미 빌드 2를 선택했다고 하더라도, 아마도 이것은 녹색으로 보일 것이다.

## 6.3 실행

### 테스트 상태(Test Status)

실행은 특정한 빌드에서 테스트 케이스에 그 결과(pass, fail, blocked)를 할당하는 프로세스를 말한다. ‘Block’ 처리된 테스트 케이스는 어떤 이유(e. g. 시험되어야 할 기능을 구동하지 못하게 하는 구성상의 문제)에서 테스트 되지 못한 것이다.

### 테스트 결과 입력

테스트 결과 화면은 네비게이션 창에서 적합한 테스트 스위트나 혹은 테스트 케이스를 클릭함으로써 보여진다. 타이틀이 현재의 빌드와 소유자를 보여준다. 색으로 칠해진 바는 테스트 케이스의 상태를 보여준다. 노란색의 상자 아이콘은 테스트 케이스의 테스트 시나리오를 포함하고 있다.

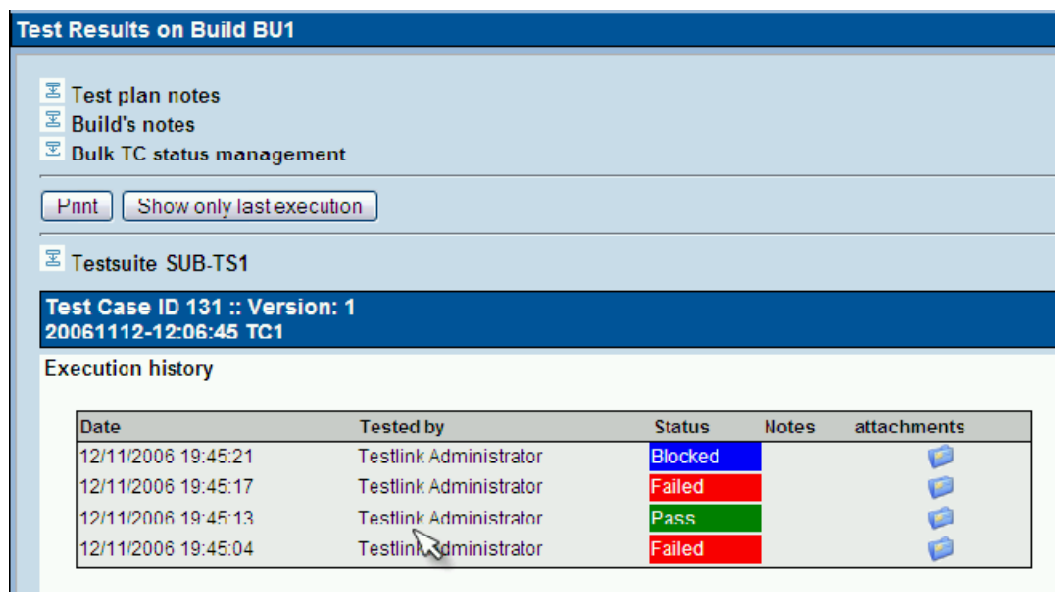


그림 9: 하나의 테스트 케이스에 대한 다양한 결과를 보여주는 프레임

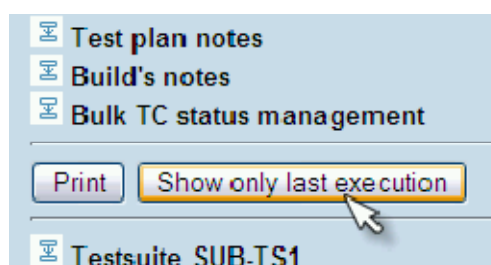


그림 10: 사용자는 가장 마지막의 결과만을 출력하는 것을 선택할 수 있다.

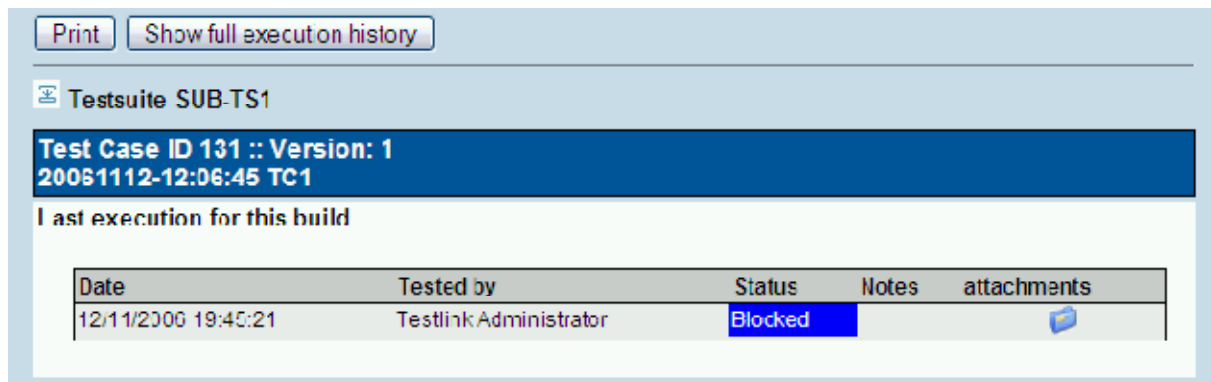


그림 11: 가장 마지막의 결과만 출력이 될 수도 있다.

테스트 명세에서 업데이트되거나 삭제된 테스트 케이스를 보여주는 기능은 1.5 버전 이후에는 지원되지 않는다.

**테스트 케이스 업데이트:** TL 1.0.4 버전은 플래그(Flag)에 의한 표시 방법을 채택하고 있는데, 이는 1.6 버전 부터 삭제되었다. 만약 사용자가 적합한 권한을 가지고 있다면, 메인 페이지의 링크를 통해 “수정된 테스트 케이스 업데이트” 항목으로 갈 수도 있다. 만약 변화(더 새로운 버전이나 삭제됨)가 발생한다고 해도 사용자는 테스트 케이스를 직접 업데이트할 필요가 없다.



## 7 커스텀 필드

커스텀 필드에 대한 정의는 전 시스템 보편적으로 사용된다. 예를 들어, 동일한 필드 ID로 두 개의 커스텀 필드를 정의할 수 없다. 당신이 커스텀 필드를 만들고 난 뒤, 당신은 당신이 사용하기 원하는 테스트 프로젝트에 이를 할당해야 할 필요가 있다.

*커스텀 필드는 맨티스(<http://www.mantisbt.org/>)와 닷프로젝트(<http://dotproject.net/>) 모델의 기능을 혼합해서 구현이 가능하다.*

### 속성 보여주기 / 활성화하기

디자인 보여주기: 커스텀 필드는 테스트 케이스 명세 도중에도 보여질 수 있다.

디자인 활성화하기: 사용자는 테스트 케이스 명세 도중에 커스텀 필드 값을 할당/변경 할 수 있다.

