

SOFTWARE

2015.04.17

VERIFICATION

TEAM 4

[정적분석도구 & 시스템테스트 도구]

컴퓨터 공학부

200911393 박현규

Presenter → 201010768 **최정한**

201111339 김민우

201211389 함진아

CONTENTS

●-----	PART 1	Purpose -----	3
●-----	PART 2	Static analysis Tools -----	8
●-----	PART 3	System testing Tool -----	58
●-----	PART 4	Make up for CTIP -----	83
●-----	PART 5	Summary -----	93
●-----	✘	References -----	95

PART 1

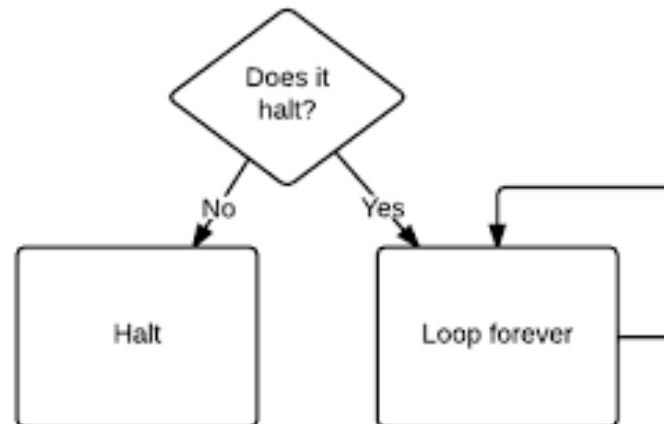
Purpose

PART 1 Purpose

What is Static analysis ?

- 정적 분석(Static analysis)은 프로그램을 실행하지 않고 진행하는 분석이다. 예상되는 문제를 미리 발견하고 해결할 수 있으며, 실제로 실행하기 어려운 분야에 적용되기도 한다.
- 대상은 소스코드 외에도 개발 전반에 걸쳐 할 수 있다.
- 예) code convention 오류, overflow, divide by zero 등을 미리 알 수 있다.

Halting Problem



PART 1 Purpose

Static analysis tools

주요 목적	툴 이름
코딩 표준	Checkstyle
코드 중복	PMD
의존성 분석	Jdepend
버그 패턴	Findbugs

PART 1 Purpose

What is System testing ?

- V&V 모델의 validation에 속하며, 명시된 요구사항을 만족하는지 확인하기 위해 통합된 시스템을 테스트하는 절차이다. 통합된 시스템의 각 부분이 요구사항에 맞게 수행되는지 평가를 하게 된다.

System testing tool



PART 2

Static analysis Tools

PART 2 Static analysis Tools

What is Checkstyle ?

- CheckStyle은 코딩 표준에 맞게 소스코드를 작성하도록 도와주는 도구이다.
- 개발자들이 잘 지키지는 않지만 중요한 것들을 지적해준다 (ex : private 설정)
- 협업시에 코딩스타일을 맞출 수 있고, 잠재적인 결함 발견도 가능하다. 팀의 코딩규칙을 문서로 정리해서 지키기로 하면, 규칙이 많아질 수록 어기기 쉽다.
- 라이선스 : Eclipse Public License 1.0

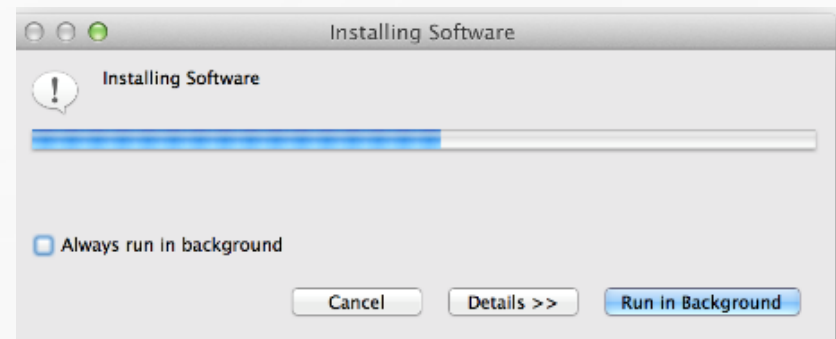
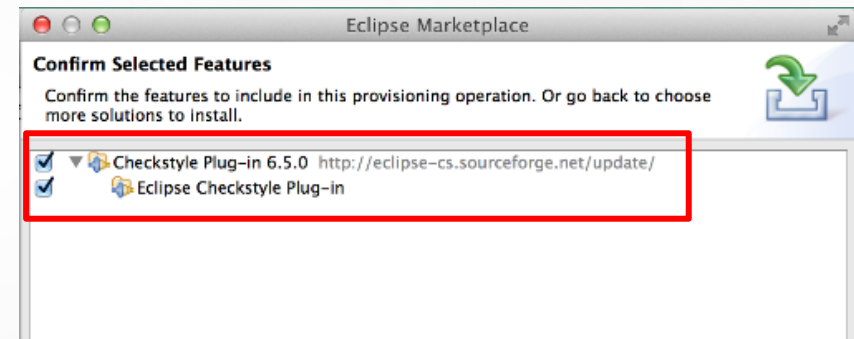
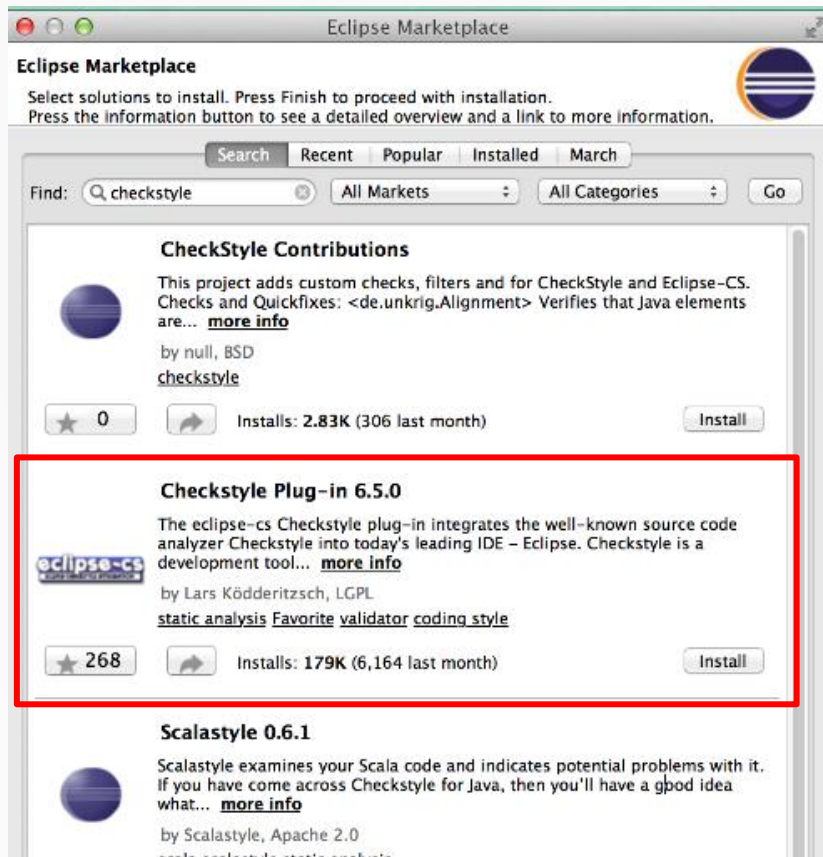
- **주요 분석 목적 : 코딩 표준 준수**

PART 2 Static analysis Tools

Eclipse : Checkstyle plugin

1. Checkstyle plugin 다운

Help → market place → checkstyle 검색 → Checkstyle Plug-in 6.5.0 Install

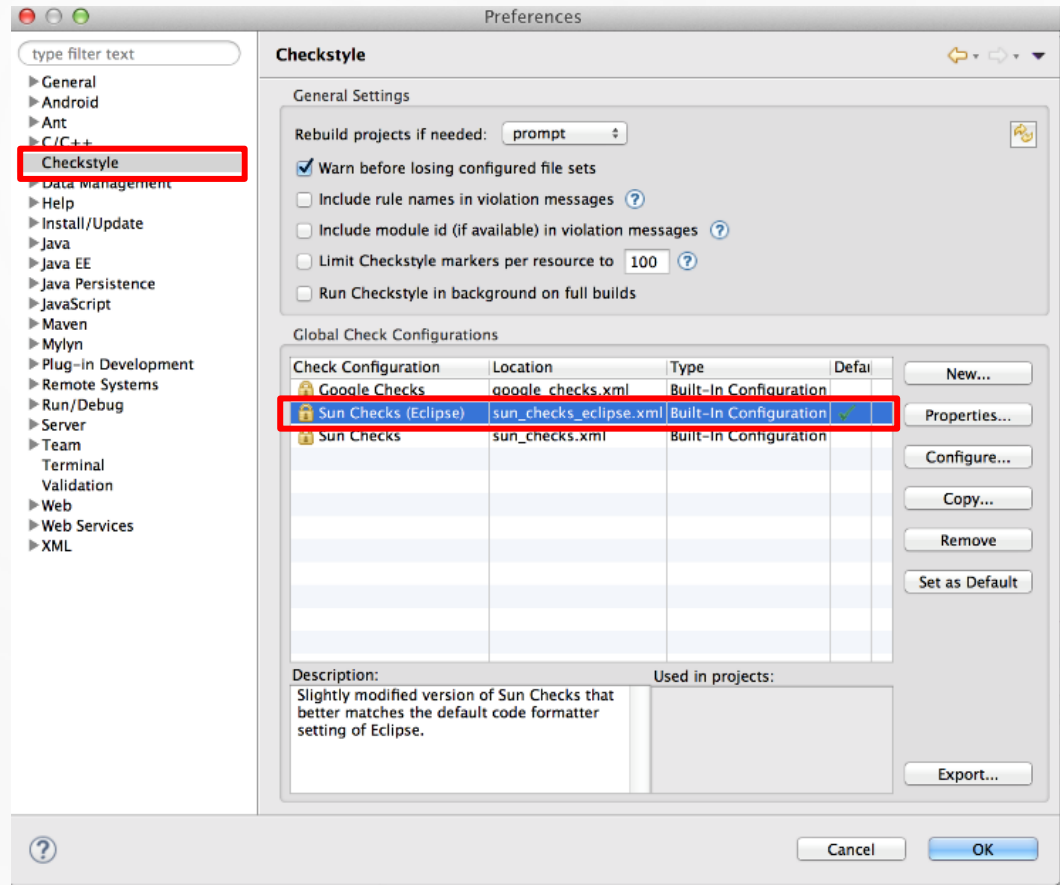


PART 2 Static analysis Tools

Eclipse : Checkstyle plugin

2. Checkstyle 환경 설정

window → preferences → checkstyle → sun check (eclipse) 선택



※ Sun_check_eclipse.xml은
check rule에 대한 xml 파일로,
Check하고자 하는 기능 및 조건과
 Checkstyle tool에 대한 **환경설정**
 내용이 들어 있는 기본 파일이다.

PART 2 Static analysis Tools

Eclipse : Checkstyle plugin

3. Checkstyle 실행

프로젝트에서 우클릭 → Checkstyle → Active Checkstyle → 코드 왼쪽에 guild line 생성

The screenshot illustrates the process of activating the Checkstyle plugin in Eclipse. On the left, the project tree shows the 'Bully' package. The middle pane shows the context menu for the package, with 'Checkstyle' and 'Active Checkstyle' highlighted. The right pane shows the 'FaultDetectionM.java' file with a yellow warning box indicating 'Multiple markers at this line' and listing violations: 'Parameter ip should be final.', 'Missing a Javadoc comment.', 'Method 'checkServer' is not designed for extension - needs to be abstract, final or empty.', and 'Parameter port should be final.'

```

import java.io.ObjectOutputStream;

public class FaultDetectionM extends Thread

public class ServerItem {
    public String IP;
    public int port;
    public int priority;
}

public ArrayList<ServerItem> serverList
public boolean stopModule = false;
public boolean pause = false;
public int mypriority;
public int old_coordinator_priority;
public int coordinator_priority;
public int messageServerPort=0;

public FaultDetectionM(int mypriority){
    this.mypriority = mypriority;
}

public boolean checkServer(String ip,
boolean ret = true;

    Socket socket;
    try {
        socket = new Socket(ip, port);
        socket.close();
    } catch (Exception e) {
        // TODO Auto-generated catch bl
        // e.printStackTrace();
        return false;
    }

    return ret;
}

public void addHealthCheckServerItem(fi
Configure project(s) from blueprint
Configure
Deactivate Checkstyle

Check Code with Checkstyle
Clear Checkstyle violations
Create Formatter-Profile
  
