

정적분석 도구 및 시스템테스트 도구

Software Verification

Team 5

201410373 고예은

201411266 김수현

201411268 김아름

Index

정적분석 도구 및 시스템테스트 도구



정적분석 도구

시스템테스트 도구

도구 간 연동

CTIP 환경

정적분석 도구 - SonarQube
Static Analysis

Static Analysis

정적 분석



sonarqube



checkstyle



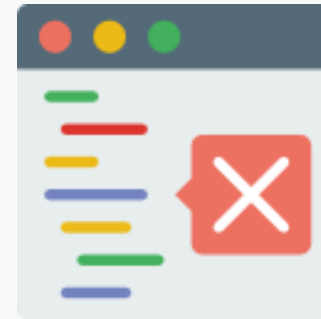
어떤 프로그램을 분석할 때 그 프로그램을 실행시키지 않고 그 자체를 분석하는 것. 프로그램에 내재한 논리적 오류는 보통 프로그램을 실행하여 확인하지 않으면 찾기가 힘들지만, 정적 분석은 이러한 오류를 찾아내는 데 도움을 줄 수 있다.

Static Analysis

정적 분석



자동화된 툴



실행되지 않는
코드 검출



규칙에 맞지 않는
코드 검출



버그 리포트

Static Analysis

정적 분석

sonarqube 

🔍 여러 분석 툴 통합 가능

📖 다양한 플러그인 제공

📁 직관적이고 쉬운 UI

📍 20개 이상의 언어 지원

Static Analysis

정적 분석

- SonarQube 설치

The screenshot shows the SonarQube website with a modal dialog open. The modal dialog has two columns: "LTS Release" and "Latest Release".

LTS Release
Released every 18 months, Long-Term Supported versions are the most stable releases to which all critical and blocker issues are back-ported.
[↓ 6.7.3](#)
[See more details](#)

Latest Release
Get it while it's hot, this is the latest version of the product. It contains all the new features and it assumes that you will upgrade at each new release.
[↓ 7.0](#)
[See what's new](#)

Both LTS and the latest version are [commercially supported](#) by SonarSource.

The background of the website shows the SonarQube logo, navigation links (FEATURES, DOWNLOADS, ROADMAP, COMMUNITY, BLOG), a banner for SonarQube 7.0, and a "Vulnerabilities" section with a lock icon. At the bottom, there are links for "On 20+ Code Analyzers" and supported languages: Java, JavaScript, C#, C/C++, COBOL, and an "AND MORE" button.

Static Analysis

정적 분석

- SonarQube 설치 - install & service start

```
svsvsv@suhyun-sv:~$ sudo apt-get install sonar
```

```
svsvsv@suhyun-sv:~$ sudo service sonar start
```

```
Starting SonarQube...
```

```
Started SonarQube.
```


Static Analysis

정적 분석

- SonarQube DB 설정 - MySQL

- > CREATE DATABASE sonar CHARACTER SET utf8 COLLATE utf8_general_ci;
- > CREATE USER 'sonar' IDENTIFIED BY 'sonar';
- > GRANT ALL ON sonar.* TO 'sonar'@'%' IDENTIFIED BY 'sonar';
- > GRANT ALL ON sonar.* TO 'sonar'@'localhost' IDENTIFIED BY 'sonar';
- > FLUSH PRIVILEGES;

```
mysql> CREATE DATABASE sonar DEFAULT CHARACTER SET utf8 COLLATE utf8_general_ci;
mysql> GRANT ALL PRIVILEGES on sonar.* to 'sonar'@'localhost' IDENTIFIED BY 'sonar' WITH GRANT OPTION;
mysql> flush privileges;
```

Static Analysis

정적 분석

- SonarQube DB 설정 – sonar.properties

```
#-----$
# DATABASE
#
# IMPORTANT:
# - The embedded H2 database is used by default. It is recommended for tests bu$
#   production use. Supported databases are MySQL, Oracle, PostgreSQL and Micro$
# - Changes to database connection URL (sonar.jdbc.url) can affect SonarSource $
#
# User credentials.
# Permissions to create tables, indices and triggers must be granted to JDBC us$
# The schema must be created first.
sonar.jdbc.username=sonar
sonar.jdbc.password=sonar
```

```
#----- MySQL 5.6 or greater
# Only InnoDB storage engine is supported (not myISAM).
# Only the bundled driver is supported. It can not be changed.
sonar.jdbc.url=jdbc:mysql://localhost:3306/sonar?useUnicode=true&characterEncod$
sonar.jdbc.driverClassName=com.mysql.jdbc.Driver
sonar.jdbc.validationQuery=select 1
```

Static Analysis

정적 분석



Static Analysis

정적 분석

다양한 개발 언어 지원

(Java, JavaScript, XML, C/C++, C#, PHP, Ruby, Fortran)

작성한 코드에 대한 위반사항을 찾음

위반 사항을 명시한 report파일에 대한 수정 쉬움

한 번의 클릭으로 수많은 규칙에 대한 수정 가능



Static Analysis

정적 분석



여러 사람과 작업 시 손쉽게 코딩 스타일을 맞출 수 있음

표준 코딩 스타일을 손쉽게 프로젝트에 적용

개발초기에 소스코드의 잠재적 결함 발견

Static Analysis

정적 분석

- SonarQube에서 플러그인 설치

sonarqube 프로젝트 이슈 코딩 규칙 품질 프로파일 품질 게이트 환경설정

환경설정

프로젝트 설정 시큐리티 프로젝트 시스템 마켓플레이스

마켓플레이스

새로운 기능들을 확인하고 설치하십시오

Community Edition ✓ 설치됨

Comes with support for 9 programming languages, numerous plugins, integration with DevOps tool chains, and ability to connect to SonarLint in the IDE.