실행 보여주기: 커스텀 필드는 테스트 케이스 실행 도중에도 보여질 수 있다.

실행 활성화하기: 사용자는 테스트 케이스 실행 동안 커스텀 필드 값을 할당/변경 할 수 있다.

할당된 값은 저장이 가능하다.

### 예제 1.

Custom Field	: Additional Notes
Type	: String

테스트 스위트에 적용하기 위해, 테스트 케이스를 명세하는 동안에만 편집이 가능하나 테스트 실행 기간 동안 보여질 수 있다.

show on design	= YES
enable on design	= YES
show on execution	= YES
enable on execution	= NO

### 예제 2.

Custom Field	: Operating System
Type	: list

테스트 케이스들에 적용하기 위해, 테스트 케이스 실행 동안만 편집이 가능하며, 테스트 케이스 디자인 동안에는 사용하지 못하게 한다.

show on design	= NO
enable on design	= NO
show on execution	= YES
enable on execution	= NO

## 8 테스트 보고서와 매트릭스

테스트 보고서와 매트릭스는 메인 페이지의 “결과(Results)” 혹은 “테스트 보고서와 매트릭스(Test reports and Metrics)” 링크를 통해 액세스된다. 보고서와 매트릭스는 선택된 테스트 플랜(콤보박스 메뉴에 있는)에 기반한다. 이 페이지는 사용자에게 다음과 같은 사항을 보여준다:

**오른쪽 창**에는 어떻게 컨트롤을 사용하고 어떻게 각 보고서를 발행하는 지가 소개되어 있다.

**왼쪽 창**은 각각의 보고서를 탐색하고 어떻게 보고서를 작성하고 출력하는지에 영향을 주는 컨트롤 조정에 사용된다. “프린트” 버튼은 오른쪽 창의 내용을 프린트한다(네비게이션 창의 내용은 프린트되지 않는다).

모든 테스트 보고서(차트 제외)가 3가지 형식 중 하나로 작성된다:

1. 일반(Normal) – 보고서가 웹 페이지(HTML)로 표시된다.
2. MS Excel – 마이크로소프트 엑셀로 내보내진다.
3. HTML Email – 보고서가 사용자의 이메일 주소로 발송된다.

현재 보고서 부분에는 선택할 수 있는 9가지 항목이 있으며, 각 항목의 목적과 기능은 아래에 설명되어 있다. 현재로서는, 다양한 여러 테스트 플랜에서 하나의 결과를 도출해 낼 수 있는 보고서는 없다.

### 8.1 일반 테스트 플랜 매트릭스

이 페이지는 당신에게 테스트 스위트, 소유자, 그리고 키워드에 따라 단지 테스트 플랜의 가장 최근 상태를 보여준다. 가장 “최근 상황”은 실행되었던 가장 최근의 빌드 테스트 케이스에 의해 결정된다. 예를 들어, 만약 테스트 케이스가 여러 빌드에 대해 실행되었다면, 가장 마지막 결과만이 고려되는 것이다. **“최종 테스트 결과(Last Test Result)”**는 많은 보고서에서 사용된 개념이며, 아래의 사항들에 의해 결정된다:

- 1) 어떤 빌드가 테스트 플랜에 추가되었는지의 순서가 가장 최신의 것인지를 결정한다. 가장 최근 빌드의 결과에 오래된 빌드의 결과들보다 우선순위가 있는 것이다. 예를 들어, 만약 당신이 빌드 1에서 테스트 결과를 “Fail”로 처리하고, 빌드 2에서 이를 “Pass” 처리했다면, 이 최종 결과는 “Pass” 처리될 것이다.
- 2) 테스트 케이스가 동일한 빌드에서 여러 번 실행된다면, 가장 최근의 실행 결과가 우선순위를 가진다. 예를 들어, 빌드 3이 당신의 팀에게 배포되고 1번 테스터가 오후 2시에 이를 “Pass” 처리하고, 2번 테스터가 이를 오후 3시에 “Fail” 처리했다면 – 이는 “Fail”로 처리되어 표시될 것이다.
- 3) 빌드에서 “Not Run”으로 표시된 항목은 고려되지 않는다. 예를 들어, 당신이 빌드 1에서 “Pass”로 처리하고, 이를 빌드 2에서 실행하지 않았다면, 이것의 최종결과는 아마도 “Pass”로 간주될 것이다.

다음과 같은 테이블들이 표시된다:

#### 최고위 테스트 스위트(Top Level Test Suite)에 의한 결과

각각의 최고위 스위트에 의한 결과를 보여준다. 총 테스트 케이스, Passed, Failed, Blocked, Not Run, 그리고 완료된 백분율 등을 보여준다. “완료된” 테스트 케이스는 Pass, Fail 혹은 Block 중 하나로 처리된 것이다. 최고위 스위트에 대한 결과는 모든 파생 스위트(Children Suites)를 포함한다.

#### 키워드에 의한 결과

현재의 테스트 플랜에 포함되어 있는 테스트 케이스에 할당된 모든 키워드를 보여주고 그와 관련

되어 있는 결과를 보여준다.

예:

Keyword	Total	Passed	Failed	Blocked	Not run	Completed [%]
P3	1128	346	47	55	680	39.72
P2	585	372	25	31	157	73.16
P1	328	257	6	51	14	95.73

#### 소유자에 의한 결과

현재의 테스트 플랜에서 테스트 케이스가 할당된 각각의 소유자를 보여준다. 할당되지 않은 테스트 케이스는 “할당되지 않은(Unassigned)” 상태로 기록된다.

예:

Tester	Total	Passed	Failed	Blocked	Not run	Completed [%]
Dominika	579	217	34	47	281	51.47
Mohammad	246	82	9	2	153	37.8
unassigned	35	19	0	1	15	57.14
Ken	289	110	1	21	157	45.67
Mallik	430	269	5	18	138	67.91
Ali	227	123	28	13	63	72.25
Mike	24	22	0	0	2	91.67
Alex	272	155	1	36	80	70.59

## 8.2 전체 빌드 현황

모든 빌드에 대한 실행 결과를 보여준다. 각각의 빌드, 총 테스트 케이스, 총 Pass, Pass율(% pass), 총 Fail, Fail율(% fail), 블록, 블록율(% Block), Not run, % Not run 등이 표시된다. 만약 같은 빌드에서 두 번 동일한 테스트 케이스가 실행되었다면, 가장 최근의 실행 결과가 영향을 끼칠 것이다.

## 8.3 쿼리 매트릭스

이 보고서는 쿼리 폼 페이지, 쿼리된 데이터를 포함하는 쿼리 결과 페이지로 구성된다.

#### **쿼리 폼 페이지:**

사용자는 4가지 컨트롤을 가진 쿼리 페이지를 제공받는다. 각각의 컨트롤은 쿼리가 수행되어야 할 수많은 경우 중, 대상이 되는 테스트 케이스와 빌드를 최대화할 수 있는 값으로 디폴트가 설정되어 있다. 컨트롤의 변경은 사용자로 하여금 그 결과를 필터링하고 특정한 소유자, 키워드, 스위트, 그리고 빌드의 조합에 따라 특별한 보고서를 작성할 수 있게끔 해준다.