```

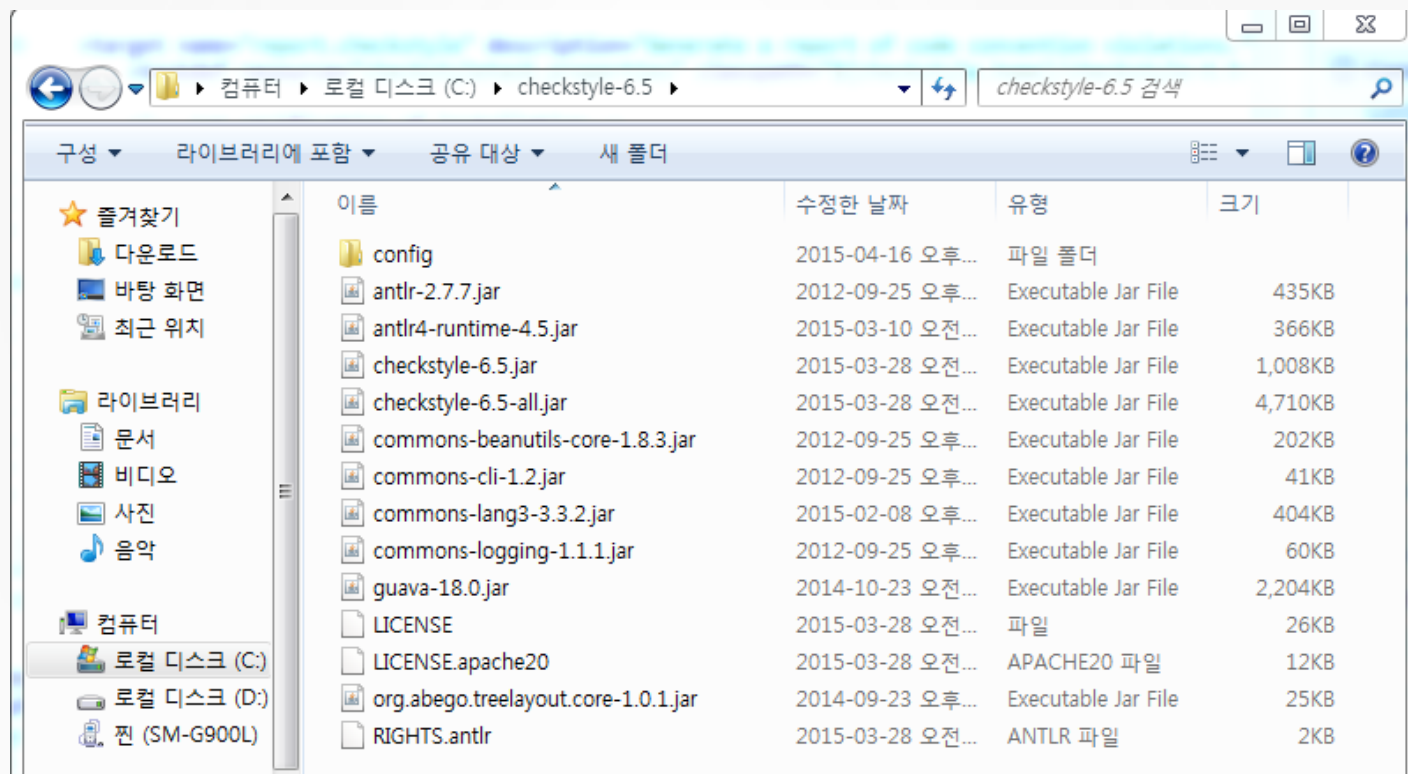
PART 2 Static analysis Tools

Ant : Checkstyle setting

1. Checkstyle 다운

<http://sourceforge.net/projects/checkstyle/files/checkstyle/6.5/checkstyle-6.5-bin.zip/download>

→ Checkstyle 6.5 버전을 다운 받아서 압축 해제



PART 2 Static analysis Tools

Ant : Checkstyle setting

2. build.xml 파일 설정

1) checkstyle 설치 경로를 checkstyle.home으로 설정

```
<property name="checkstyle.home" value="C:/checkstyle-6.5" />
```

2) Target 작성을 위해 참조할 classpath를, 다운 받은 checkstyle 내의 .jar 파일로 지정

```
<target name="report.checkstyle" description="Generate a report of code convention violations.">
  <taskdef resource="checkstyletask.properties" classpath="{checkstyle.home}/checkstyle-6.5-all.jar" />
```

3) 실행 시킬 파일(checkstyle-6.5-all.jar)을 available file로 지정

```
<!-- run verification of installation-->
<available file="{checkstyle.home}/checkstyle-6.5-all.jar" property="checkstyle.available" />
<fail unless="checkstyle.available" message="Error: CHECKSTYLE_HOME not set or checkstyle-all-6.5.jar not found." />
```

4) checkstyle 태그를 이용해서 checkstyle을 run 하는 부분 작성

config : checkstyle_checks.xml(이클립스에서 지정한 sun_checks.xml과 같은 것)

formatter : 결과 저장 형태 및 저장 경로와 이름 / fileset : 분석할 파일 지정

```
<!-- run analysis-->
<checkstyle config="{checkstyle.home}/config/checkstyle_checks.xml" failureProperty="checkstyle.failure" failOnViolation="false">
  <formatter type="xml" tofile="{report.checkstyle.dir}/checkstyle_report.xml" />
  <fileset dir="src" includes="**/*.java" />
</checkstyle>
</target>
```

PART 2 Static analysis Tools

Ant : Checkstyle setting

3. build 시 에러 해결

BUILD FAILED

C:\Users\홍진아\workspace\Repository1\hnhWeapons\build.xml:102: Unable to create a Checker: unable to read C:\checkstyle-6.5\config\checkstyle checks.xml

Total time: 11 seconds

unable to parse configuration stream - Property \${checkstyle.suppressions.file} has not been set

Checkstyle_check.xml에서 오류 : `${checkstyle.suppressions.file}`이 지정되지 않음
그래서 checkstyle_check.xml 내부 확인

```
<module name="SuppressionFilter">
  <property name="file" value="${checkstyle.suppressions.file}"/>
  <!--property name="file" value="C:/checkstyle-6.5/config/suppressions.xml"/-->
</module>
```

`${checkstyle.suppressions.file}`이 xml 파일 내부에 정의되어 있지 않는데,
suppressionFilter 부분에서 사용하고 있다는 것 확인

해결 → 실제 checkstyle 폴더 내부에 있는 suppressions.xml 폴더의 절대 경로로 수정

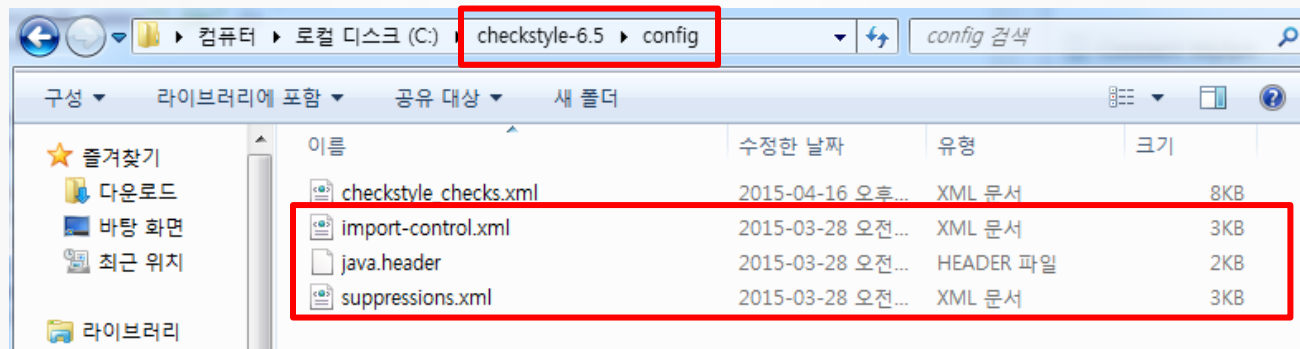
```
<module name="SuppressionFilter">
  <!--property name="file" value="${checkstyle.suppressions.file}"/-->
  <property name="file" value="C:/checkstyle-6.5/config/suppressions.xml"/>
</module>
```

PART 2 Static analysis Tools

Ant : Checkstyle setting

3. build 시 에러 해결

마찬가지 이유로, {Checkstyle설치경로}/config 내 3가지 파일에 대한 xml 부분을 모두 절대 경로로 수정



```
<module name="Header">
  <property name="headerFile" value="${checkstyle.header.file}"/>
  <!--property name="headerFile" value="C:/checkstyle-6.5/config/java.header"/-->
  <property name="fileExtensions" value="java"/>
  <property name="id" value="header"/>
</module>
```

```
<module name="ImportControl">
  <property name="file" value="${checkstyle.importcontrol.file}"/>
  <!--property name="file" value="C:/checkstyle-6.5/config/import-control.xml"/-->
</module>
<module name="IllegalImport"/>
<module name="IllegalInstantiation"/>
```


PART 2 Static analysis Tools

Ant : Checkstyle setting

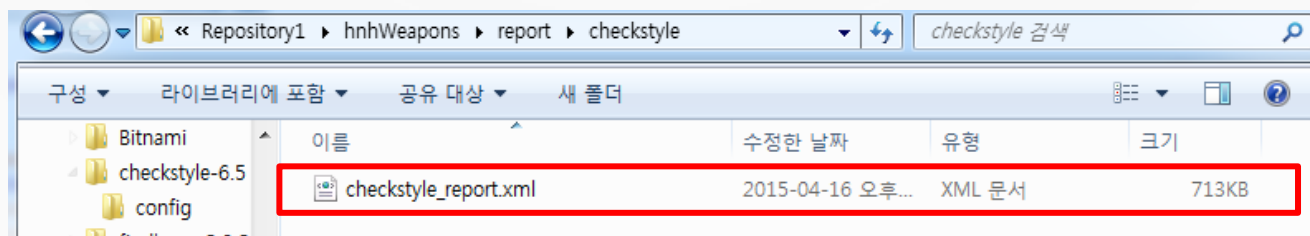
4. build 완료 및 checkstyle 수행

Checkstyle 수행 문구와 함께 build 성공

```

buildfile: C:\Users\홍진아\workspace\Repository1\hnhWeapons\build.xml
clean:
  [delete] Deleting directory C:\Users\홍진아\workspace\Repository1\hnhWeapons\report\checkstyle
  [delete] Deleting directory C:\Users\홍진아\workspace\Repository1\hnhWeapons\bin
init:
  [mkdir] Created dir: C:\Users\홍진아\workspace\Repository1\hnhWeapons\bin
build-subprojects:
build-project:
  [echo] hnhWeapons: C:\Users\홍진아\workspace\Repository1\hnhWeapons\build.xml
  [javac] Compiling 86 source files to C:\Users\홍진아\workspace\Repository1\hnhWeapons\bin
  [javac] warning: [options] bootstrap class path not set in conjunction with -source 1.7
  [javac] Note: C:\Users\홍진아\workspace\Repository1\hnhWeapons\src\gui\StoreListPanel.java uses unchecked or unsafe operations.
  [javac] Note: Recompile with -Xlint:unchecked for details.
  [javac] 1 warning
build:
report.checkstyle:
  [mkdir] Created dir: C:\Users\홍진아\workspace\Repository1\hnhWeapons\report\checkstyle
  [checkstyle] Running Checkstyle 6.5 on 86 files
run:
BUILD SUCCESSFUL
Total time: 13 seconds
  
```

지정한 경로에 지정한 이름으로 결과 xml 파일 저장된 것 확인 가능.



PART 2 Static analysis Tools

Ant : Checkstyle setting

4. build 완료 및 checkstyle 수행

결과 xml 내부

```

<?xml version="1.0" encoding="UTF-8"?>
- <checkstyle version="6.5">
  - <file name="C:\Users\함진아\workspace\Repository1\hnhWeapons\src\factories\BodyDefenseFactory.java">
    <error source="com.puppycrawl.tools.checkstyle.checks.javadoc.JavadocPackageCheck" message="Missing package-info.java file." severity="error" line="0"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.header.HeaderCheck" message="Line does not match expected header line of
////////////////////////////////////". severity="error" line="1"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.imports.ImportControlCheck" message="Import control file does not handle this package."
severity="error" line="1" column="9"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.javadoc.JavadocTypeCheck" message="Missing a Javadoc comment." severity="error" line="7"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.LeftCurlyCheck" message="{ should be on a new line." severity="error" line="7" column="56"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.whitespace.FileTabCharacterCheck" message="File contains tab characters (this is the first
instance)." severity="error" line="9" column="1"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.javadoc.JavadocMethodCheck" message="Missing a Javadoc comment." severity="error" line="9"
column="5"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.whitespace.WhitespaceAroundCheck" message="{ is not preceded with whitespace." severity="error"
line="9" column="32"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.LeftCurlyCheck" message="{ should be on a new line." severity="error" line="9" column="32"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.regexp.RegexpSinglelineCheck" message="Line matches the illegal pattern '\s+$'." severity="error"
line="10"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.LeftCurlyCheck" message="{ should be on a new line." severity="error" line="13"
column="73"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.coding.EqualsAvoidNullCheck" message="String literal expressions should be on the left side of an
equals comparison." severity="error" line="15" column="24"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.NeedBracesCheck" message="'else' construct must use '{}''s." severity="error" line="17"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.RightCurlyCheck" message="}' should be alone on a line." severity="error" line="17"
column="9"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.whitespace.WhitespaceAfterCheck" message=",' is not followed by whitespace." severity="error"
line="21" column="47"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.LeftCurlyCheck" message="{ should be on a new line." severity="error" line="21"
column="83"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.coding.EqualsAvoidNullCheck" message="String literal expressions should be on the left side of an
equals comparison." severity="error" line="23" column="24"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.NeedBracesCheck" message="'else' construct must use '{}''s." severity="error" line="25"/>
    <error source="com.puppycrawl.tools.checkstyle.checks.blocks.RightCurlyCheck" message="}' should be alone on a line." severity="error" line="25"
column="9"/>
  
```

PART 2 Static analysis Tools

Jenkins : Checkstyle plugin

1. checkstyle plugin 설치 및 build 설정

jenkins 설정 → 플러그인 관리 → 설치 가능 → checkstyle plugin & Analysis Collector Plugin 체크 및 설치 (Static Code Analysis Plug-ins가 자동 설치 된다.)