[더 살펴보기](#)

Developer Edition

Community Edition + branch analysis, SonarLint push notifications, and 16 languages.

[더 살펴보기](#) [업그레이드](#)

Enterprise Edition

Developer Edition + portfolio management, executive reporting, parallel processing of analysis reports and 20 languages.

[더 살펴보기](#) [업그레이드](#)

Data Center Edition

Enterprise Edition + component redundancy and data integrity.

[더 살펴보기](#) [업그레이드](#)

모두 업데이트만

PMD External Analysers 2.6 설치됨

Analyze Java code with [PMD](http://pmd.sourceforge.net/).

[홈페이지](#) [이슈 추적기](#) [제거](#)

GNU LGPL 3(으)로 라이선스 됨
SonarSource이(가) 개발함

Static Analysis

정적 분석

- SonarQube에서 플러그인 설치

마켓플레이스
새로운 기능들을 확인하고 설치하십시오

| | | | |
|---|--|---|---|
| Community Edition ✓ 설치됨 Comes with support for 9 programming languages, numerous plugins, integration with DevOps tool chains, and ability to connect to SonarLint in the IDE. 더 살펴보기 | Developer Edition Community Edition + branch analysis, SonarLint push notifications, and 16 languages. 더 살펴보기 업그레이드 | Enterprise Edition Developer Edition + portfolio management, executive reporting, parallel processing of analysis reports and 20 languages. 더 살펴보기 업그레이드 | Data Center Edition Enterprise Edition + component redundancy and data integrity. 더 살펴보기 업그레이드 |
|---|--|---|---|

모두 업데이트만 ×

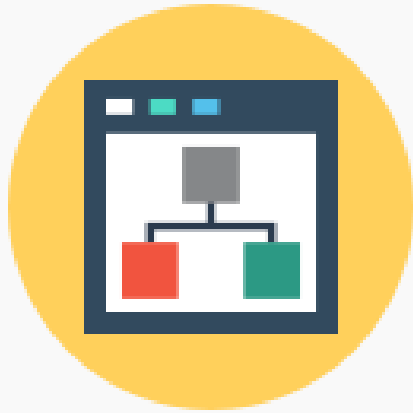
| | | |
|--|--|--------------------|
| PMD External Analysers 2.6 설치됨 Analyze Java code with PMD . | 홈페이지 이슈 추적기 GNU LGPL 3(으)로 라이선스 됨 SonarSource 이(가) 개발함 | 제거 |
| Checkstyle External Analysers 4.8 설치됨 Analyze Java code with Checkstyle | 이슈 추적기 LGPL-3.0(으)로 라이선스 됨 Checkstyle 이(가) 개발함 | 제거 |
| Findbugs External Analysers 3.7.0 설치됨 Analyze Java, Scala, Closure and JSP code with SpotBugs. | 홈페이지 이슈 추적기 GNU LGPL 3(으)로 라이선스 됨 SpotBugs Team 이(가) 개발함 | 제거 |

시스템테스트 도구 - Testlink

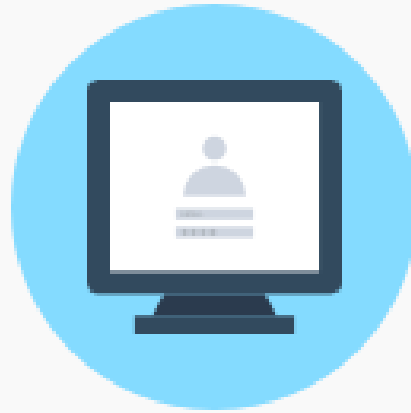
System Test

System Test

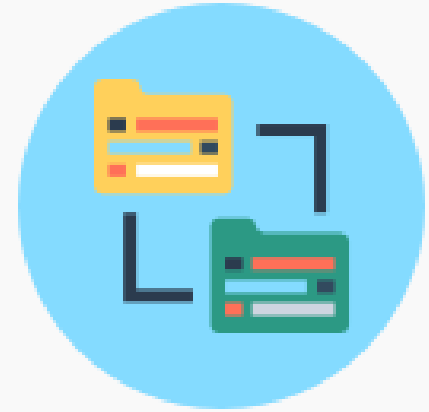
시스템 테스트



시스템 전체의
정상 작동 확인



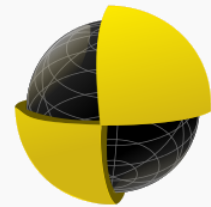
실제 사용환경과 유사한
테스트 환경 조성



모듈 통합 후
요구사항 만족 여부 체크

System Test

시스템 테스트



TestLink

🔍 테스트 우선순위 부여

📖 버그 시스템 직접연동

👤 사용자 활동영역 정의

✈️ 요구사항과 TC 매핑

System Test

시스템 테스트

- Testlink 설치

```
[ec2-user@ip-172-31-22-165 ~]$ wget https://github.com/TestLinkOpenSourceTRMS/testlink-code/archive/1.9.16.tar.gz
```