키워드 - 0 -> 1 키워드 선택이 가능하다. 기본값으로는 키워드 선택 불가(No Keyword)가 선택되어 있다. 만약 키워드가 선택되지 않는다면, 키워드 할당과 상관없이 모든 테스트 케이스가 선택된다. 키워드는 테스트 명세나 혹은 키워드 관리 페이지에서 할당이 가능하다. 키워드는 모든 테스트 플랜의 테스트 케이스, 그리고 모든 버전의 테스트 케이스에 걸쳐 할당이 가능하다. 만약 당신이 특정한 키워드에 대한 결과에 관심이 있다면 이 컨트롤 값을 변경하면 된다.

소유자 - 0 -> 1 소유자 선택이 가능하다. 기본값으로는 소유자 선택 불가(No Owner)가 선택되어 있다. 만약

소유자가 선택되지 않는다면, 소유자 할당과 상관없이 모든 테스트 케이스가 선택된다. 현재로서는 “할당되지 않은” 테스트 케이스를 찾는 기능은 지원하지 않는다. 소유권은 “테스트 케이스 실행 할당(Assign Test Case execution)” 페이지를 통해서 일어나고, 각각의 테스트 플랜을 기초로 해서 완료된다. 만약 당신이 특정한 테스터가 완료한 작업에 관심이 있다면 이 컨트롤 값을 변경하면 된다.

최고위 스위트 - 0 -> n 최고위 스위트가 선택 가능하다. 기본값은 모든 스위트가 선택된 것이다. 단지 선택된 스위트에서만 결과 매트릭스를 위한 쿼리가 수행될 것이다. 만약 당신이 단지 특정한 스위트에 대한 결과에만 관심이 있다면 이 컨트롤 값을 변경하면 된다.

빌드 - 1 -> n 빌드를 선택할 수 있다. 기본값은 모든 빌드가 선택되어 있는 것이다. 단지 당신이 선택한 빌드에서 실행된 것들만이 매트릭스를 생성할 때 고려될 것이다. 예를 들어 - 만약 당신이 최근의 세 빌드에서 수행된 테스트 케이스가 몇 개나 되는지를 알고 싶다면 - 당신은 이 컨트롤 값을 변경하면 된다.

키워드, 소유자, 그리고 최고위 스위트 선택은 스위트나 테스트 플랜 매트릭스를 작성하는 데 사용될 테스트 케이스 숫자를 알려줄 것이다. 예를 들어, 만약 당신이 소유자 = “Greg”, 키워드 = “Priority 1”, 그리고 모든 활용 가능한 테스트 스위트를 선택한다면 - Greg에게 할당된 Priority 1 테스트 케이스만이 선택될 것이다. 당신이 보고서에서 보게 될 “테스트 케이스의 개수” 총합은 이 세 가지 조건에 의해 영향을 받는 것이다.

빌드 선택은 테스트 케이스가 “Pass”, “Fail”, “Blocked” 그리고 “Not Run”으로 간주될지 아닐지에 영향을 끼칠 것이다. 이들이 어떻게 표시되는지는 “**최종 테스트 결과**” 항목을 참조하라.

쿼리 수행을 진행하고 결과 페이지를 출력하기 위해 “승인” 버튼을 눌러보라.

쿼리 보고 페이지:

보고 페이지는 다음과 같은 항목을 보여줄 것이다:

1. 보고서 생성에 사용되는 쿼리 파라미터
2. 전체 테스트 플랜의 모든 요소들(총계 / Pass / Fail / Blocked / Not Run)
3. 각각의 스위트에서의 모든 요소들(총계 / Pass / Fail / Blocked / Not Run)에 대한 내역과 해당 스위트에 대해서 수행된 **모든** 실행 결과. 만약 테스트 케이스가 다양한 빌드에 대해 한 번 이상 수행되었다면 - 모든 수행이 표시되고 이것이 선택된 빌드에 대해 저장될 것이다. 그러나, 해당 스위트에 대한 요약(Summary)은 단지 선택된 빌드의 “최종 테스트 결과”만을 포함할 것이다.

## 8.4 Blocked, Failed, and Not Run Test Case Reports

이 보고서는 현재 블록 처리되고, Fail 처리되고, 혹은 수행되지 않은 모든 테스트 케이스를 보여준다. 만약 테스트 케이스가 블록 처리되거나, Fail, 혹은 수행되지 않은 것으로 간주되어야 하는지 결정하기 위해, “**최종 테스트 결과**” 로직(이는 일반 테스트 플랜 매트릭스에서 묘사된)이 다시 사용된다. 만약 사용자가 통합된 버그 추적 시스템을 사용하고 있다면, 블록 처리 되고 실패한 테스트 케이스 보고서는 이와 관련된 버그 역시 표시해 줄 것이다.

## 8.5 테스트 보고서

모든 빌드의 모든 테스트 케이스의 상태를 보여준다. 만약 테스트 케이스가 같은 빌드에서 여러 번 실행되었다면 - 가장 최근의 실행 결과만이 사용된다. 만약 대규모의 데이터 셋이 사용되었다면, 쉬운 검색을 위해 이 보고서를 엑셀 형식으로 변환해서 사용할 것을 추천한다.

## 8.6 차트

이 보고서 페이지는 당신이 웹 브라우저에 플래시 플러그인(Flash plugin)을 설치할 것을 요구한다. “최종 테스트 결과” 로직은 당신이 보게 될 4종류의 차트 모두에 사용된다. 그래프가 사용자에게 현재의 테스트 플랜으로부터 매트릭스를 시각화할 수 있도록 도와줄 것이다.

네 종류의 차트는 다음과 같은 사항들을 제공한다:

1. 전체 pass/fail/blocked/그리고 not run 테스트 케이스에 대한 원형 차트
2. 키워드에 의한 결과를 보여주는 막대 차트
3. 소유자에 의한 결과를 보여주는 막대 차트
4. 최고위 스위트에 의한 결과를 보여주는 막대 차트

막대 차트의 막대들은 사용자가 pass, fail, blocked 그리고 Not run 처리된 테스트 케이스들의 숫자를 대략 근접해서 구별할 수 있도록 색칠이 되어 있다.

그래픽 형식으로 결과를 보여주기 위해 <http://www.maani.us> 에서 제공되는 플래시 기술을 사용한다.

## 8.7 각 테스트 케이스에서 발생한 총 버그

이 보고서는 전체 테스트 프로젝트에 대해 발생한 버그를 가진 모든 테스트 케이스를 보여준다. 이 보고서는 버그 추적 시스템이 연결되어 있을 때에만 사용이 가능하다.