<input checked="" type="checkbox"/>	Checkstyle Plugin This plugin generates the trend report for Checkstyle , an open source static code analysis program.	3.42
<input type="checkbox"/>	Analysis Collector Plugin This plug-in is an add-on for the plug-ins Checkstyle , Dry , FindBugs , PMD , Task Scanner , and Warnings : the plug-in collects the different analysis results and shows the results in a combined trend graph. Additionally, the plug-in provides health reporting and build stability based on these combined results	1.43

Jenkins job 구성 → Build → Invoke Ant의 Target

ant 설정에서 checkstyle 수행을 위해 build.xml에 추가한 target을 입력

Targets

```
clean
init
build
report.findbugs
report.checkstyle
```

빌드 후 조치 → Publish checkstyle analysis results → 추출된 checkstyle 결과 xml

Publish Checkstyle analysis results

Checkstyle results

hnhWeapons/*/checkstyle/checkstyle_report.xml

PART 2 Static analysis Tools

Jenkins : Checkstyle plugin

2. 빌드 후 결과 확인

```
report.checkstyle:
[checkstyle] Running Checkstyle 6.5 on 86 files
```

◀ Ant에서의 Checkstyle 수행 메시지

```
BUILD SUCCESSFUL
```

```
Total time: 34 seconds
```

```
[CHECKSTYLE] Collecting checkstyle analysis files...
[CHECKSTYLE] Finding all files that match the pattern hnhWeapons/*/checkstyle/checkstyle_report.xml
[CHECKSTYLE] Parsing 1 file in C:\Users\함진아\.jenkins\jobs\hnhWeapons\workspace
[CHECKSTYLE] Successfully parsed file C:\Users\함진아
\.jenkins\jobs\hnhWeapons\workspace\hnhWeapons\report\checkstyle\checkstyle_report.xml with 4000 unique warnings
and 0 duplicates.
[CHECKSTYLE] Computing warning deltas based on reference build #37
```

▲ Checkstyle plugin을 통해 결과 xml을 가져와서 dashboard로 뿌려주기 위한 작업

 빌드 #38 (2015. 4. 16 오후 8:46:10)



Revision: 35
No changes.



익명 사용자에게 의해 시작됨



Checkstyle: [4,000 warnings](#) from one analysis.

• [4,000 new warnings](#)

◀ 빌드 후 결과 화면에서
checkstyle analysis 결과 추가

PART 2 Static analysis Tools

Jenkins : Checkstyle plugin

3. jenkins dashboard에서 checkstyle의 결과 확인

프로젝트의 왼쪽 메뉴에 Checkstyle Warning가 추가 되고, 클릭하면 dashboard 형태로 결과 확인

The screenshot displays the Jenkins dashboard for a project. On the left sidebar, the 'Checkstyle Warnings' menu item is highlighted with a red box. The main content area is titled 'CheckStyle Result' and contains the following sections:

Warnings Trend

All Warnings	New Warnings	Fixed Warnings
4000	4000	0

Summary

Total	High Priority	Normal Priority	Low Priority
4000	4000	0	0

Details

Navigation: Packages | Files | Categories | Types | Warnings | Details | New

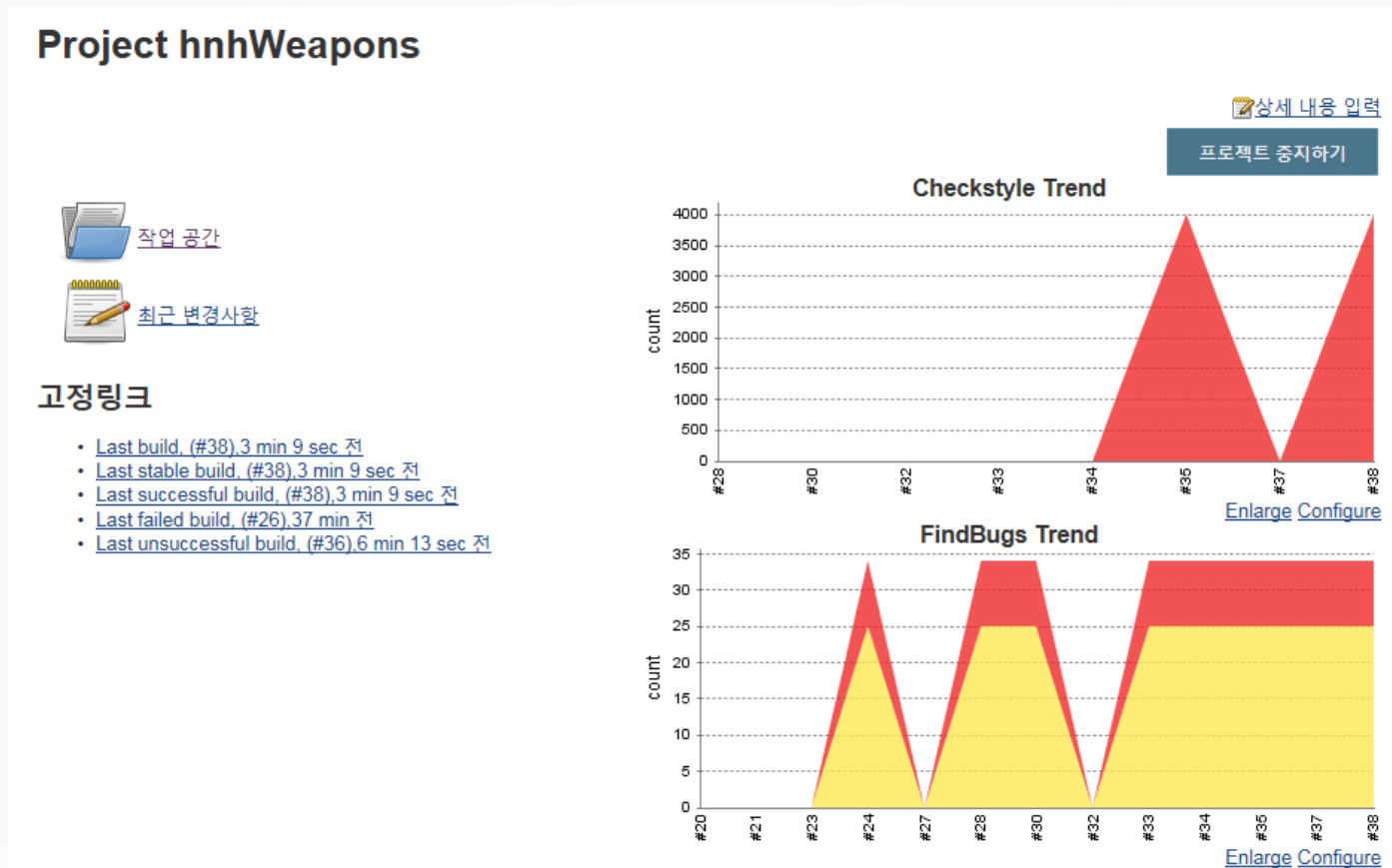
Package	Total	Distribution
factories	392	
gui	1846	
items	129	
items.defenses	140	
items.weapons	705	
player	455	
stores	333	
Total	4000	

PART 2 Static analysis Tools

Jenkins : Checkstyle plugin

3. jenkins dashboard에서 checkstyle의 결과 확인

프로젝트의 메인 화면에서 build # 에 따른 warning의 경향을 그래프로 확인할 수 있다.



PART 2 Static analysis Tools

What is PMD ?

- Program May Dependable의 약자
- 정해진 규칙에 따라 소스코드를 검사해서 위반사항을 찾아낼 수 있다.
 - 규칙 기반 소스코드 검사 (코드 최적화, 데드코드 등)
 - 중복코드 검색 (Cut & Paste Detector)
- Eclipse -plugin으로 사용할 수 있으며, 별도로 커맨드라인에서 사용 가능하다.
- 대상은 java, javaScript, XML 등이 있다.
- 라이선스는 BSD-style
- **주요 분석 목적 : 규칙 준수 & Cut & Paste 체크**

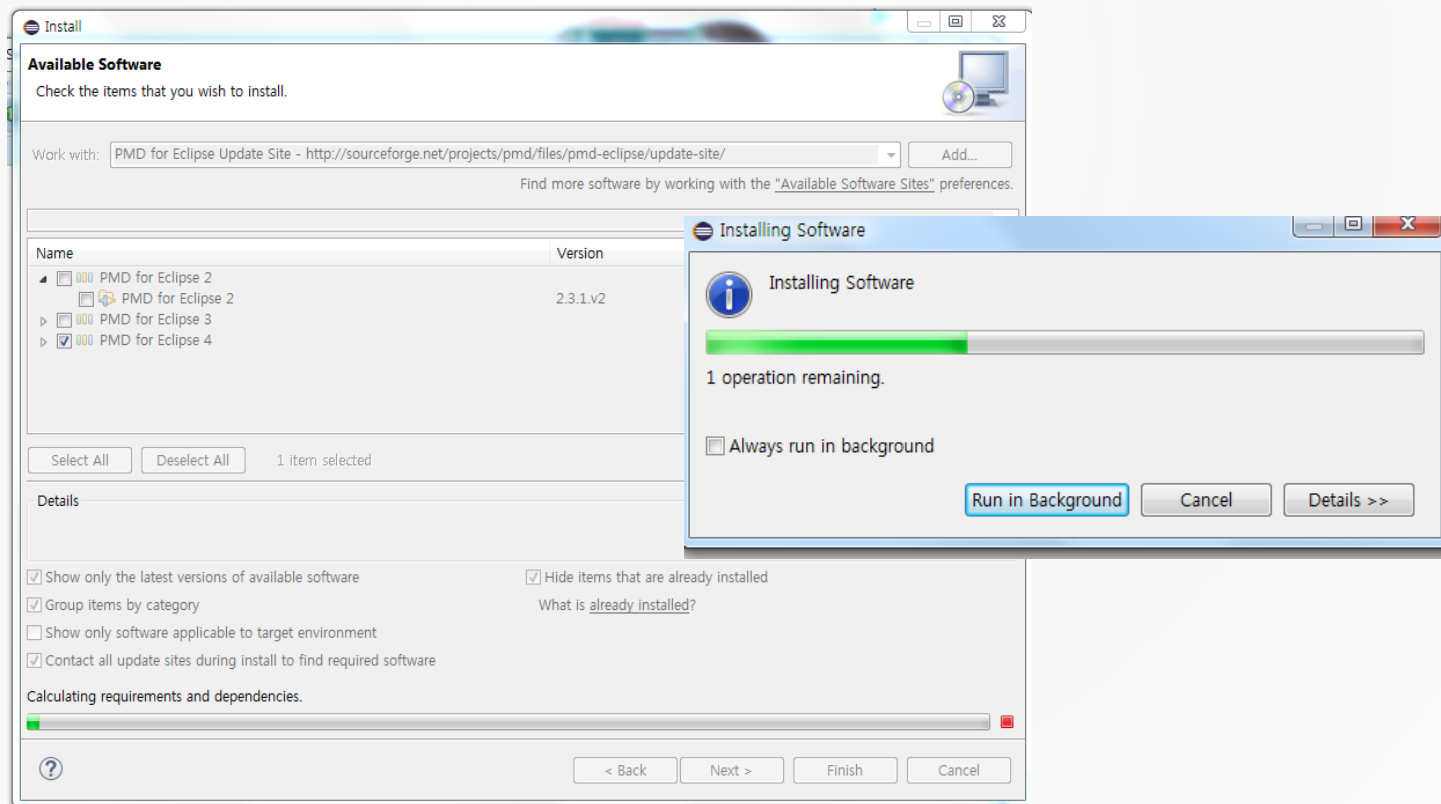
PART 2 Static analysis Tools

Eclipse : PMD plugin

1. PMD plugin 다운

<http://sourceforge.net/projects/pmd/files/pmd-eclipse/update-site/>

Help → Install new Software → PMD for Eclipse 다운로드

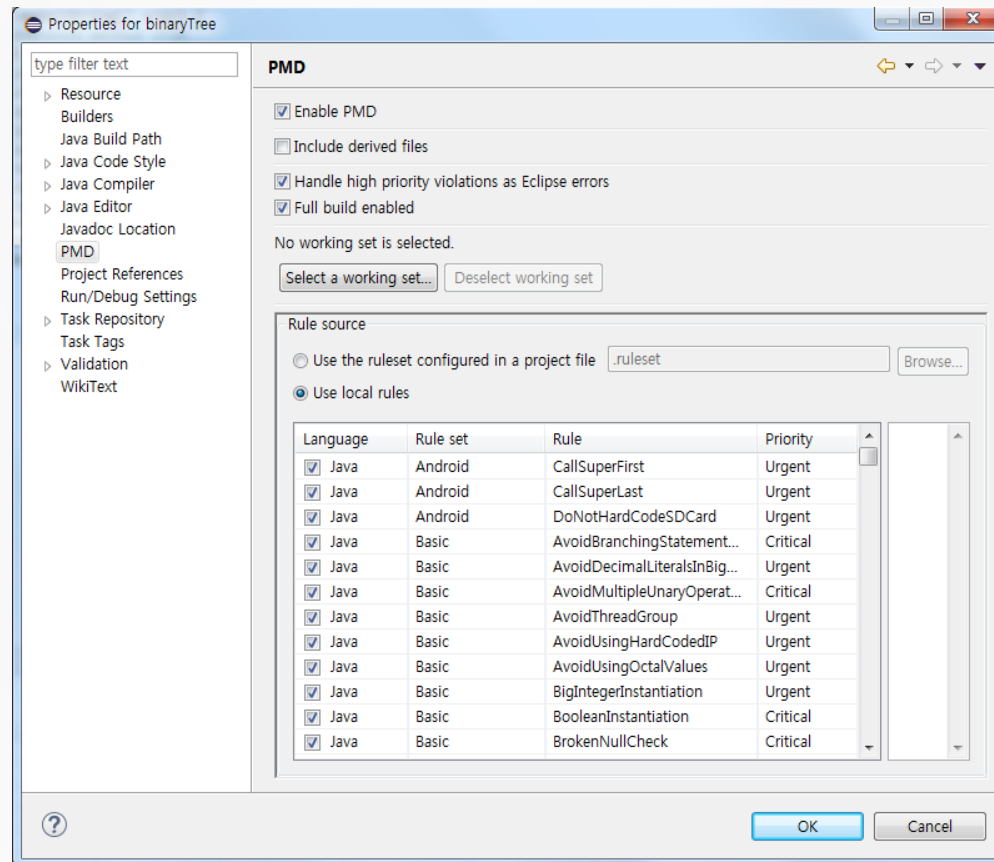


PART 2 Static analysis Tools

Eclipse : PMD plugin

2. PMD 환경설정

Project → properties → Rule source : PMD에서 검사할 rule을 정할 수 있다.