System requirements

| | |
|---|---|
| Server Operating System (no constrains) | Linux |
| PHP version | OK (5.5.0 [minimum version] <= 5.6.35 [your version]) |

Web and PHP configuration

| | |
|---|--|
| Maximum Session Idle Time before Timeout | 24 minutes and 0 seconds - (Short. Consider to extend.) |
| Checking max. execution time (Parameter max_execution_time) | 30 seconds - We suggest 120 seconds in order to manage hundred of test cases (edit php.ini) |
| Checking maximal allowed memory (Parameter memory_limit) | OK (128 MegaBytes) |
| Checking if Register Globals is disabled | OK |
| Checking MySQL Database | OK |
| Checking Postgres Database | Failed! Postgres Database cannot be used. |
| Checking MSSQL Database | Failed! MSSQL Database cannot be used. |
| Checking GD Graphic library | Failed! GD Graphic library not enabled. Graph rendering requires it. This feature will be disabled. It's recommended to install it.. |
| Checking LDAP library | Failed! LDAP library not enabled. LDAP authentication cannot be used. (default internal authentication will works). |
| Checking JSON library | OK |
| Checking cURL library | OK |

```
TestLink 1.9.16 (Moka pot) TestLink 1.9.16 (Moka pot) - New installation

TestLink setup will now attempt to setup the database:

Creating connection to Database Server:OK!

Connecting to database `testlink` :OK!
Creating Testlink DB user `testlink` :OK! ( ok - new user)
Processing:sql/mysql/testlink_create_tables.sql

-----
DB Access Error - debug_print_backtrace() OUTPUT START

ATTENTION: Enabling more debug info will produce path disclosure weakness (CVE-200)
            Having this additional information could be useful for reporting
            issue to development TEAM.

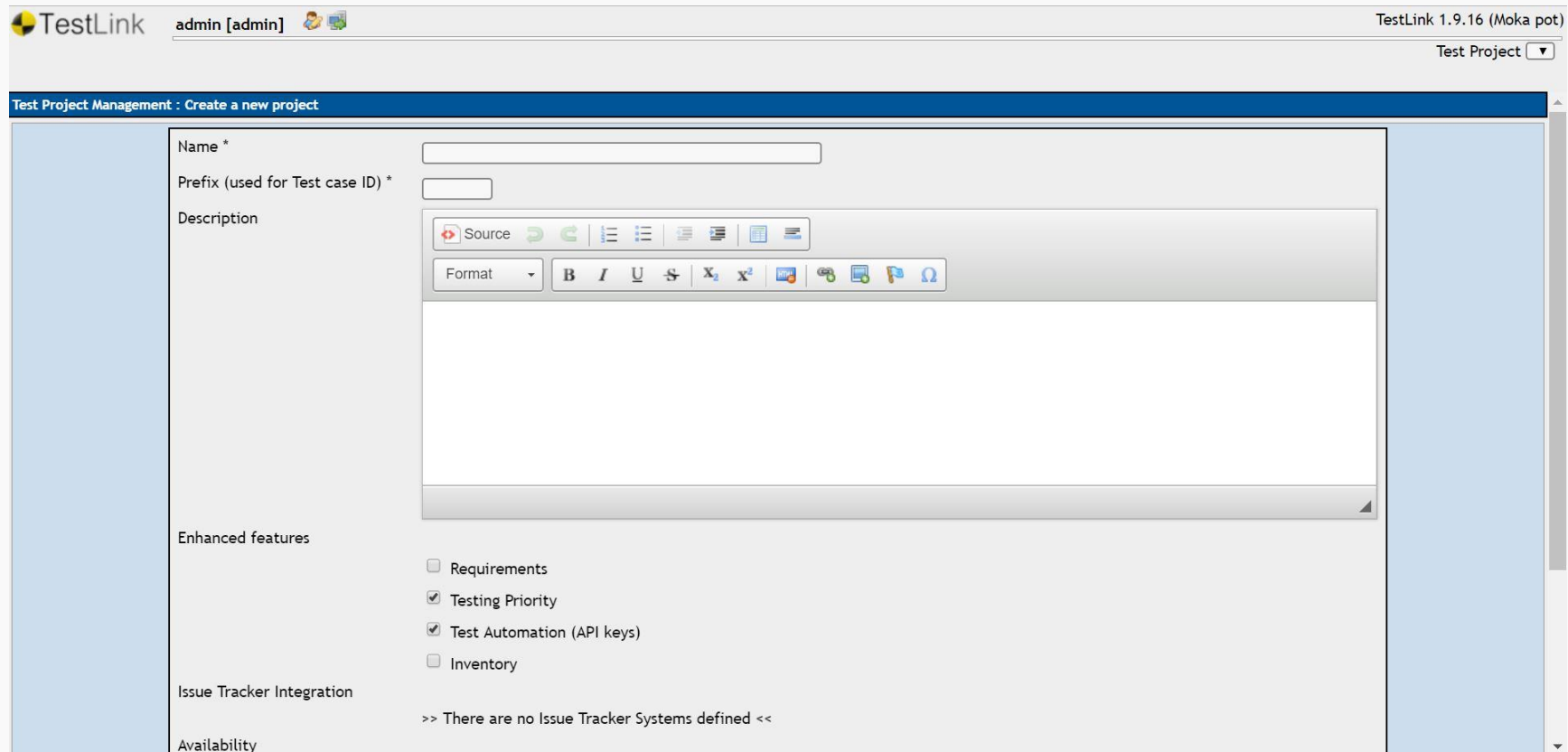
-----

#0 database->exec_query(CREATE TABLE /*prefix*/inventory (
  id int(10) unsigned NOT NULL auto_increment,
  `testproject_id` INT( 10 ) UNSIGNED NOT NULL ,
  `owner_id` INT(10) UNSIGNED NOT NULL ,
  `name` VARCHAR(255) NOT NULL ,
  `ipaddress` VARCHAR(255) NOT NULL ,
  `content` TEXT NULL ,
  `creation_ts` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `modification_ts` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY /*prefix*/inventory_idx1 (`testproject_id`)
) DEFAULT CHARSET=utf8) called at [/var/www/html/testlink-code-1.9.16/install/sqlParser.class.php:103]
#1 SqlParser->process(sql/mysql/testlink_create_tables.sql) called at [/var/www/html/testlink-code-1.9.16/install/installNewDB.php:427]
```