## 8.8 요구사항 기반한 보고서

이 보고서는 현재의 프로젝트에 대해 요구사항 기능이 사용 가능할 때에만 사용 가능하다. 보고서는 콤보박스로부터 선택된 하나의 요구사항 명세 문서에 대해 생성된다.

두 가지의 섹션이 있다: 매트릭스와 결과 오버뷰이다.

아래 매트릭스들이 활용 가능하다:

- 요구사항의 총 개수
- 테스트링크에 포함된 요구사항
- 테스트 케이스에 의해 커버된 요구사항
- 테스트 케이스에 의해 커버되지 못한 요구사항
- 커버되지 못하거나 테스트 되지 못한 테스트 케이스
- 테스트되지 못한 요구사항

요구사항은 네 가지 섹션으로 구분된다. 각각의 요구사항은 관련되어 있는 테스트 케이스(테스트 케이스 결과에 따라 색이 칠해진)들과 함께 정리되어 보여진다:

- Pass 처리된 요구사항
- Fail 처리된 요구사항
- 블록 처리된 요구사항
- 실행되지 못한 요구사항

## 8.9 새 보고서 추가하기

현재의 보고서를 복사하고 당신의 필요에 따라 수정해 보라. 우리가 렌더링을 위한 템플릿(<testlink\_root>/gui/templates/<report\_name>.tpl)과 로직(<testlink\_root>/lib/results/<report\_name>.php)을 사용하고 있다는 것을 잊지 마라. 새로운 형식의 보고서 작성을 위해 새로운 기능을 만드는 것보다, 데이터를 수집하는 현재 기능을 재사용할 것을 추천한다.

새로운 보고서에 링크를 추가하기 위해 <testlink\_root>/lib/results/resultsNavigator.php를 편집하라. 해당 파일에는 쉽게 변경할 수 있는 배열이 있다. 당신은 새로운 URL과 보고서의 “제목 키워드(Name keyword)”<sup>5</sup>를 추가해야 한다.

당신은 보고서의 CSS 스타일도 편집할 수 있다. 이미 존재하는 것을 편집하는 것보다 새로운 클래스를 생성할 것을 추천한다(다른 페이지에서의 원치 않은 변화를 피하기 위해).

*만약 당신이 당신의 새 보고서를 우리의 추적기(tracker)를 통해서 배포한다면, 당신이 다음 버전에서도 역시 이것을 찾아볼 수 있을 것이다... 그렇지 않다면, 당신은 다음 주요 릴리즈에서 동작하지 않는 위험을 감수해야 할 것이다.*

---

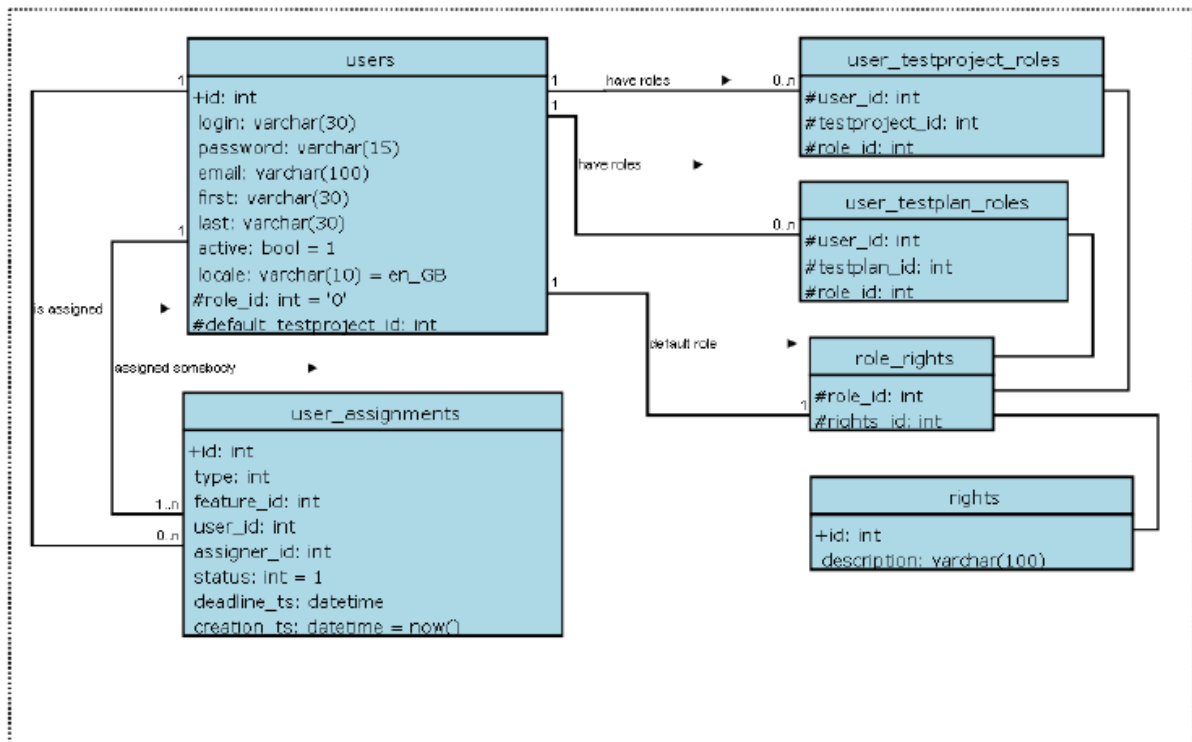
<sup>5</sup> 여기에 사용될 문자값은 반드시 locale/en\_gb/string.txt 에서 정의되어 있어야 한다.

## 9 사용자 관리

### 9.1 계정 셋팅

시스템 상의 모든 사용자들은 계정 셋팅 화면(메뉴 바의 퍼스널 링크를 통해)를 통해 그들의 정보를 편집할 수 있다.

관리자는 시스템 안에서 사용자를 생성하고, 편집하고 삭제할 수 있다. 하지만, 관리자가 사용자의 비밀번호를 보거나 편집할 수는 없다. 만약 사용자가 비밀번호를 잊어버렸다면, 로그인 화면의 링크를 통해 사용자에게 사용자가 입력한 사용자 명과 이메일 주소에 기반해 그들의 비밀번호를 메일로 보내게 된다.