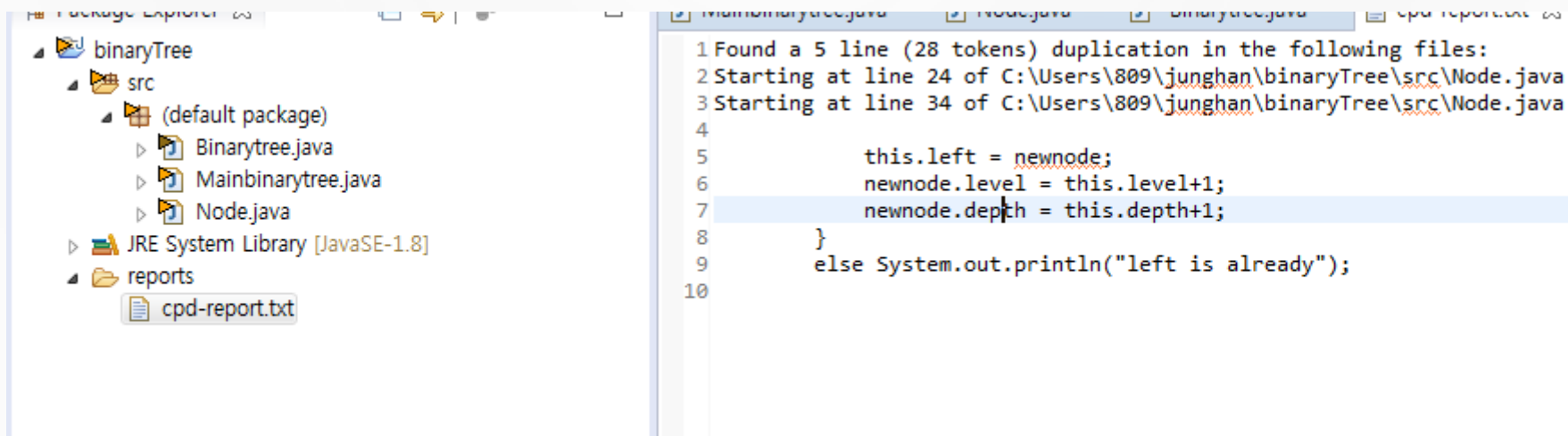
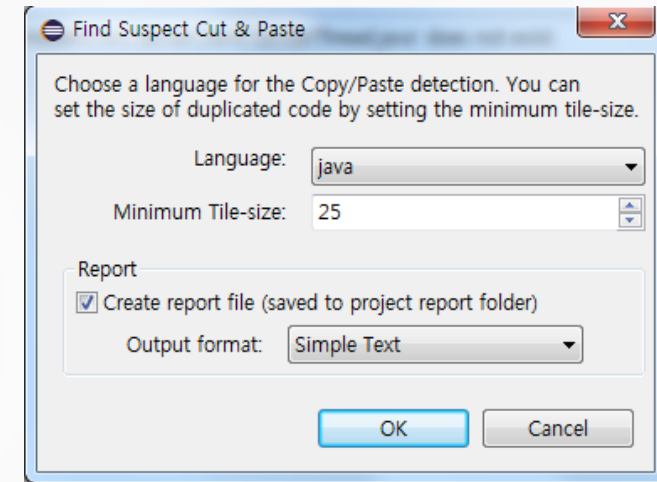


PART 2 Static analysis Tools

Eclipse : PMD plugin

3. PMD 중복검사 수행

프로젝트 우클릭 -> PMD -> Find Suspect Cut & Paste
CPD View 창에 결과가 보고된다.



PART 2 Static analysis Tools

Ant : PMD setting

1. PMD 다운로드

<http://pmd.sourceforge.net> → 5.3.0버전 다운로드 후 압축해제

The image shows a composite of two screenshots. On the left is the PMD website homepage, featuring the logo 'Pmd' with a gun barrel forming the letter 'P' and the slogan 'DON'T SHOOT THE MESSENGER'. Below the logo, it describes PMD as a source code analyzer. A 'Latest version' button is highlighted with a blue box. On the right is a Windows file explorer window showing the contents of the 'pmd-bin-5.3.0' directory, which includes subfolders for 'bin', 'lib', and 'rulesets', and a 'LICENSE' file.

Latest version

5.3.0 (1st April 2015)

- [Release Notes](#)
- [Download \(Sourcecode, Documentation\)](#)
- [Online Documentation](#)

이름	수정한 날짜	유형	크기
bin	2015-04-15 오후...	파일 폴더	
lib	2015-04-15 오후...	파일 폴더	
rulesets	2015-04-15 오후...	파일 폴더	
LICENSE	2015-04-01 오후...	파일	13KB

PART 2 Static analysis Tools

Ant : PMD setting

2. build.xml 파일 설정

```
<property name="pmd.home" value="C:/pmd-bin-5.3.0" />
```

1) pmd.home : pmd 설치 경로 설정

```
<property name="report.dir" location="${basedir}/report" />
```

```
<property name="report.pmd.dir" location="${report.dir}/pmd" />
```

2) 결과 폴더 지정

```
<path id="run.classpath">
  <pathelement path="${bin.dir}" />
</path>
```

3) Pmd 수행을 위한 Classpath 지정

```
<path id="pmd.classpath">
  <pathelement location="${build}" />
  <fileset dir="${pmd.home}/lib/">
    <include name="*.jar" />
  </fileset>
</path>
```

4) pmd 수행을 위한 target 지정

```
<taskdef name="pmd" classname="net.sourceforge.pmd.ant.PMDTask" classpathref="pmd.classpath" />
```

```
<target name="report.pmd">
```

```
  <mkdir dir="${report.pmd.dir}" />
```

```
  <pmd rulesetfiles="${pmd.home}/rulesets/ExcepGen.xml">
```

```
    <formatter type="xml" tofile="${report.pmd.dir}/pmd_report.xml" />
```

```
    <fileset dir="${src.dir}">
```

```
      <include name="**/*.java" />
```

```
    </fileset>
```

```
  </pmd>
```

```
</target>
```

5) 실제 pmd 수행 태그 <pmd>
rulesetfiles : pmd 수행을 위한
ruleset이 정의된 xml 파일이 필요!

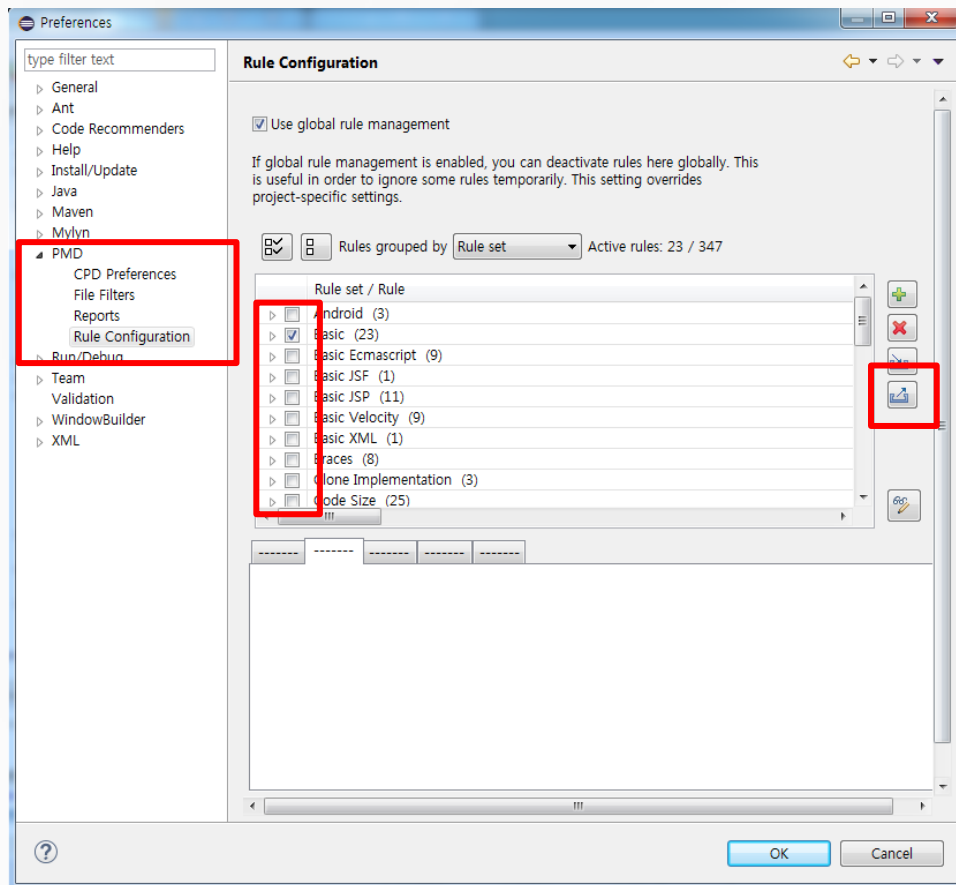
formatter : 결과 파일 경로 및 이름

fileset : 분석할 파일 경로 지정

PART 2 Static analysis Tools

Ant : PMD setting

3. Eclipse를 이용하여 PMD 분석을 위한 Ruleset이 정의된 xml 파일 추출



Eclipse → Windows → Preferences
→ PMD → Rule Configuration

원하는 Rule을 선택하여 Apply 한 뒤
오른쪽 Export 버튼 클릭 후
{Pmd 설치경로}/rulesets 폴더를 만들
어서 그 안에 저장한다.

여기서 추출 된 ruleset xml 파일을
build.xml에서 사용한다 !

PART 2 Static analysis Tools

Ant : PMD setting

3. Eclipse를 이용하여 PMD 분석을 위한 Ruleset이 정의된 xml 파일 추출

```
<?xml version="1.0" encoding="UTF-8"?>
<ruleset xmlns="http://pmd.sourceforge.net/ruleset/2.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  name="ruleset"
  xsi:schemaLocation="http://pmd.sourceforge.net/ruleset/2.0.0 http://pmd.sourceforge.net/ruleset/2.0.0" >
  <description>PMD Plugin preferences rule set</description>
  <rule ref="rulesets/java/basic.xml/AvoidBranchingStatementAsLastInLoop">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/AvoidDecimalLiteralsInBigDecimalConstructor">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/AvoidMultipleUnaryOperators">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/AvoidThreadGroup">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/AvoidUsingHardCodedIP">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/AvoidUsingOctalValues">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/BigIntegerInstantiation">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/BooleanInstantiation">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/BrokenNullCheck">
    <priority>4</priority>
  </rule>
  <rule ref="rulesets/java/basic.xml/CheckResultSet">
    <priority>4</priority>
  </rule>
</ruleset>
```

추출된 xml 파일에서
Eclipse에서 선택한 rule들이 추가되어
있는 모습을 확인할 수 있다.

PART 2 Static analysis Tools

Jenkins : PMD plugin

1. PMD plugin 설치 및 build 설정

jenkins 설정 → 플러그인 관리 → 설치 가능 → PMD Plugin 체크 및 설치



PMD Plugin

This plugin generates the trend report for [PMD](#), an open source static code analysis program.

3.41

Jenkins job 구성 → Build → Invoke Ant의 Target

ant 설정에서 PMD 수행을 위해 build.xml에 추가한 target을 입력

Build

Invoke Ant ?

Ant Version ▼

Targets ?

빌드 후 조치 → Publish PMD analysis results → 추출된 PMD 결과 xml

Publish PMD analysis results ?