System Test

시스템 테스트

- Testlink 프로젝트 생성



The screenshot shows the TestLink web interface for creating a new project. The page title is "Test Project Management : Create a new project". The user is logged in as "admin [admin]". The version is "TestLink 1.9.16 (Moka pot)".

The form fields include:

- Name *
- Prefix (used for Test case ID) *
- Description (with a rich text editor toolbar containing options like Source, Format, Bold, Italic, Underline, etc.)
- Enhanced features (checkboxes for Requirements, Testing Priority, Test Automation (API keys), and Inventory)
- Issue Tracker Integration (with a message: ">> There are no Issue Tracker Systems defined <<")
- Availability

System Test

시스템 테스트

- Testlink 요구사항 관리

The screenshot displays the TestLink 1.9.16 (Moka pot) interface. The top navigation bar shows the user 'admin [admin]' and the current 'Test Project' as 'testTest'. The main content area is titled 'Create Requirements Specification Test Project : Test'. On the left, a 'Navigator - Requirement Specifications' sidebar contains 'Settings' (with 'Update tree after every operation' checked) and 'Filters' (with various dropdown menus for Document ID, Title, Status, Requirement Type, Req. Spec. Type, Expected no. of testcases, Has relation of type, and Test Case ID). The main form includes 'Document ID' (test1) and 'Title' (Requirement1) fields. Below these is a 'Scope' section with a rich text editor containing the text '요구사항 test'. The 'Type' dropdown is set to 'Section'. 'Save' and 'Cancel' buttons are present at the top and bottom of the form.

System Test

시스템 테스트

- Testlink 요구사항 관리

The screenshot displays the Testlink web interface for managing requirements. The browser title bar shows "Req. Spec. : [test1] :: Requirement1". The main content area includes a gear icon, a title bar "Req. Spec. : [test1] :: Requirement1", and details for "revision : 1" of type "Section". A "Scope" section contains the text "요구사항 test". Below this, it states "Created on 16/04/2018 18:56:50 by admin". At the bottom, there is an "Attached files" section with a "File" button, a "파일 선택" button, the text "선택된 파일 없음", a "Title/name:" input field, and an "Upload file" button.