### 9.2 역할 승인

테스트링크는 내장되어 있는 서로 다른 6가지 기본 승인 레벨을 가지고 있다. 이 권한을 바꾸는 것은 관리자가 접속할 수 있는 사용자 관리 링크에 의해 조정된다. 이러한 승인 레벨은 다음과 같다:

- **게스트(Guest):** 게스트는 단순히 테스트 케이스와 프로젝트 매트릭스를 열람할 권한만 가지고 있다.
- **테스트 실행자(Test Executer):** 회사 외의 조직에 속하는 테스터로 그들에게 할당된 테스트만 실행할 수 있는 권한을 가진다. (초기 1.0.4 버전에는 'otester'라고 표기되어 있음)
- **테스트 디자이너(Test Designer):** 테스트 명세와 요구사항에 관한 모든 권한을 가지고 정상적으로 작업할 수 있는 사용자
- **테스트 분석가(Test Analyst):** 테스트 케이스를 실행할 수 있을 뿐만 아니라 검색, 생성, 편집 및 삭제할 수 있는 테스터. 테스터 분석가는 테스트 플랜, 테스트 프로젝트 관리, 마일스톤 생성 혹은 권한 할당 등의 권한은 없다. (초기에는 테스터, 시니어 테스터라고 불렀음)
- **테스트 리더(Test Leader):** 리더는 테스터 분석가와 동일한 권한을 가지나 테스트 플랜 관리, 권한 할당, 마일스톤 생성, 키워드 관리 등의 권한도 같이 가진다.
- **관리자(Administrator):** 관리자는 가능한 모든 권한(테스트 프로젝트와 사용자를 관리하는 권한이

리더 권한에 더해짐)을 가진다.

주의: 테스트 플랜과 관련된 기능은 역시 테스트 플랜 할당이 활성화된 상태에서만 사용이 가능하다. 테스트 플랜 할당을 참조하라.

### 사용자 역할

미리 정의된 사용자의 역할을 말한다. 관리자가 테스트링크에서 데이터를 수정할 수 있는 적합한 권한을 줄 수 있다. 각각의 사용자들은 단지 하나의 역할에만 할당되어 있다.

만약 당신이 테이블을 볼 수 있다면 당신은 각각의 승인 레벨(게스트, 테스터, 시니어 테스터, 리더, 관리자)에 따라 해당 항목을 볼 수 있을 것이다. 두 번째 칸은 아래에 정의되어 있는 서로 다른 모든 권한 레벨을 보여주고 있다. 이러한 레벨은 사용자 표준으로부터 정의되지만 새로운 역할 정의로 편집도 가능하다(경험있는 관리자에 의해). 사용자 테이블은 오른쪽 테이블에 표시되어 있는 적합한 승인 레벨을 표시하기 위한 외부 키(foreign key)를 포함하고 있다.

Role	List of Rights	Permissions
Guest	mgt_view_tc, mgt_view_key, tp_metrics	Browse data only.
Test Executor (Tester)	tp_execute, tp_metrics	Execute test only.
Test Analyst (Senior Tester)	tp_execute, tp_metrics, tp_create_build, mgt_view_tc, mgt_modify_tc, mgt_view_key, mgt_view_req	Edit Test Specification, execute tests, create Build.
Test Designer	tp_metrics, mgt_view_tc, mgt_modify_tc, mgt_view_key, mgt_modify_req, mgt_view_req	Edit Test Specification and Requirements.
Test Leader	tp_execute, tp_create_build, tp_metrics, tp_planning, tp_assign_rights, mgt_view_tc, mgt_modify_tc, mgt_view_key, mgt_modify_key, mgt_view_req, mgt_modify_req, mgt_modify_product, mgt_users	All Test Plan permissions, edit Test Specification and execute tests.
Administrator	tp_execute, tp_create_build, tp_metrics, tp_planning, tp_assign_rights, mgt_view_tc, mgt_modify_tc, mgt_view_key, mgt_modify_key, mgt_view_req, mgt_modify_req	Everything possible. Only this role can maintain Test Projects and users.

Table 1: 역할 내용

### 케이스 스터디 - 기본값에 의해 제한된 액세스

#### 상황

조직 내에서 특별히 승인되지 않은 인원의 프로젝트 접근을 엄격하게 제한하길 원한다. 우리는 현재 90개가 넘는 서로 다른 프로젝트에 150여명의 사용자를 가지고 있다. 사용자들 중 다수가 QA팀이 아니고, 비즈니스 분석가이며, 그리고 몇몇 경우에는 최종 사용자가 UAT를 수행한다. 제시한 조건으로부터, 사용자를 어떻게 설정하느냐에 기반해 모든 권한이 상속되어야 한다.

그러나 우리는 하나의 프로젝트에서 일하는 비즈니스 분석가들이 모든 90개의 프로젝트에 접속하는 것을 원하지 않는다.



## 해결책

이러한 사용자들의 전역 역할(Global role)을 <No rights>로 설정하고 테스트 프로젝트나 테스트 플랜에 기반해 적합한 역할만을 승인하라. Const.inc.php 파일에서 기본 역할 ID를 <No rights>로 설정할 수 있다 (parameter \$g\_default\_roleid).

또한 당신은 역할의 이름을 “No rights”에서 좀 더 공손한 표현으로 바꿀 수도 있다.

## 9.3 역할 정의

역할별 능력 정의를 위해 사용되는 키워드는 다음과 같다.

Right	Description
mgt_view_tc	View Test Specification (Test Suites and Test Cases)
mgt_modify_tc	Edit Test Specification (create, modify, delete, order, move, and copy - both Test Suites and Test Cases)
mgt_view_key	View keywords
mgt_modify_key	Modify keywords
mgt_modify_product	Create, edit and delete Test Projects
mgt_view_req	View requirements
mgt_modify_req	Create, edit, associate and delete requirements

테이블 2: 테스트 프로젝트와 관련된 권한들

Right	Description
tp_execute	Ability to execute Test Cases (insert Test Results)
tp_create_build	Ability to create Builds
tp_metrics	View metrics
tp_planning	Create, edit, delete Test Plans, assign risk/ownership, milestones, edit Test Case Suite
tp_assign_rights	Assign the rights to view projects
mgt_modify_req	Create, edit, associate and delete requirements

테이블 3: 테스트 플랜과 관련된 권한들

## 9.4 테스트 플랜 할당

사용자는 사용자에게 할당된 테스트 플랜만을 볼 수 있다. 테스트 플랜에 대한 승인을 얻기 위해서는 리더 혹은 관리자 권한을 가진 사용자가 “테스트 플랜 관리(Test Plan Management)” 아래의 “사용자/프로젝트 권한 정의”를 통해 그들에게 권한을 부여해야 한다.

시스템의 모든 사용자는 기본적으로 새롭게 생성되는 테스트 플랜에 대해서는 어떠한 권한도 없다(테스트 플랜 생성자가 스스로에게 생성 권한을 준 경우를 제외하고). 한 개의 테스트 플랜에도 접근할 수 없다는 것은 사용자가 주 화면의 테스트 플랜 드롭다운 박스에서 어떤 테스트 플랜도 볼 수 없다는 것을 의미한다.

여기 테스트 플랜 권한에 대한 테이블이 있다(i.e. 어느 사용자가 어떤 테스트 플랜을 볼 수 있는지). 이 테이블은 사용자 ID와 프로젝트 ID를 혼합해 구성된다. 주 페이지는 로그인한 사용자가 적합한 권한을 가지고

있는지 검사하는 코드를 포함하고 있다(그리고 아마 허락된 프로젝트를 보여줄 것이다). 이것은 해킹될 가능성이 있으므로 추천되지 않는다.

## 10 데이터 가져오기와 내보내기

테스트링크는 다양한 방법으로 데이터의 공유를 지원한다.