PMD results

[Fileset includes](#) setting that specifies the generated raw PMD XML report files, such as `**/pmd.xml`. Basedir of the fileset is [the workspace root](#). If no value is set, then the default `**/pmd.xml` is used. Be sure not to include any non-report files into this pattern.

PART 2 Static analysis Tools

Jenkins : PMD plugin

2. 빌드 후 결과 확인

◀ Ant에서의 PMD 수행 메시지

```
report.pmd:
```

```
Removed misconfigured rule: LoosePackageCoupling cause: No packages or classes specified
```

```
[pmd] 4월 13, 2013 8:43:32 오후 net.sourceforge.pmd.PMD removeBrokenRules
```

```
[pmd] 경고: Removed misconfigured rule: LoosePackageCoupling cause: No packages or classes specified
```

```
BUILD SUCCESSFUL
```

```
Total time: 5 seconds
```

```
[PMD] Collecting PMD analysis files...
```

```
[PMD] Finding all files that match the pattern **/report/pmd/pmd_report.xml
```

```
[PMD] Parsing 1 file in C:\Users\함진아\.jenkins\jobs\Project1\workspace
```

```
[PMD] Successfully parsed file C:\Users\함진아\.jenkins\jobs\Project1\workspace\Project1\report\pmd\pmd_report.xml  
with 6 unique warnings and 0 duplicates.
```

```
Sending e-mails to: jinaham11@gmail.com
```

```
Finished: SUCCESS
```

▲ PMD plugin을 통해 결과 xml을 가져와서 dashboard로 뿌려주기 위한 작업

PART 2 Static analysis Tools

Jenkins : PMD plugin

3. jenkins dashboard에서 PMD의 결과 확인

프로젝트의 왼쪽 메뉴에 PMD Warnings가 추가 되고, 클릭하면 dashboard 형태로 결과 확인 checkstyle과 마찬가지로, 프로젝트 메인 화면에서 그래프 형태로 build #에 따른 PMD warnings의 경향을 살펴볼 수 있다.

- [프로젝트로 돌아가기](#)
- [상태](#)
- [바뀐점](#)
- [Console Output](#)
- [빌드 정보 수정](#)
- [이 빌드를 삭제](#)
- [Tag this build](#)
- [PMD Warnings](#)
- [이전 빌드](#)

PMD Result

Warnings Trend

All Warnings	New Warnings	Fixed Warnings
6	6	0

Summary

Total	High Priority	Normal Priority	Low Priority
6	2	4	0

Details

Categories	Types	Warnings	Details	New	High	Normal
Category		Total	Distribution			
Design		2	<div style="width: 100%; height: 10px; background-color: yellow;"></div>			
Java Logging		2	<div style="width: 100%; height: 10px; background-color: red;"></div>			
Naming		1	<div style="width: 100%; height: 10px; background-color: yellow;"></div>			
Optimization		1	<div style="width: 100%; height: 10px; background-color: yellow;"></div>			
Total		6				

PART 2 Static analysis Tools

What is JDepend ?

- 패키지 의존성과 관련된 설계 품질을 관리/분석 할 수 있다.
- 클래스 파일들을 읽어서 분석을 한다.
- 각 패키지별로 의존성 측정이 가능하며, 수치화, 그래프로 표현이 가능하다.
- 개발과정 중에도 아키텍처 품질을 평가할 수 있다.
- XML 형식으로도 저장 가능하다.

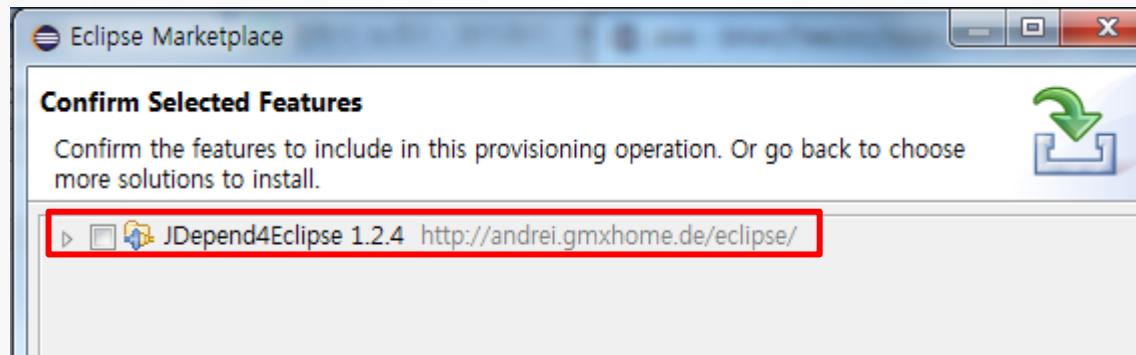
- **주요 분석 목적 : 아키텍처 품질 검사 (의존성 분석)**

PART 2 Static analysis Tools

Eclipse : JDepend plugin

1. JDepend Plugin 다운로드

Help → Install new Software → <http://andrei.gmxhome.de/eclipse/>



PART 2 Static analysis Tools

Eclipse : JDepend plugin

2. JDepend 실행 및 결과

프로젝트 우클릭 → run jdepend analysis

The screenshot displays the JDepend Eclipse plugin interface. The left pane shows a package tree with several packages selected, including those from 'kr.co.springframework' and 'net.openseed.core'. The right pane shows the analysis results for the selected packages.

Selected object(s)

Package	CC(...)	AC(...)	Ca(a...)	Ce(e...)	A	I	D	Cycle
kr.co.springframework.command	1	0	3	1	0.00	0.25	0.75	

Packages with cycle

Package	CC(...)	AC(...)	Ca(a...)	Ce(e...)	A	I	D	Cycle
kr.co.springframework.member.dao	7	1	1	9	0.12	0.90	0.02	⚠
kr.co.springframework.member.service	3	1	2	9	0.25	0.81	0.06	⚠
kr.co.springframework.member.sorting	1	0	1	3	0.00	0.75	0.25	⚠
kr.co.springframework.model	2	0	5	5	0.00	0.50	0.50	⚠

Depends upon - efferent dependencies

Package	CC(...)	AC(...)	Ca(a...)	Ce(e...)	A	I	D	Cycle
kr.co.springframework.command	1	0	3	1	0.00	0.25	0.75	
kr.co.springframework.exception	3	0	2	1	0.00	0.33	0.66	
kr.co.springframework.member.dao	7	1	1	9	0.12	0.90	0.02	⚠
kr.co.springframework.member.serv...	3	1	2	9	0.25	0.81	0.06	⚠
kr.co.springframework.member.sort...	1	0	1	3	0.00	0.75	0.25	⚠
kr.co.springframework.model	2	0	5	5	0.00	0.50	0.50	⚠
kr.co.springframework.model.enum...	1	0	4	2	0.00	0.33	0.66	

Used by - afferent dependencies

Package	CC(...)	AC(...)	Ca(a...)	Ce(e...)	A	I	D	Cycle
kr.co.springframework.controller	10	0	0	17	0.00	1.00	0.00	⚠
kr.co.springframework.interceptor	1	0	0	5	0.00	1.00	0.00	
kr.co.springframework.member.dao	7	1	1	9	0.12	0.90	0.02	⚠
kr.co.springframework.member.service	3	1	2	9	0.25	0.81	0.06	⚠
kr.co.springframework.member.sorting	1	0	1	3	0.00	0.75	0.25	⚠
kr.co.springframework.model	2	0	5	5	0.00	0.50	0.50	⚠
kr.co.springframework.model.enumeration	1	0	4	2	0.00	0.33	0.66	
kr.co.springframework.validator	3	0	0	5	0.00	1.00	0.00	⚠
net.openseed.core.test	2	3	3	18	0.60	0.85	0.45	

Metrics

Instability -> (graph showing a downward trend)

Abstractness (graph showing an upward trend)

14 elements selected

PART 2 Static analysis Tools

Eclipse : JDepend plugin

※ 결과가 가지는 의미

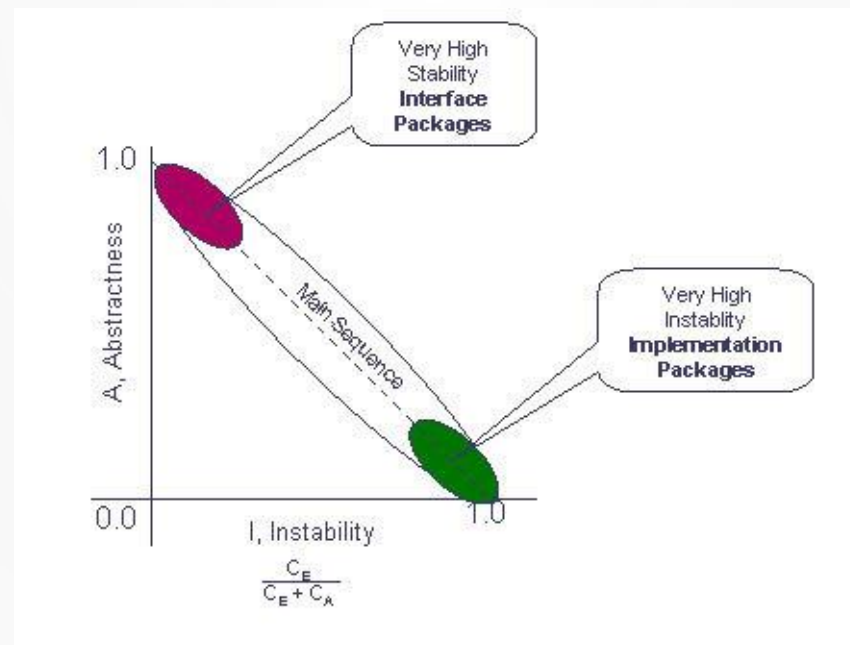
CC	Concrete Class
AC	Abstract Class 추상클래스나 인터페이스의 개수. 확정성의 척도
Ca	Afferent Couplings 현재 패키지에 종속성을 갖는 패키지 개수.
Ce	Efferent Couplings 현재 패키지가 종속하고 있는 패키지 개수.
A	추상화 정도. 0은 완전 구체적 패키지. 1은 추상적 패키지.
D	Main Sequence로 부터의 거리. Main Sequence란 추상적이면서 안정적인거나, 완전 구체적이면서 불안정한 패키지. 0일수록 가깝고 1일수록 멀다.

PART 2 Static analysis Tools

Eclipse : JDepend plugin

※ 결과가 가지는 의미

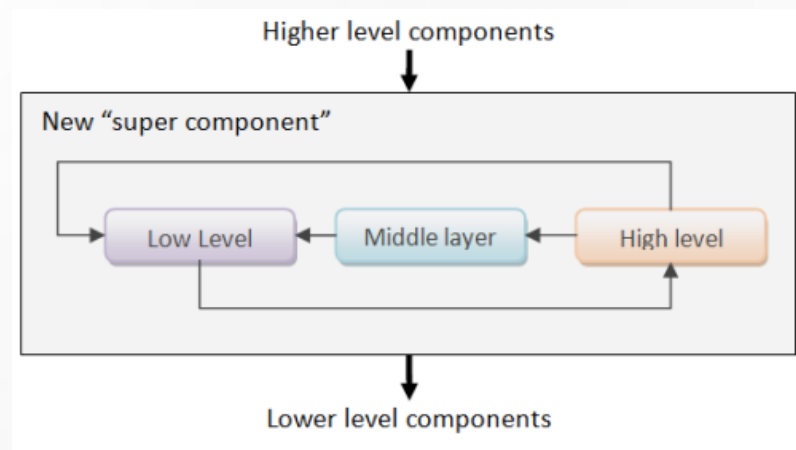
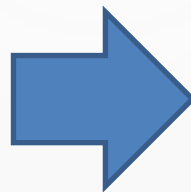
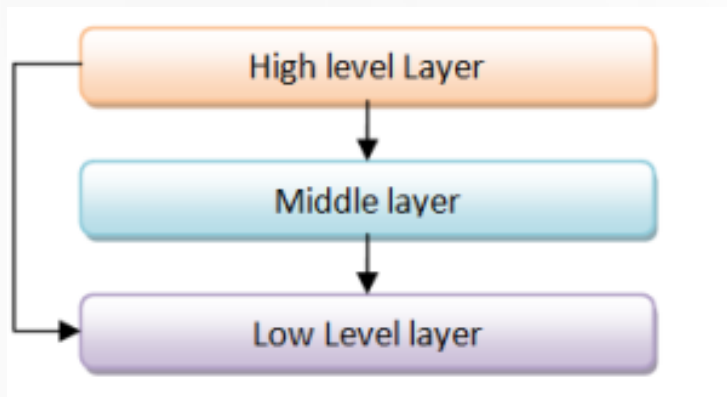
- D 값이 낮을 수록 인터페이스 기반의 개발. Coupling이 적음
- 1에 가까울수록 다른 클래스들이 의존 해야 될 클래스이고
- 0에 가까울수록 다른 클래스에 의존성을 가져야 한다.



PART 2 Static analysis Tools

Why Dependency ?

- 큰 규모의 개발을 쉽게 하기 위해서 덩어리 기반으로 쪼개어 개발을 하게 된다.
- 쪼개어 개발함으로써 개발과 유지보수에 도움이 된다.
- Dependency 구조가 꼬여서 cyclic 구조가 생기게 되면, 쪼개놓은 것이 하나로 합쳐지는 것이 된다.