Jenkins - SonarQube

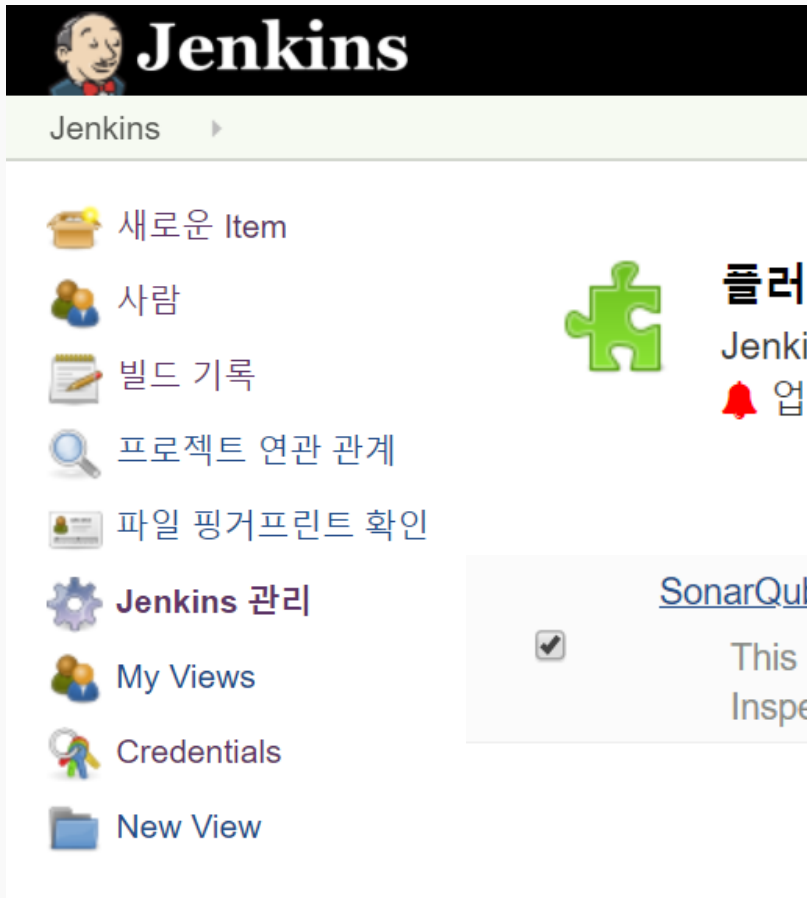
Redmine - Testlink

CTIP 도구 연동

Jenkins - SonarQube

CTIP 도구 연동

- Jenkins – SonarQube Scanner 플러그인 설치




The screenshot shows the Jenkins dashboard interface. At the top left is the Jenkins logo and name. Below it is a navigation menu with the following items: 새로운 Item, 사람, 빌드 기록, 프로젝트 연관 관계, 파일 핑거프린트 확인, Jenkins 관리, My Views, Credentials, and New View.



플러그인 관리

Jenkins의 기능을 확장하기 위한 플러그인을 추가, 제거, 사용, 미사용으로 설정할 수 있습니다.

 업데이트 가능함

SonarQube Scanner for Jenkins



This plugin allows an easy integration of [SonarQube](#), the open source platform for Continuous Inspection of code quality.

[2.7.1](#)

Jenkins - SonarQube

CTIP 도구 연동

- Jenkins – 시스템 설정

Jenkins 관리



시스템 설정

환경변수 및 경로 정보등을 설정합니다.

SonarQube servers

Environment variables

Enable injection of SonarQube server configuration as build environment variables

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

SonarQube installations

Name

Sonarqube

Server URL

http://13.82.40.242:9000/

Default is http://localhost:9000

Server authentication token

.....

SonarQube authentication token. Mandatory when anonymous access is disabled.

고급...

Delete SonarQube

Jenkins - SonarQube

CTIP 도구 연동

- Jenkins – Global Tool Configuration



Global Tool Configuration

SonarQube Scanner

SonarQube Scanner installations

Add SonarQube Scanner

SonarQube Scanner

Name

Install automatically [?](#)

Install from Maven Central

Version

Delete Installer

Jenkins - SonarQube

CTIP 도구 연동

- Jenkins – 프로젝트 – Add build step

The image shows the Jenkins configuration page for the 'Execute SonarQube Scanner' build step. The page has a tabbed interface with 'Build' selected. The configuration fields are as follows:

- Task to run:** An empty text input field.
- JDK:** A dropdown menu with 'JAVA_HOME' selected.
- JDK to be used for this SonarQube analysis:** A label indicating the selected JDK.
- Path to project properties:** A text input field containing 'sonar-project.properties'.
- Analysis properties:** A large, empty text area for additional analysis properties.
- Additional arguments:** A text input field with a dropdown arrow on the right.
- JVM Options:** A text input field with a dropdown arrow on the right.

At the bottom left, there is a button labeled 'Add build step' with a dropdown arrow. A red 'X' button is visible in the top right corner of the configuration area.