Item	File format	What you get
Keyword	CSV XML	All Test Project's keywords
Test Project	XML	All Test Suites and Test Cases. You can choose whether to also export assigned keywords.
Test Suite	XML	Test Suite details, All Test Cases and child test suites and Test Cases. You can choose whether to also export assigned keywords.
Test Case	XML	Two types of exports can be done: - Just one Test Case - All Test Cases in Test Suite. You can choose whether to also export assigned keywords.
Requirement	CSV CSV DOORS (*) XML	(*) Only import is supported for this format.

테이블 4: 내보내기/가져오기 될 수 있는 항목들

제한사항: 첨부 파일과 커스텀 필드는 내보내 질 수 없다.

### 10.1 키워드 가져오기/내보내기

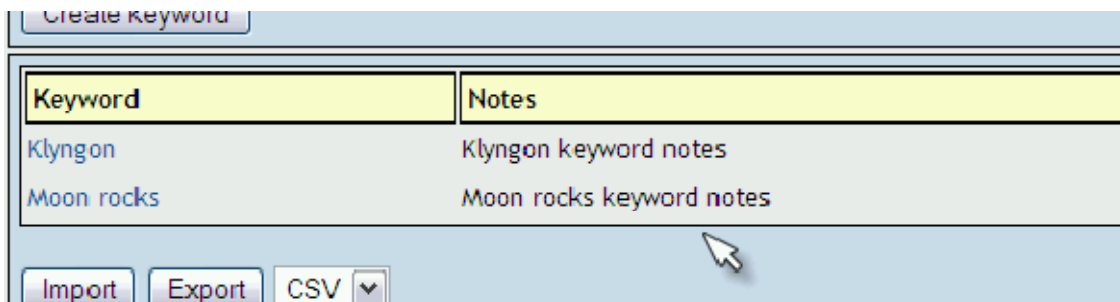


그림 12: 가져오기와 내보내기 버튼을 포함하고 있는 키워드 프레임

#### CSV “Keyword;Notes” 예제

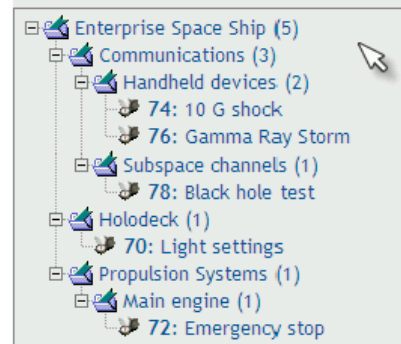
```
Klyngon;Klyngon keyword notes
Moon rocks;Moon rocks keyword notes
```

## 키워드를 포함하는 XML 예제

```
<?xml version="1.0" encoding="UTF-8"?>
<keywords>
  <keyword name="Klyngon">
    <notes>
      <![CDATA[Klyngon keyword notes]]>
    </notes>
  </keyword>
  <keyword name="Moon rocks">
    <notes>
      <![CDATA[Moon rocks keyword notes]]>
    </notes>
  </keyword>
</keywords>
```

## 10.2 테스트 프로젝트 내보내기/가져오기

사용자는 프로젝트에 대한 간략한 설명, 테스트 명세와 키워드가 포함된 상태로 테스트 프로젝트를 가져오거나 내보내기 할 수 있다. 다음 두 그림이 트리 구조의 데이터와 동일한 데이터를 XML 파일로 보여준다.



```
<?xml version="1.0" encoding="UTF-8"?>
<testsuite name="">
  <details><![CDATA[]]></details>
  <testsuite name="Communications">
    <details><![CDATA[<p>Communication Systems of all types</p>]]></details>
    <testsuite name="Handheld devices">
      <details><![CDATA[]]></details>
      <testcase name="10 G shock">
        <summary><![CDATA[]]></summary>
        <steps><![CDATA[]]></steps>
        <expectedresults><![CDATA[]]></expectedresults>
      </testcase>
      <testcase name="Gamma Ray Storm">
        <summary><![CDATA[]]></summary>
        <steps><![CDATA[]]></steps>
        <expectedresults><![CDATA[]]></expectedresults>
      </testcase>
    </testsuite>
    <testsuite name="Subspace channels">
      <details><![CDATA[<p>Only basic subspace features</p>]]></details>
      <testcase name="Black hole test">
        <summary><![CDATA[]]></summary>
        <steps><![CDATA[]]></steps>
        <expectedresults><![CDATA[]]></expectedresults>
      </testcase>
    </testsuite>
  </testsuite>
```

```

<testsuite name="Holodeck">
  <details><![CDATA[]]></details>
  <testcase name="Light settings">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
  </testcase>
</testsuite>
<testsuite name="Propulsion Systems">
  <details><![CDATA[]]></details>
  <testsuite name="Main engine">
    <details><![CDATA[]]></details>
    <testcase name="Emergency stop">
      <summary><![CDATA[]]></summary>
      <steps><![CDATA[]]></steps>
      <expectedresults><![CDATA[]]></expectedresults>
    </testcase>
  </testsuite>
</testsuite>
</testsuite>

```

### 10.3 테스트 스위트 가져오기/내보내기



#### XML 예제 - 키워드가 없는 테스트 스위트

```

<?xml version="1.0" encoding="UTF-8"?>
<testsuite name="Handheld devices">
  <details><![CDATA[]]></details>
  <testcase name="10 G shock">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
  </testcase>
  <testcase name="Gamma Ray Storm">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
  </testcase>
</testsuite>

```

#### XML 예제 - 키워드를 가진 테스트 스위트

```

<?xml version="1.0" encoding="UTF-8"?>
<testsuite name="Handheld devices">
  <details><![CDATA[]]></details>
  <testcase name="10 G shock">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
    <keywords>
      <keyword name="Klyngon">
        <notes><![CDATA[Klyngon keyword notes]]></notes>
      </keyword>
    </keywords>
  </testcase>
  <testcase name="Gamma Ray Storm">
    <summary><![CDATA[]]></summary>

```

```

<steps><![CDATA[]]></steps>
<expectedresults><![CDATA[]]></expectedresults>
<keywords>
  <keyword name="Klyngon">
    <notes><![CDATA[Klyngon keyword notes]]></notes>
  </keyword>
  <keyword name="Moon rocks">
    <notes><![CDATA[Moon rocks keyword notes]]></notes>
  </keyword>
</keywords>
</testcase>
</testsuite>

```

#### 10.4 단 하나의 테스트 케이스

ID 78 :: Test Case Black hole test

Version 1

Summary

This procedure must be done once a week, with this safety device disabled:

1. X45HH  
2. YY89-000-JI

Steps

Preset bias to 0  
Enable long range communications control  
Simulate black hole interference

Expected Results

Main Results	
Spin value	9.9
Opposite Angle	18 rad

Keywords: Moon rocks

Created on 27/07/2007 15:16:52 by admin  
Last modified on 27/07/2007 16:16:33 by admin