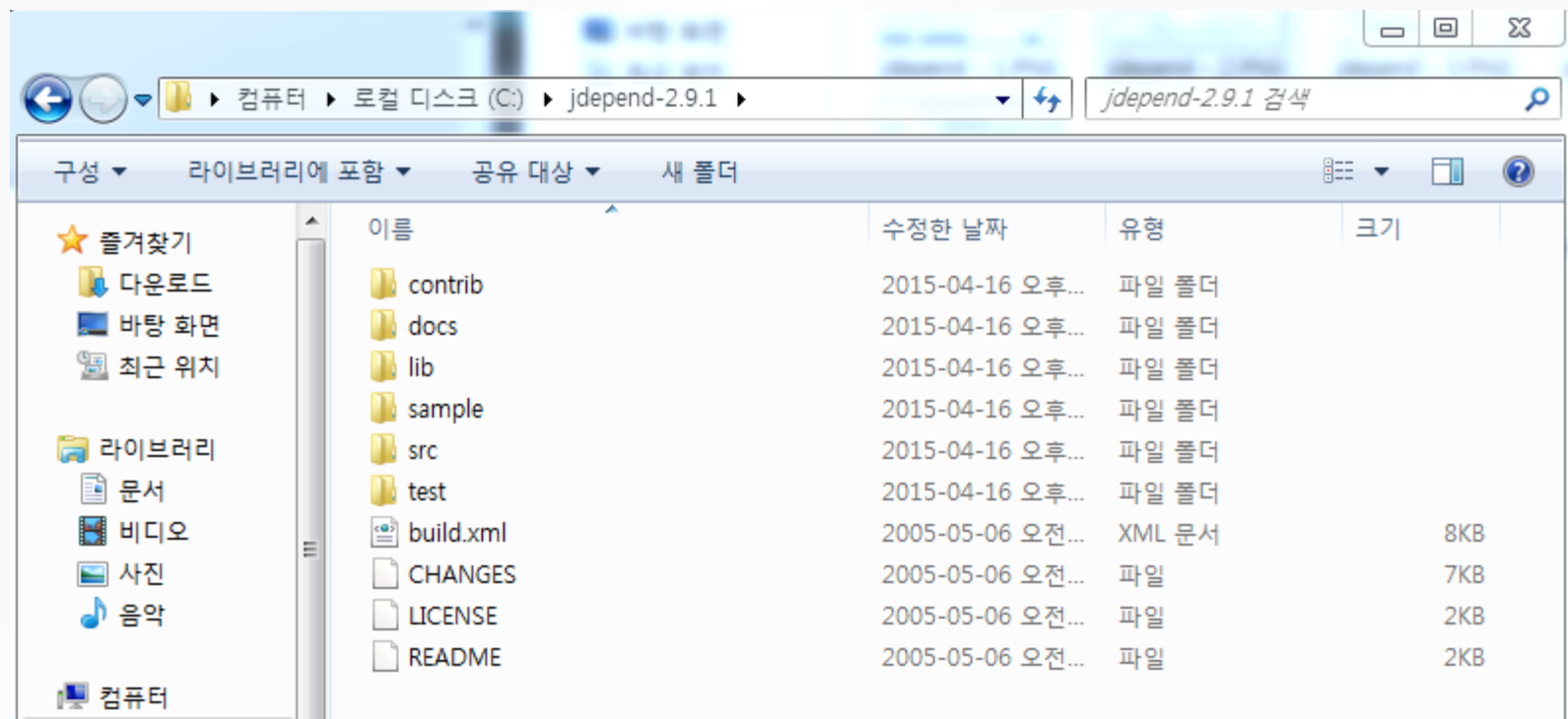
PART 2 Static analysis Tools

Ant : JDepend setting

1. JDepend 다운

<http://www.softpedia.com/dyn-postdownload.php/c2597537e5026e84c0c4180697e22b64/552fab3e/2de28/0/1>

→ JDepend 2.9.1 버전을 다운 받아서 압축 해제



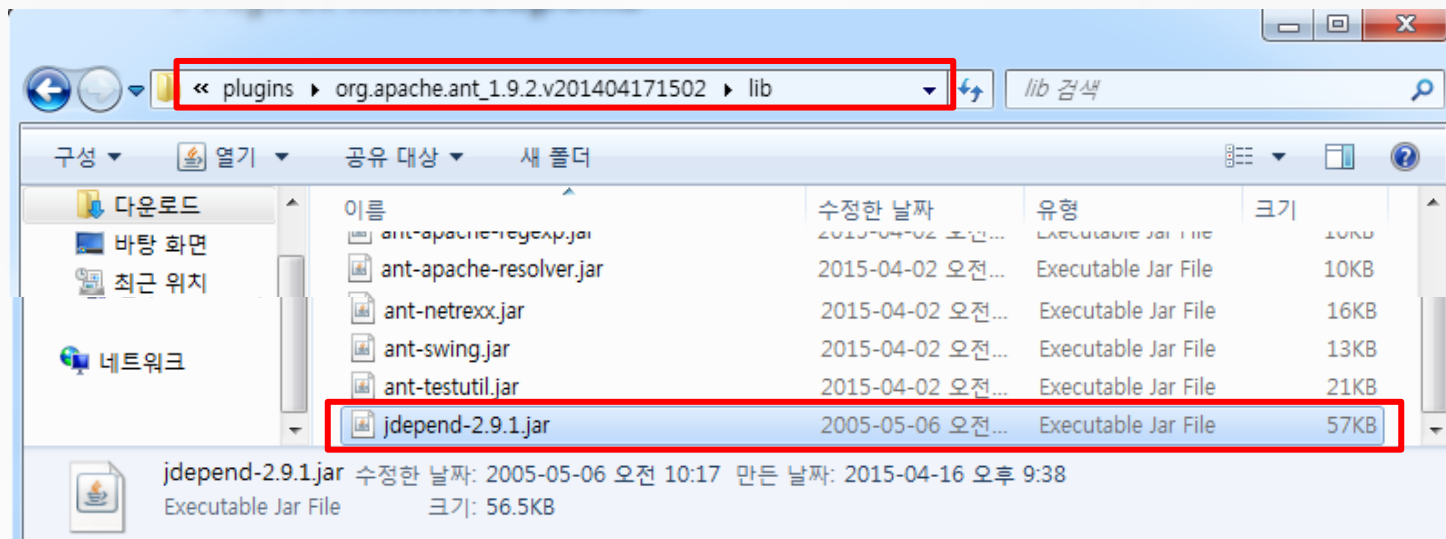
PART 2 Static analysis Tools

Ant : JDepend setting

2. Ant library에 JDepend 추가

Eclipse ant plugin 폴더 안에 다운 받은 jdepend-2.9.1.jar를 복사해 넣는다.

→ Jdepend의 경우, Classpath를 따로 설정하지 않아도 build.xml에서 jdepend 실행을 위한 jdepend target을 지정할 수 있다.(ant에서 제공) 그러나 ant는 태그에 대한 format만 정의해줄 뿐, 실제 jdepend 실행을 위한 .jar 파일은 제공하지 않고 있어서 직접적으로 ant의 library 폴더 안에 jdepend.jar을 복사해주어야 한다.



PART 2 Static analysis Tools

Ant : JDepend setting

3. build.xml 파일 설정

- 1) 실제 jdepend가 실행되는 <jdepend> 태그 지정
- 2) Format / outputfile : 결과 파일 xml 경로 및 이름 지정
- 3) Classpath location : 태그를 위한 참조 영역이 아닌 검사를 수행 할 파일들의 경로 !
- 4) 결과를 html 형태로 추출이 가능하다.

```
<target name="jdepend">|
  <jdepend format="xml" outputfile="${report.jdepend.dir}/jdepend-report.xml">
    <exclude name="java.*"/>
    <exclude name="javax.*"/>
    <classpath>
      <pathelement location="bin" />
    </classpath>
    <classpath location="bin" />
  </jdepend>

  <style basedir="${report.jdepend.dir}" destdir="${report.jdepend.dir}"
    includes="jdepend-report.xml"
    style="${ant.home}/etc/jdepend.xsl" />
</target>
```

PART 2 Static analysis Tools

Ant : JDepend setting

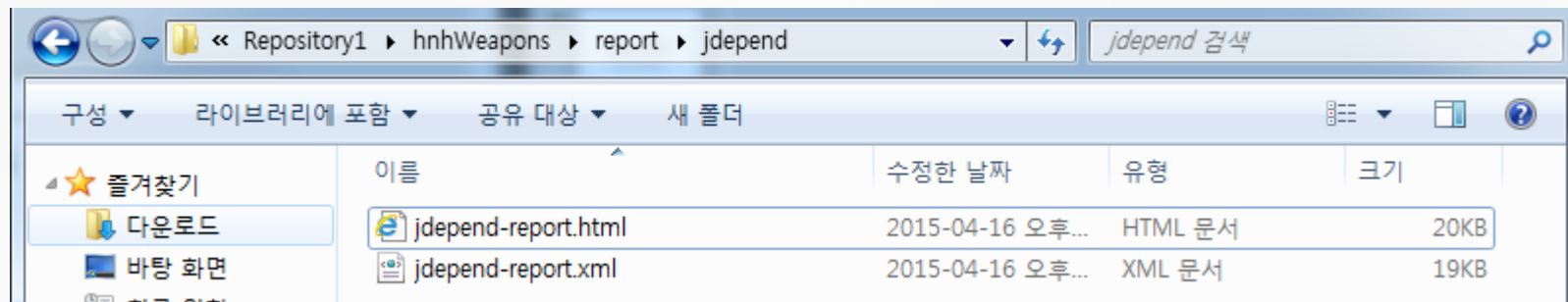
4. build 완료 및 jdepend 수행

```

jdepend:
[jdepend] Output to be stored in C:\Users\#진아\workspace\Repository1\hnhWeapons\report\jdepend\jdepend-report.xml
[style] Warning: the task name <style> is deprecated. Use <xslt> instead.
[style] Transforming into C:\Users\#진아\workspace\Repository1\hnhWeapons\report\jdepend
[style] Processing C:\Users\#진아\workspace\Repository1\hnhWeapons\report\jdepend\jdepend-report.xml to C:\Users\#진아\workspace\Repository1
[style] Loading stylesheet C:\Users\#진아\Downloads\eclipse-java-luna-SR2-win32-x86_64\eclipse\plugins\org.apache.ant_1.9.2.v201404171502\e
run:
BUILD SUCCESSFUL
Total time: 5 seconds

```

지정한 경로에 지정한 이름으로 결과 xml, html파일 저장된 것 확인 가능.



PART 2 Static analysis Tools

Ant : JDepend setting

4. build 완료 및 jdepend 수행

xml 파일로부터 추출된 결과 html 화면

JDepend Analysis

Designed for use with [JDepend](#) and [Ant](#).

Summary

[\[summary\]](#) [\[packages\]](#) [\[cycles\]](#) [\[explanations\]](#)

Package	Total Classes	Abstract Classes	Concrete Classes	Afferent Couplings	Efferent Couplings	Abstractness	Instability	Distance
factories	14	1	13	3	4	0.07	0.57	0.36
gui	28	1	27	0	6	0.04	1	0.04
items	4	2	2	6	0	0.5	0	0.5
items.defenses	8	4	4	4	1	0.5	0.2	0.3
items.weapons	25	9	16	4	2	0.36	0.33	0.31
player	4	1	3	4	5	0.25	0.56	0.19
stores	6	3	3	2	5	0.5	0.71	0.21

Packages

[\[summary\]](#) [\[packages\]](#) [\[cycles\]](#) [\[explanations\]](#)

factories

[Afferent Couplings](#): 3

[Efferent Couplings](#): 4

[Abstractness](#): 0.07

[Instability](#): 0.57

[Distance](#): 0.36

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
factories.ItemFactory	factories.BodyDefenseFactory factories.DefenseFactory factories.EnemyFactory factories.FactoryTypeException factories.FootDefenseFactory factories.GeneralWeaponFactory factories.HeadDefenseFactory factories.OneShotWeaponFactory factories.PlayerFactory factories.RFunc factories.UseUpWeaponFactory factories.WarheadFactory factories.WeaponFactory	gui player stores	items items.defenses items.weapons player

gui

[Afferent Couplings](#): 0

[Efferent Couplings](#): 6

[Abstractness](#): 0.04

[Instability](#): 1

[Distance](#): 0.04

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
gui.IconPics	gui.AskFightDialog gui.BattleFrame gui.BattleFrame\$1 gui.BattleFrame\$2 gui.BattleFrame\$2\$1 gui.ChangeDefenseDialog	None	factories items items.defenses items.weapons player stores

PART 2 Static analysis Tools

Jenkins : JDepend plugin

1. JDepend plugin 설치 및 build 설정

jenkins 설정 → 플러그인 관리 → 설치 가능 → JDepend plugin 체크 및 설치

JDepend Plugin 1.2.4
 The JDepend Plugin is a plugin to generate JDepend reports for builds.