Jenkins - SonarQube

CTIP 도구 연동

- sonar-project.properties 내용

```
GNU nano 2.8.7 File: sonar-project.properties
sonar.projectKey=Test
sonar.projectName=Test
sonar.projectVersion=1.0

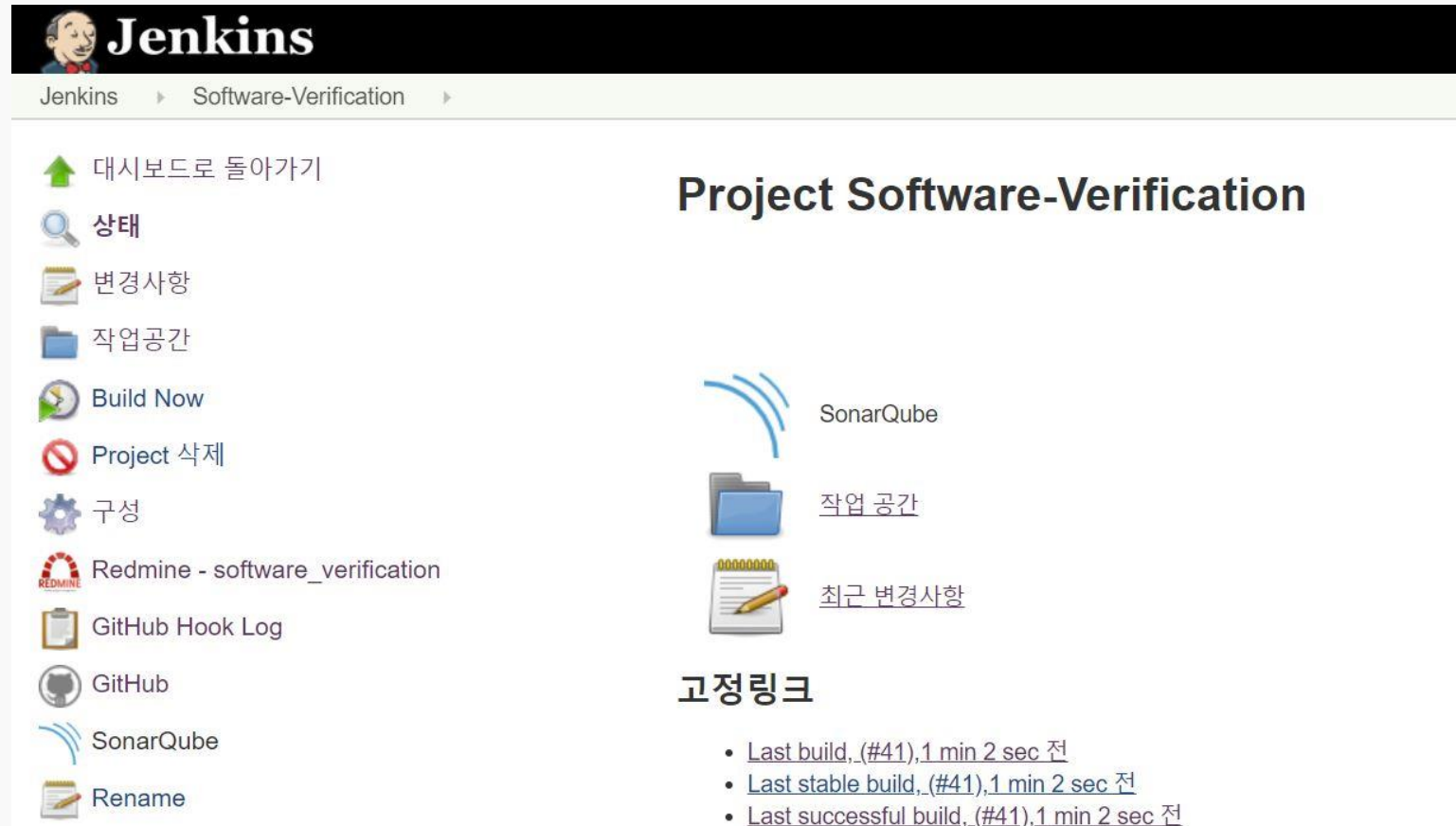
sonar.sources=Test/src/main/java, Test/src/test/java
sonar.java.binaries=Test/bin

sonar.sourceEncoding=UTF-8
```

Jenkins - SonarQube

CTIP 도구 연동

- Build 성공 후 Jenkins



The screenshot shows the Jenkins web interface for a project named 'Software-Verification'. The interface includes a navigation menu on the left with options like '대시보드로 돌아가기', '상태', '변경사항', '작업공간', 'Build Now', 'Project 삭제', '구성', 'Redmine - software_verification', 'GitHub Hook Log', 'GitHub', 'SonarQube', and 'Rename'. The main content area displays the project name 'Project Software-Verification' and three icons: 'SonarQube', '작업 공간', and '최근 변경사항'. Below these icons, there is a section titled '고정링크' (Fixed Links) with three bullet points: 'Last build,_(#41),1 min 2 sec 전', 'Last stable build,_(#41),1 min 2 sec 전', and 'Last successful build,_(#41),1 min 2 sec 전'.

Jenkins - SonarQube

CTIP 도구 연동

- Build 성공 후 SonarQube

The screenshot displays the SonarQube dashboard for a project named 'Test'. The interface includes a top navigation bar with the SonarQube logo and menu items like '프로젝트', '이슈', '코딩 규칙', '품질 프로파일', '품질 게이트', and '환경설정'. A search bar is located on the right side of the top bar. Below the navigation bar, there are filters for 'My Favorites' and '모두'. The main content area shows the project 'Test' with a 'Success' status. Key metrics are displayed: 0 Bugs (Grade A), 0 취약점 (Grade A), 2 코드 약취 (Grade B), 0.0% Coverage, and 0.0% Duplications. The last analysis was performed on 2018년 4월 17일 오전 3:35. The project is identified as '13 XS Java'. A message at the bottom indicates '1 중 1이(가) 표시되었습니다.'

sonarqube 프로젝트 이슈 코딩 규칙 품질 프로파일 품질 게이트 환경설정

프로젝트, 서브 프로젝트 및 파일로 검색할 수 ? A

관점: 전체 상태 정렬 기준: 이름 프로젝트 이름 혹은 키로 검색할 수 있습니다. 1 projects

Filters

Quality Gate

Success 1

Warning 0

Failed 0

신뢰성 (Bugs)

A 1

B 좋지 않음 0

☆ Test Success

0 A Bugs

0 A 취약점

2 B 코드 약취

0.0% Coverage

0.0% Duplications

13 XS Java

마지막 분석: 2018년 4월 17일 오전 3:35

1 중 1이(가) 표시되었습니다.