#### XML 파일 예제:

```

<?xml version="1.0" encoding="UTF-8"?>
<testcases>
  <testcase name="Black hole test">
    <summary>
      <![CDATA[<p>This procedure must be done once a week, with this safety device disabled:</p>
        <ol><li>X45HH</li><li>YY89-000-JI</li></ol>]]>
    </summary>
    <steps><![CDATA[
      <p>Preset bias to 0</p>
      <p>Enable <strong>long range</strong> communications control</p>
      <p><b>Simulate black hole interference</b></p>]]> </steps>
    <expectedresults><![CDATA[
      <table width="200" cellpadding="1" cellspacing="1" border="1">
        <caption>Main Results</caption>
        <tbody>
          <tr><td>Spin value</td><td>9.9</td></tr>
          <tr><td>Opposite Angle</td><td>18 rad</td></tr>
          <tr><td>&nbsp;</td><td>&nbsp;</td></tr>
        </tbody>
      </table>]]>
    </expectedresults>
    <keywords>
      <keyword name="Moon rocks">
        <notes><![CDATA[Moon rocks keyword notes]]></notes>
      </keyword>
    </keywords>
  </testcase>
</testcases>

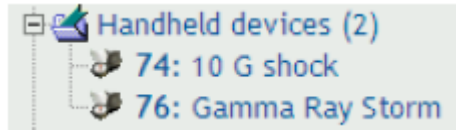
```

```

</keyword>
</keywords>
</testcase>
</testcases>

```

## 10.5 테스트 스위트 안의 모든 테스트 케이스

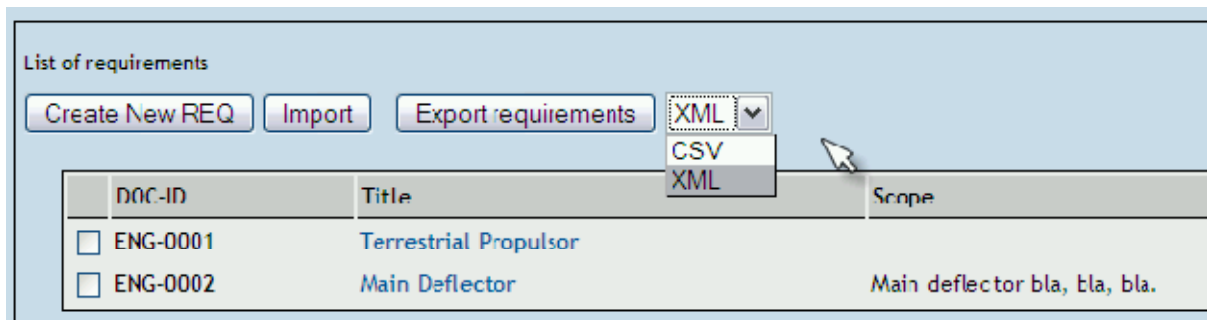


```

<?xml version="1.0" encoding="UTF-8"?>
<testcases>
  <testcase name="10 G shock">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
  </testcase>
  <testcase name="Gamma Ray Storm">
    <summary><![CDATA[]]></summary>
    <steps><![CDATA[]]></steps>
    <expectedresults><![CDATA[]]></expectedresults>
  </testcase>
</testcases>

```

## 10.6 소프트웨어 요구사항 импорт/엑스포트



CSV 파일은 “문서 식별자, 제목, 간략한 설명” 등을 포함한다.

### CSV 파일 예제:

```

ENG-0001, Terrestrial Propulsor,
ENG-0002, Main Deflector, "<p>Main deflector bla, bla, bla.</p>"

```

### XML 파일 예제:

```

<?xml version="1.0" encoding="UTF-8"?>
<requirements>
  <requirement>
    <docid><![CDATA[ENG-0001]]></docid>
    <title><![CDATA[Terrestrial Propulsor]]></title>
    <description><![CDATA[]]></description>
  </requirement>

```

```

<requirement>
  <docid><![CDATA[ENG-0002]]></docid>
  <title><![CDATA[Main Deflector]]></title>
  <description><![CDATA[<p>Maindeflector bla, bla, bla.</p>]]></description>
</requirement>
</requirements>

```

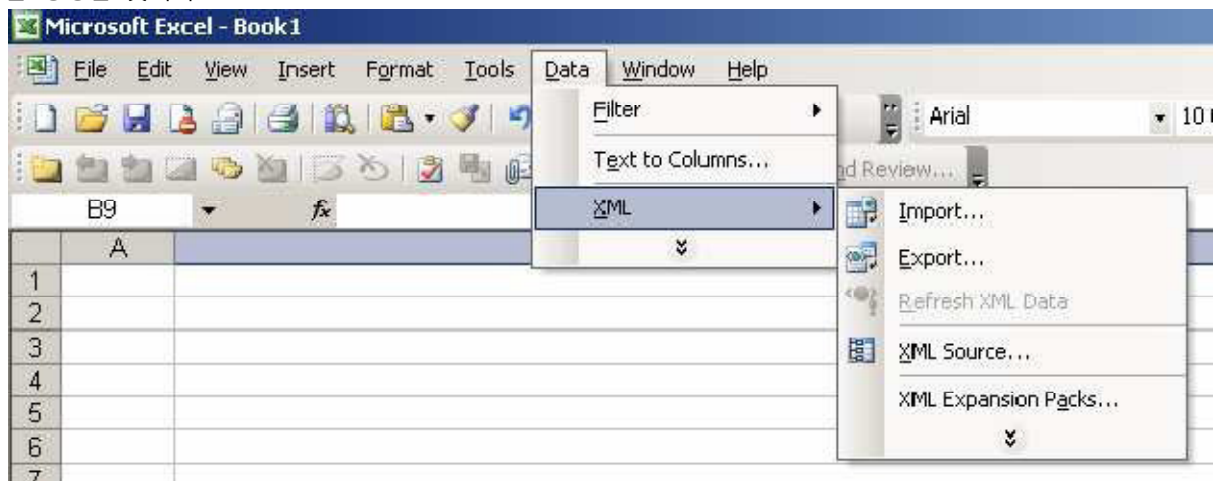
## 10.7 XML을 통해 엑셀로부터 테스트 케이스 불러오기

### 테스트링크에서 불러오기 위해 XML 파일 만들기

Step 1. 하나 혹은 그 이상의 더미 테스트 케이스를 테스트 링크로부터 XML 파일로 생성하라.