Jenkins job 구성 → Build → Invoke Ant의 Target

ant 설정에서 JDepend 수행을 위해 build.xml에 추가한 target을 입력

Targets

- clean
- init
- build
- report.findbugs
- report.checkstyle
- jdepend**

빌드 후 조치 → Report JDepend → 추출된 JDepend 결과 xml

Report JDepend
 Pre-generated JDepend File












*Provide a path to a JDepend file created during the build.
 Use a preceding "/" to specify an absolute path, leave off the "/" to specify a path within the workspace.
 Leave blank to have the plugin generate its own file.*

PART 2 Static analysis Tools

Jenkins : JDepend plugin

2. jenkins dashboard에서 JDepend의 결과 확인

프로젝트의 왼쪽 메뉴에 JDepend가 추가 되고, 클릭하면 dashboard 형태로 결과 확인

-  프로젝트로 돌아가기
-  상태
-  바뀐점
-  Console Output
-  빌드 정보 수정
-  이 빌드를 삭제
-  Tag this build
-  Checkstyle Warnings
-  FindBugs Warnings
-  JDepend
-  이전 빌드

Metric Results

[\[summary \]](#) [\[packages \]](#) [\[cycles \]](#) [\[explanations \]](#)

The following document contains the results of a JDepend metric analysis. The various metrics are defined at the bottom of this document.

Summary

[\[summary \]](#) [\[packages \]](#) [\[cycles \]](#) [\[explanations \]](#)

Package	TC	CC	AC	Ca	Ce	A	I	D	V
factories	14	13	1	3	4	7.0%	57.0%	36.0%	1
gui	28	27	1	0	6	4.0%	100.0%	4.0%	1
items	4	2	2	6	0	50.0%	0.0%	50.0%	1
items.defenses	8	4	4	4	1	50.0%	20.0%	30.000002%	1
items.weapons	25	16	9	4	2	36.0%	33.0%	31.0%	1
player	4	3	1	4	5	25.0%	56.0%	19.0%	1
stores	6	3	3	2	5	50.0%	71.0%	21.0%	1

Packages

[\[summary \]](#) [\[packages \]](#) [\[cycles \]](#) [\[explanations \]](#)

factories

Afferent Couplings	Efferent Couplings	Abstractness	Instability	Distance
3	4	7.0%	57.0%	36.0%

Abstract Classes	Concrete Classes	Used by Packages	Uses Packages
factories.ItemFactory	factories.BodyDefenseFactory factories.DefenseFactory factories.EnemyFactory factories.FactoryTypeException factories.FootDefenseFactory factories.GeneralWeaponFactory factories.HeadDefenseFactory factories.OneShotWeaponFactory	gui player stores	items items.defenses items.weapons player

PART 2 Static analysis Tools

What is Findbugs ?

- 자바프로그램 분석도구로, 메릴랜드 대학에서 개발했다.
- data flow, control flow 분석등을 이용한다.
- 버그 패턴을 자동으로 찾아서 알려준다 (ex : race condition, null Pointer exception 등)
- source code(*.java 파일) 보다는 byte code(*.class 파일)를 기반으로 분석을 진행한다.

- **주요 분석 목적 : 버그 패턴 분석**

PART 2 Static analysis Tools

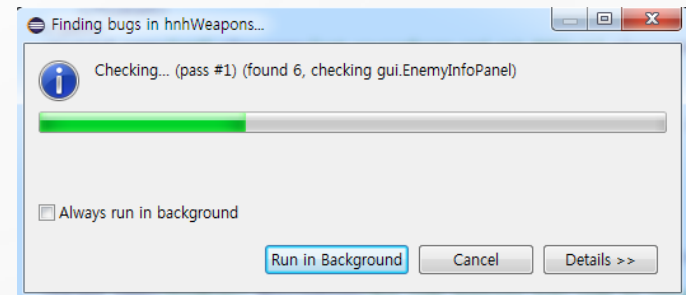
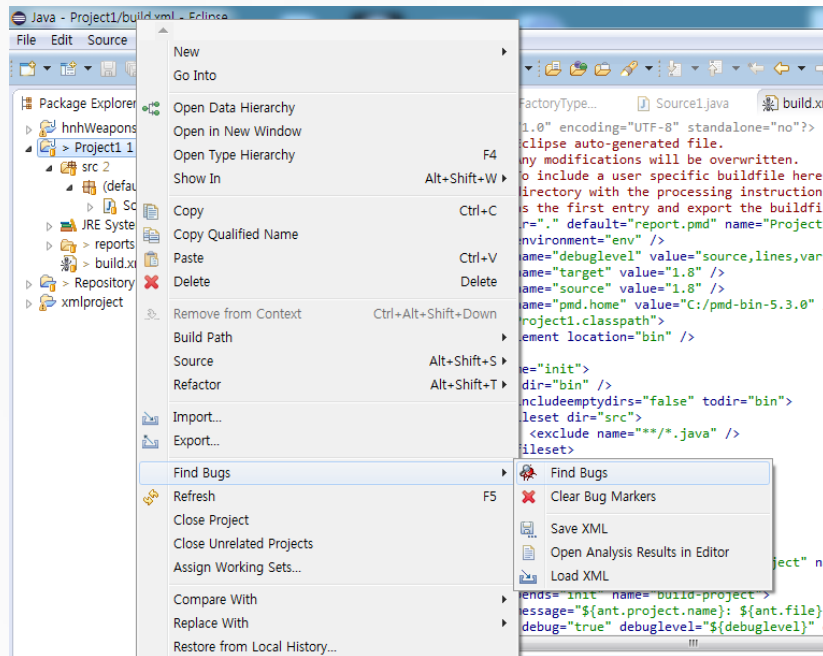
Eclipse : Findbugs plugin