Redmine - Testlink

CTIP 도구 연동

- Testlink – Issue Tracker 등록

TestLink admin [admin] test-

Issue Tracker Management

Create

| | |
|-----------------------|---|
| Issue Tracker | redmine |
| Type | redmine (Interface: rest) Show configuration example |
| Configuration | <pre><issuetracker> <dbhost>http://18.220.47.156:3306</dbhost> <dbname>bitnami_redmine</dbname> <dbtype>mysql</dbtype> <dbuser>root</dbuser> <dbpassword>redmine</dbpassword> <apikey>Jp9x075Ce5pGfz2lKdW</apikey> <uribase>http://18.220.47.156/</uribase> <uriview>http://18.220.47.156/issues/</uriview> </issuetracker></pre> |
| Configuration example | |

Save Check connection Cancel

- Testlink – Project와 연동

TestLink admin [admin] test-

Enhanced features

- Requirements
- Testing Priority
- Test Automation (API keys)
- Inventory

Issue Tracker Integration

- Active

Issue Tracker redmine (redmine (Interface: rest))

Availability

- Active
- Public

API Key d55a059d01adff369e272884cebaff079ff0a3c2d2425dcf3fc804cd437292f

Save Cancel

* Mandatory fields

Team 5
CTIP 환경

CTIP 환경

Jenkins

- Jenkins 연결 (Redmine)

Redmine

Redmine websites

Name

Redmine

Base url

http://18.220.47.156/

Version number

3.4.4



삭제

Assign Redmine project

Redmine website

Redmine (http://18.220.47.156/)



Redmine project identifier

software_verification



CTIP 환경

Jenkins

- Jenkins 연결 (Github)

GitHub Server

Name: Software Verification

API URL: https://api.github.com

Credentials: shyun-ab

Manage hooks

소스 코드 관리

None
 Git

Repositories

Repository URL: https://github.com/shyun-ab/Software_Verification.git

Credentials: shyun-ab/***** [Add](#)

[고급...](#)

[Add Repository](#)

Branches to build

Branch Specifier (blank for 'any'): */master

[Add Branch](#)

Repository browser: (자동)

CTIP 환경

Jenkins

- Jenkins 연결 (Build - Gradle)

Build


Invoke Gradle script ✕ ?

Invoke Gradle ?

Gradle Version ▼

Use Gradle Wrapper ?

Tasks ?

 고급...

Add build step ▼

CTIP 환경

Jenkins

- Jenkins 연결 (Build step - SonarQube)

SonarQube servers

Environment variables Enable injection of SonarQube server configuration as build environment variables

If checked, job administrators will be able to inject a SonarQube server configuration as environment variables in the build.

SonarQube installations

| | |
|-----------------------------|--|
| Name | <input type="text" value="Sonarqube"/> |
| Server URL | <input type="text" value="http://13.82.40.242:9000/"/> |
| | Default is http://localhost:9000 |
| Server authentication token | <input type="text" value="....."/> |
| | SonarQube authentication token. Mandatory when anonymous access is disabled. |

CTIP 환경

Jenkins

- Jenkins 프로젝트 화면

Jenkins

Jenkins > Software-Verification

대시보드로 돌아가기

상태

변경사항

작업공간

Build Now

Project 삭제

구성

Redmine - software_verification

GitHub Hook Log

GitHub

SonarQube

Rename

Project Software-Verification

SonarQube

작업 공간

최근 변경사항

고정링크

- [Last build, \(#41\), 1 min 2 sec 전](#)
- [Last stable build, \(#41\), 1 min 2 sec 전](#)
- [Last successful build, \(#41\), 1 min 2 sec 전](#)

CTIP 환경

Github

- Github 프로젝트

The screenshot shows the GitHub interface for the repository 'shyun-ab / Software Verification'. At the top, there are buttons for 'Watch' (0), 'Star' (0), and 'Fork' (0). Below this is a navigation bar with 'Code', 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', 'Insights', and 'Settings'. The main content area shows 'No description, website, or topics provided.' with an 'Edit' button. Below this is a progress bar showing '14 commits', '1 branch', '0 releases', and '1 contributor'. At the bottom, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. A list of files is shown below, including 'Test', 'hooks', 'README.md', 'Redmine.RefreshRepository.vbs', and 'build.gradle', each with its commit message and time since last update.

shyun-ab / [Software Verification](#) Watch 0 Star 0 Fork 0

[Code](#) Issues 0 Pull requests 0 Projects 0 Wiki Insights Settings

No description, website, or topics provided. Edit

[Add topics](#)

🕒 14 commits 🌿 1 branch 📦 0 releases 👤 1 contributor

Branch: master ▾ New pull request Create new file Upload files Find file Clone or download ▾

| | |
|---------------------------------|------------------------------------|
| shyun-ab build test' | Latest commit 54d324c 22 hours ago |
| 📁 Test | build test' 22 hours ago |
| 📁 hooks | hook post-commit test 13 days ago |
| 📄 README.md | Update README.md 16 days ago |
| 📄 Redmine.RefreshRepository.vbs | hook post-commit test 13 days ago |
| 📄 build.gradle | Update build.gradle a day ago |

CTIP 환경

Redmine

- Redmine 프로젝트

초기화면 [내 페이지](#) [프로젝트 관리](#) [도움말](#)




Software_Verification

검색:

+ [개요](#) [작업내역](#) [일감](#) [소요 시간](#) [Gantt 차트](#) [달력](#) [뉴스](#) [문서](#) [위키](#) [파일](#) [저장소](#) **Jenkins** [설정](#)

List of jobs

 Refresh the list of jobs

| Name | Latest build | State | Duration | Latest changes | History |
|---------------------------------------|-------------------------------|----------------|----------|----------------|--|
| Software-Verification | #41 (2018/04/04 21:37) | Success | 00:10 | |    |