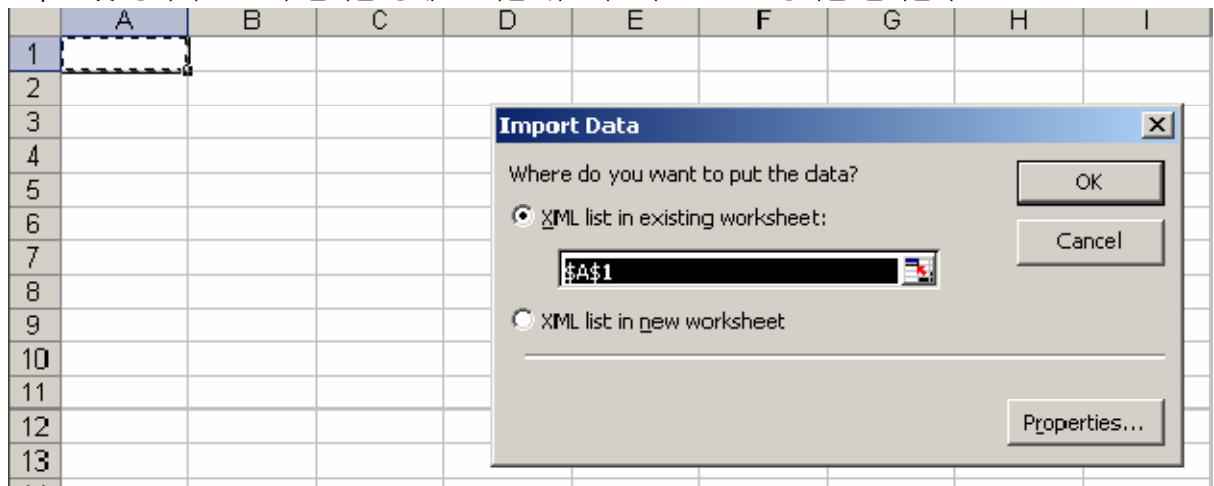
Step 2. 하나의 새 스프레드 시트 문서를 열어서.

Step 3. 엑셀의 메뉴 바에서 Data > XML > Import 를 샘플 XML 파일을 선택한다. 이는 엑셀에 적합한 구조를 생성할 것이다.



Step 4. 그 이후 “데이터가 들어갈 위치를 선택하십시오”라는 대화상자가 뜨는지 확인한다.

Step 5. 첫 항목의 \$A\$1가 선택된 상태로 “기존 워크시트의 XML 표” 항목을 선택한다.



Step 6. 당신은 다음과 같은 항목을 확인할 수 있을 것이다: 제목(Name), 요약(Summary), 순서와 기대결과(Steps & expected results)



	A	B	C	D
1	name	summary	steps	expectedresults
			user has logged in the application.	On navigating to 'Accounts' screen, the following view should be displayed.
2	ACC 1.1	Whether the user can view	Navigate to 'Accounts' screen by clicking on the Accounts screen tab.	'My Accounts List' view (By default this view should be displayed). 'All Accounts List' view.
				be loaded.
3	ACC 1.2	Whether the required applet	i) Click on the 'Accounts' screen. ii) Check for applets	It should display all the following applets:
4	*			
5				

Step 7. 당신의 데이터를 이 시트의 형식에 맞게 복사하고 XML 데이터 형식(\*.xml)으로 저장하라.

Step 8. 인터넷 익스플로러로 해당 XML 파일을 열어 정확한지 확인해 보라.

```

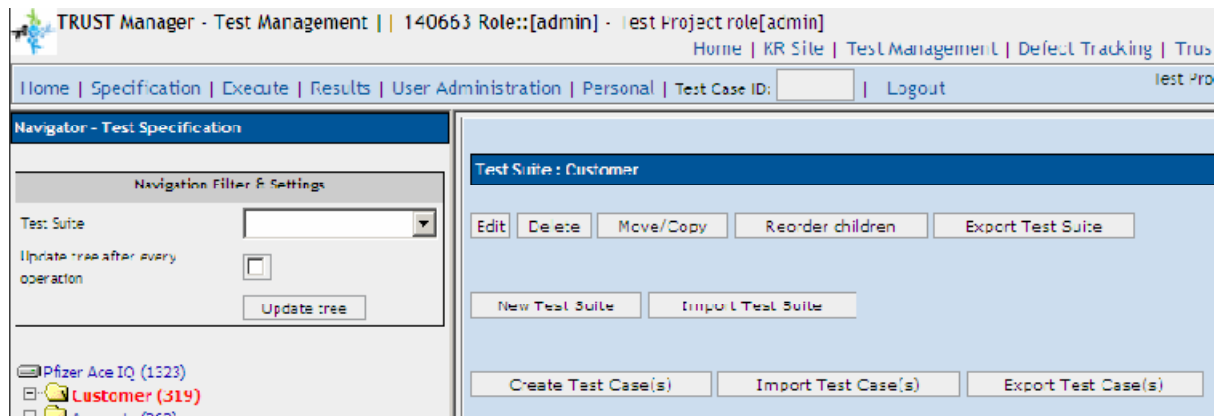
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <testcases>
- <testcase name="ACC 1.1">
  <summary>Whether the user can view Accounts List View as
    a default view under Accounts screen.</summary>
  <steps>Pre-condition: A valid user has logged in the
    application. Navigate to 'Accounts' screen by clicking on
    the Accounts screen tab.</steps>
  <expectedresults>On navigating to 'Accounts' screen, the
    following view should be displayed. 'My Accounts List'
    view (By default this view should be displayed). 'All
    Accounts List' view.</expectedresults>
</testcase>
- <testcase name="ACC 1.2">
  <summary>Whether the user can view Accounts List View as
    a default view under Accounts screen.</summary>
  <steps>Pre-condition: A valid user has logged in the
    application. Navigate to 'Accounts' screen by clicking on
    the Accounts screen tab.</steps>
  <expectedresults>On navigating to 'Accounts' screen, the
    following view should be displayed. 'My Accounts List'
    view (By default this view should be displayed). 'All
    Accounts List' view.</expectedresults>
</testcase>
</testcases>

```

테스트링크에서 XML 파일 가져오기

Step 1. 테스트링크에 로그인 한 후 드롭다운 리스트에서 해당 프로젝트를 선택하라.

Step 2. Specification > Create New Suite > Select Suite > Import Test Case 를 선택한다.



Step 3. XML 파일을 검색해서 선택하고, 승인하면 가져오기 작업이 완료된다.

## Revision History

#	Description	Date	Author
0.x	Documents for TL 1.5 and update for TL 1.6	2005	M. Havlát A. Morsing F. Mancardi
1	Converted to OO2 format;	2005-03-12	M. Havlát
1.1	Minor update; FIX 372, 352	2006-02-14	M. Havlát
1.2	Updated as draft for TL 1.7	2006-11-17	M. Havlát
1.3	Removed TL 1.6 terms Added initial information about Custom Fields	2007-03-01	F. Mancardi
1.4	Added content and updated Francisco's "jumpstart_manual" and tl_file_format.  General style clean-up and update.	2007-09-06	M. Havlát
1.5	General update and restructuring; added Test Suite chapter; requirements report	2007-12-17	M. Havlát
1.6	Overall language review	2008-01-24	W. Pollans
1.7	Minor update; Added section Import Test Cases from Excel via XML (prepared by Prem)	2008-02-02	M. Havlat