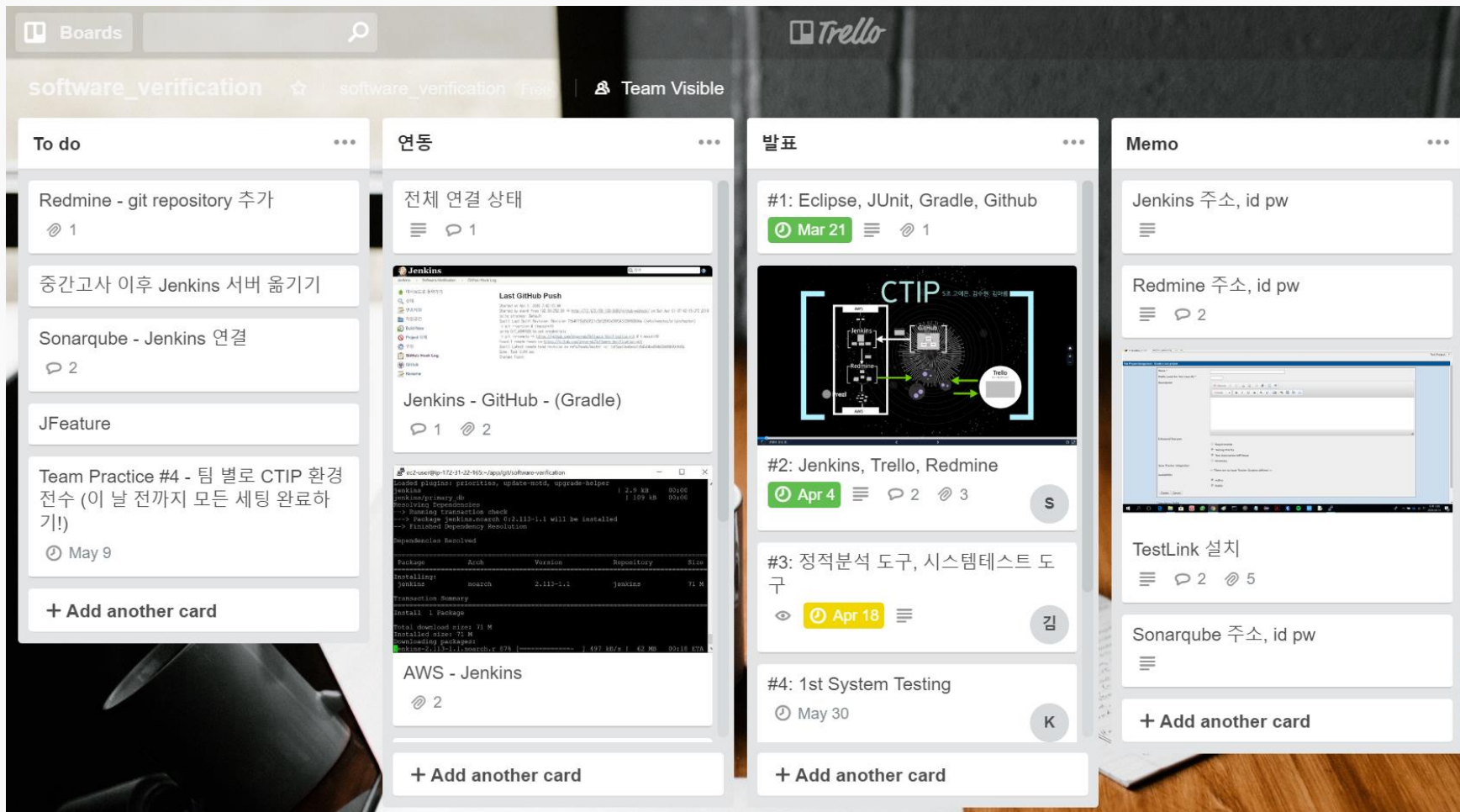
Build stability: 4 out of the last 5 builds failed.



CTIP 환경

Trello

- Trello – 팀원 간 요구사항 관리



CTIP 환경

Summary

amazon web services™ **EC2**

amazon web services™ **EC2**



gradle
Jenkins



TestLink




REDMINE
flexible project management



GitHub

sonarqube



Pmd
DON'T SHOOT THE MESSENGER



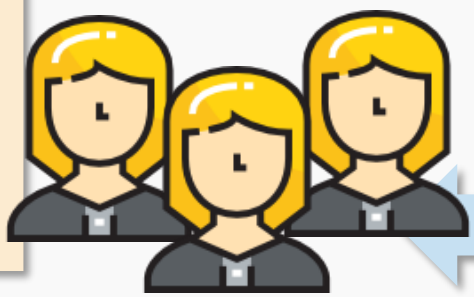
checkstyle



FindBugs

JUnit

eclipse



Trello

Microsoft
Azure

CTIP 환경

Summary

- 개발 – Eclipse
- 유닛 테스트 – Junit
- CI 서버 – Jenkins
- 빌드 – Gradle
- 형상관리 – Github
- 이슈/프로젝트 관리 – Redmine / Trello
- 품질관리 – SonarQube / PMD / Checkstyle / Findbugs
- 시스템 테스트 – TestLink

Thank you.

Q & A