1. Findbugs Plugin 다운

Help → Install new Software → <http://findbugs.cs.umd.edu/eclipse/>

2. Findbugs 실행

프로젝트 우클릭 → findbugs → findbugs

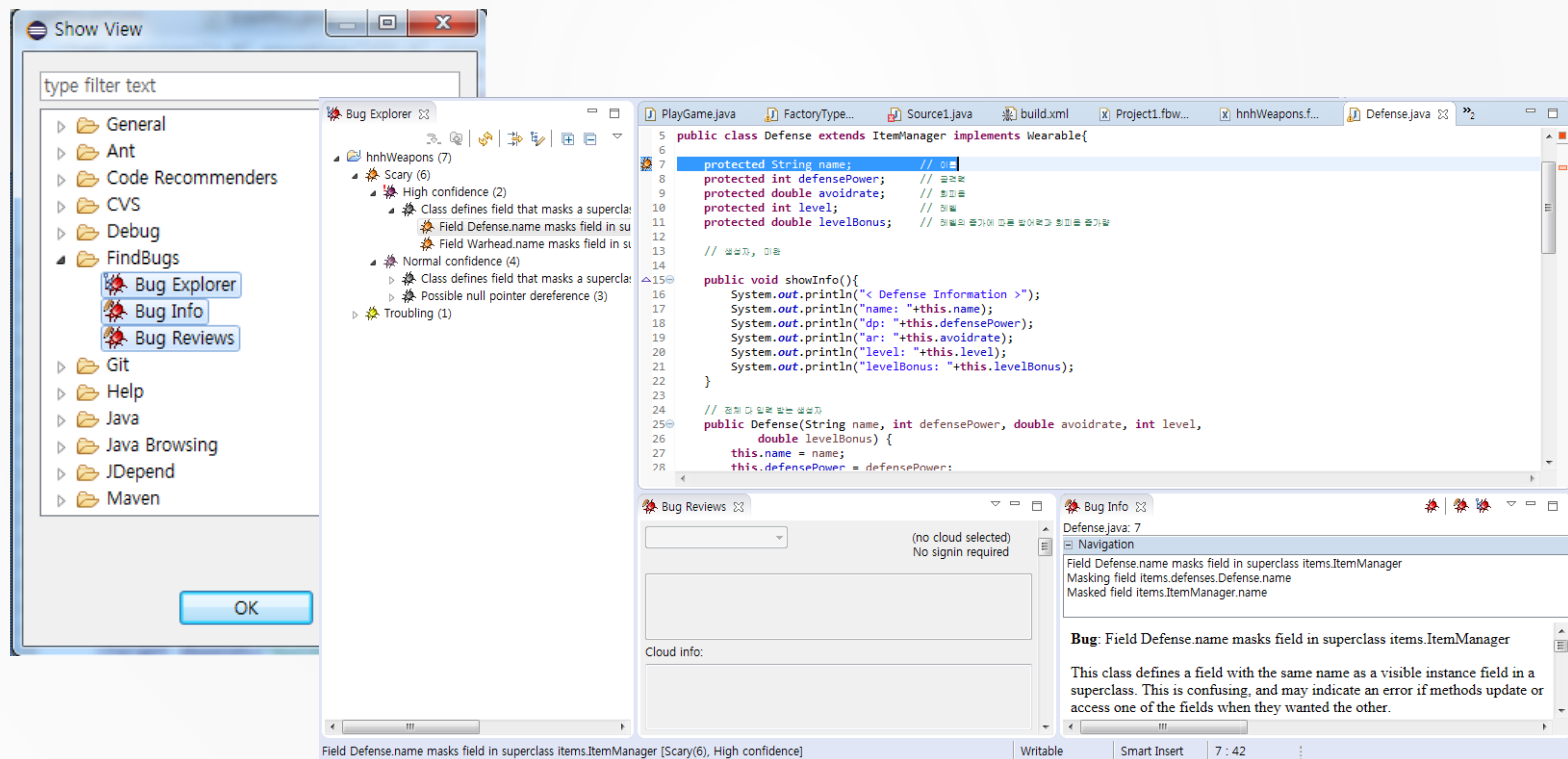


PART 2 Static analysis Tools

Eclipse : Findbugs plugin

3. Findbugs 결과 확인

Windows → Show view → Others → findbugs관련 결과 확인 가능



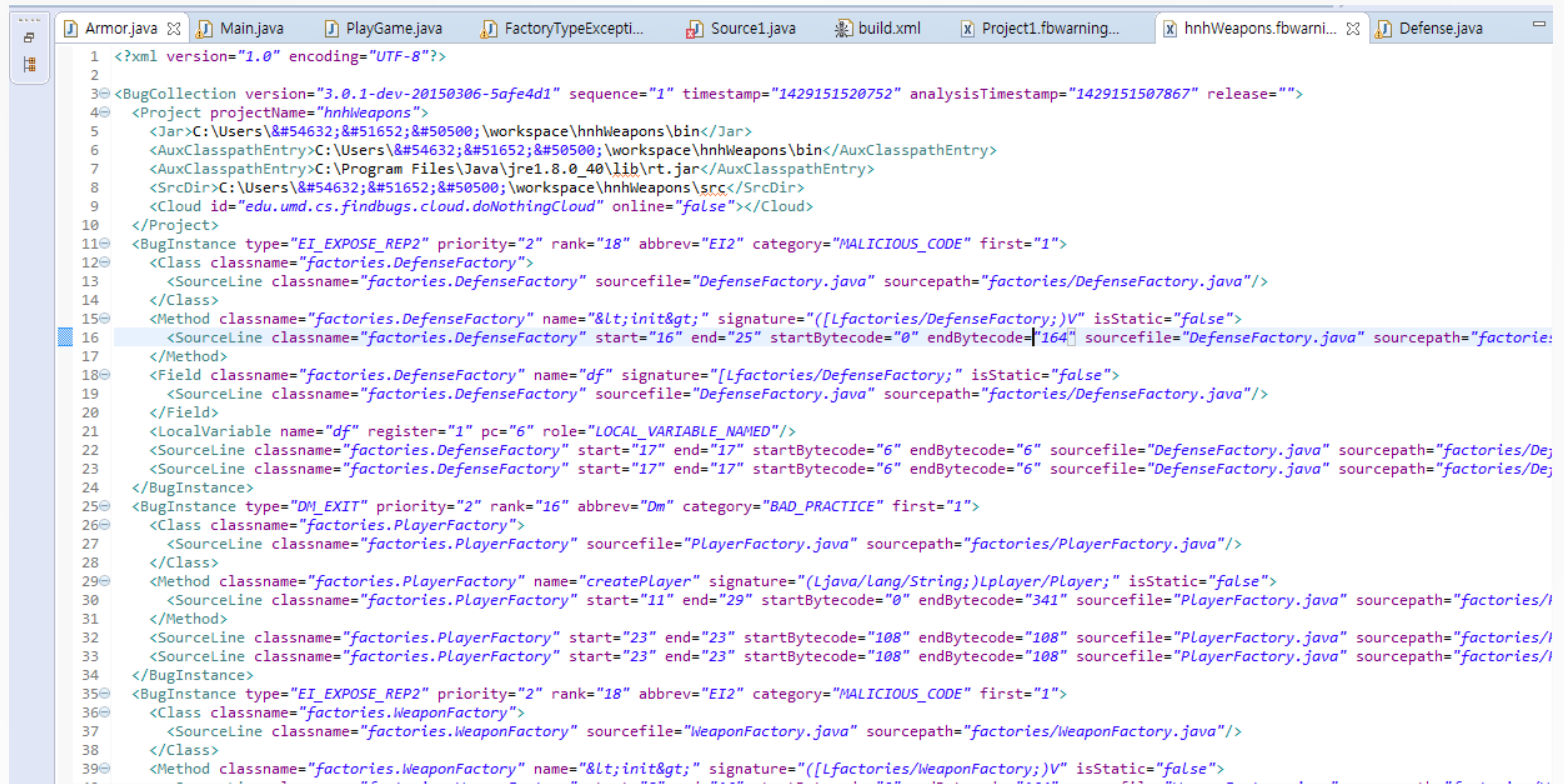
PART 2 Static analysis Tools

Eclipse : Findbugs plugin

3. Findbugs 결과 확인

프로젝트 우클릭 → findbugs → open analysis result in editor

Save XML을 통해 결과를 xml로 저장 가능



```

1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <BugCollection version="3.0.1-dev-20150306-Safe4d1" sequence="1" timestamp="1429151520752" analysisTimestamp="1429151507867" release="">
4   <Project projectName="hhWeapons">
5     <Jar>C:\Users\&#54632;&#51652;&#50500;\workspace\hhWeapons\bin</Jar>
6     <AuxClasspathEntry>C:\Users\&#54632;&#51652;&#50500;\workspace\hhWeapons\bin</AuxClasspathEntry>
7     <AuxClasspathEntry>C:\Program Files\Java\jre1.8.0_40\lib\rt.jar</AuxClasspathEntry>
8     <SrcDir>C:\Users\&#54632;&#51652;&#50500;\workspace\hhWeapons\src</SrcDir>
9     <Cloud id="edu.umd.cs.findbugs.cloud.doNothingCloud" online="false"></Cloud>
10  </Project>
11  <BugInstance type="EI_EXPOSE_REP2" priority="2" rank="18" abbrev="EI2" category="MALICIOUS_CODE" first="1">
12    <Class classname="factories.DefenseFactory">
13      <SourceLine classname="factories.DefenseFactory" sourcefile="DefenseFactory.java" sourcepath="factories/DefenseFactory.java"/>
14    </Class>
15    <Method classname="factories.DefenseFactory" name="&lt;init&gt;" signature="(Ljava/lang/Class;)V" isStatic="false">
16      <SourceLine classname="factories.DefenseFactory" start="16" end="25" startBytecode="0" endBytecode="164" sourcefile="DefenseFactory.java" sourcepath="factories/DefenseFactory.java"/>
17    </Method>
18    <Field classname="factories.DefenseFactory" name="df" signature="[Ljava/lang/String;" isStatic="false">
19      <SourceLine classname="factories.DefenseFactory" sourcefile="DefenseFactory.java" sourcepath="factories/DefenseFactory.java"/>
20    </Field>
21    <LocalVariable name="df" register="1" pc="6" role="LOCAL_VARIABLE_NAMED"/>
22    <SourceLine classname="factories.DefenseFactory" start="17" end="17" startBytecode="6" endBytecode="6" sourcefile="DefenseFactory.java" sourcepath="factories/DefenseFactory.java"/>
23    <SourceLine classname="factories.DefenseFactory" start="17" end="17" startBytecode="6" endBytecode="6" sourcefile="DefenseFactory.java" sourcepath="factories/DefenseFactory.java"/>
24  </BugInstance>
25  <BugInstance type="DM_EXIT" priority="2" rank="16" abbrev="Dm" category="BAD_PRACTICE" first="1">
26    <Class classname="factories.PlayerFactory">
27      <SourceLine classname="factories.PlayerFactory" sourcefile="PlayerFactory.java" sourcepath="factories/PlayerFactory.java"/>
28    </Class>
29    <Method classname="factories.PlayerFactory" name="createPlayer" signature="(Ljava/lang/String;)Ljava/lang/String;" isStatic="false">
30      <SourceLine classname="factories.PlayerFactory" start="11" end="29" startBytecode="0" endBytecode="341" sourcefile="PlayerFactory.java" sourcepath="factories/PlayerFactory.java"/>
31    </Method>
32    <SourceLine classname="factories.PlayerFactory" start="23" end="23" startBytecode="108" endBytecode="108" sourcefile="PlayerFactory.java" sourcepath="factories/PlayerFactory.java"/>
33    <SourceLine classname="factories.PlayerFactory" start="23" end="23" startBytecode="108" endBytecode="108" sourcefile="PlayerFactory.java" sourcepath="factories/PlayerFactory.java"/>
34  </BugInstance>
35  <BugInstance type="EI_EXPOSE_REP2" priority="2" rank="18" abbrev="EI2" category="MALICIOUS_CODE" first="1">
36    <Class classname="factories.WeaponFactory">
37      <SourceLine classname="factories.WeaponFactory" sourcefile="WeaponFactory.java" sourcepath="factories/WeaponFactory.java"/>
38    </Class>
39    <Method classname="factories.WeaponFactory" name="&lt;init&gt;" signature="(Ljava/lang/Class;)V" isStatic="false">

```

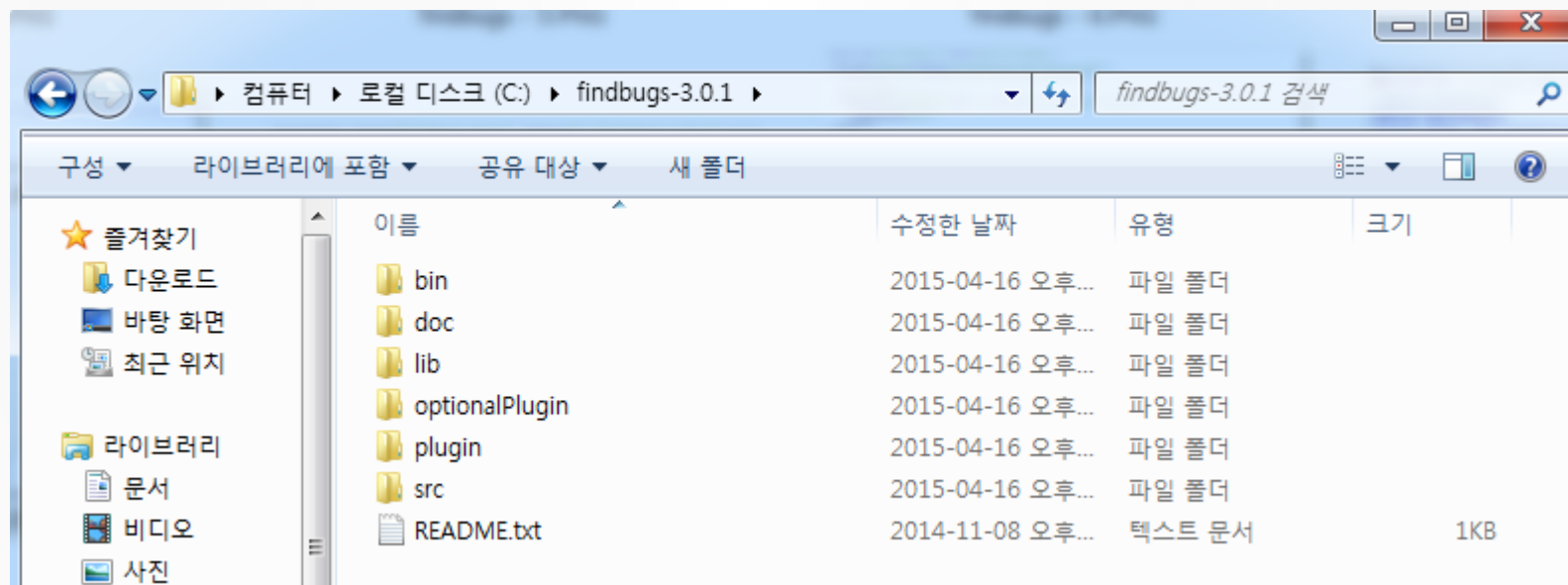
PART 2 Static analysis Tools

Ant : Findbugs setting

1. Findbugs 다운

<http://sourceforge.net/projects/findbugs/files/findbugs/3.0.1/findbugs-3.0.1.zip/download>

→ Findbugs 3.0.1 버전을 다운 받아서 압축 해제



PART 2 Static analysis Tools

Ant : Findbugs setting

2. build.xml 파일 설정

```
<property name="findbugs.home" value="C:/findbugs-3.0.1" />
```

```
<path id="findbugs.classpath">
  <pathelement location="${build}" />
  <fileset dir="${findbugs.home}/lib/">
    <include name="*.jar" />
  </fileset>
</path>
```

- 1) Findbugs.home : findbugs 설치 경로 지정
- 2) classpath : findbugs 설치 경로 내의 library 폴더 지정

```
<property name="report.findbugs.dir" location="${report.dir}/findbugs" />
```

```
<target name="findbugs" description="Run code analysis over code to check for problems.">
  <taskdef name="findbugs" classname="edu.umd.cs.findbugs.anttask.FindBugsTask" classpathref="findbugs.classpath"/>
```

```
<!-- Run FindBugs. -->
```

```
<mkdir dir="${report.findbugs.dir}" />
```

```
<findbugs home="${findbugs.home}" output="xml" outputFile="${report.findbugs.dir}/findbugs.xml">
```

```
  <sourcePath path="src" />
```

```
  <class location="bin" />
```

```
</findbugs>
```

```
</target>
```

- 3) 실제 findbugs가 수행되는 <findbugs> 태그
output, outputFile : 결과 xml 파일 경로 및 이름 지정
class location : 분석할 .class 파일들이 있는 경로 폴더

```
<target name="report.findbugs" description="Generate a report on error analysis.">
```

```
  <xslt in="${report.findbugs.dir}/findbugs.xml" style="${findbugs.home}/src/xsl/fancy.xsl" out="${report.findbugs.dir}/findbugs-default.html" />
```

```
</target>
```

- 4) 결과 xml 파일을 html 형태로 저장



PART 2 Static analysis Tools

Ant : Findbugs setting

3. build 완료 및 findbugs 수행

```
Buildfile: C:\Users\홍진아\workspace\Repository1\hnhWeapons\build.xml
report.findbugs:
  [xslt] Processing C:\Users\홍진아\workspace\Repository1\hnhWeapons\${report.dir}\findbugs\findbugs.xml to C:\Users\홍진아\w
  [xslt] Loading stylesheet C:\findbugs-3.0.1\src\xsl\fancy.xsl
BUILD SUCCESSFUL
Total time: 2 seconds
```

지정한 경로에 지정한 이름으로 결과 xml, html파일 저장된 것 확인 가능.

 findbugs.xml	2015-04-16 오후...	XML 문서	49KB
 findbugs-default.html	2015-04-16 오후...	HTML 문서	45KB

PART 2 Static analysis Tools

Ant : Findbugs setting

3. build 완료 및 findbugs 수행

결과 xml 및 html 화면

```
<?xml version="1.0" encoding="UTF-8"?>
- <BugCollection release="" analysisTimestamp="1429154262467" timestamp="1429154262436" sequence="0" version="3.0.1">
  - <Project projectName="">
    <Jar>C:\Users\함진아\workspace\hnhWeapons\bin</Jar>
    <SrcDir>C:\Users\함진아\workspace\hnhWeapons\src\gui</SrcDir>
  </Project>
  - <BugInstance category="MALICIOUS_CODE" abbrev="EI2" rank="18" priority="2" type="EI_EXPOSE_REP2">
    - <Class classname="factories.DefenseFactory">
      <SourceLine classname="factories.DefenseFactory" sourcepath="factories/DefenseFactory.java" sourcefile="DefenseFactory.java" end="63" start="12"/>
    </Class>
    - <Method classname="factories.DefenseFactory" isStatic="false" signature="([Lfactories/DefenseFactory;])V" name="<init>">
      <SourceLine classname="factories.DefenseFactory" sourcepath="factories/DefenseFactory.java" sourcefile="DefenseFactory.java" end="25" start="16"
        endBytecode="164" startBytecode="0"/>
    </Method>
    - <Field classname="factories.DefenseFactory" sourcepath="factories/DefenseFactory.java" sourcefile="DefenseFactory.java" end="25" start="16"
      endBytecode="164" startBytecode="0"/>
    </Field>
    <LocalVariable role="LOCAL_VARIABLE_NAME" sourcepath="factories/DefenseFactory.java" sourcefile="DefenseFactory.java" end="25" start="16"
      endBytecode="6" startBytecode="6"/>
  </BugInstance>
  - <BugInstance category="BAD_PRACTICE" abbrev="V" rank="1" priority="1" type="V">
    - <Class classname="factories.PlayerFactory">
      <SourceLine classname="factories.PlayerFactory" sourcepath="factories/PlayerFactory.java" sourcefile="PlayerFactory.java" end="12" start="12"/>
    </Class>
  </BugInstance>
</BugCollection>
```

FindBugs (3.0.1) Analysis for

Bug Summary Analysis Information List bugs by bug category List bugs by package

FindBugs Analysis generated at: Thu, 16 Apr 2015 12:17:42 +0900

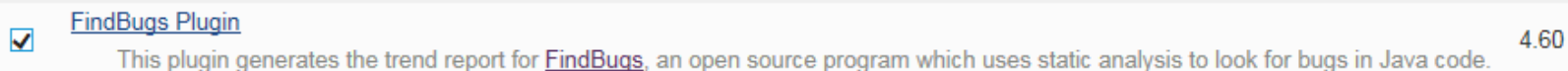
Package	Code Size	Bugs	High Prio Bugs	Medium Prio Bugs	Low Prio Bugs	Exp. Bugs	Ratio
Overall (7 packages), (89 classes)	3315	34	9	25			
factories	237	3		3			
gui	1719	10	2	8			
items	60	3	2	1			
items.defenses	85	1	1				
items.weapons	431	1	1				
player	487	6	1	5			
stores	296	10	2	8			

PART 2 Static analysis Tools

Jenkins : Findbugs plugin

1. Findbugs plugin 설치 및 build 설정

jenkins 설정 → 플러그인 관리 → 설치 가능 → Findbugs plugin 체크 및 설치

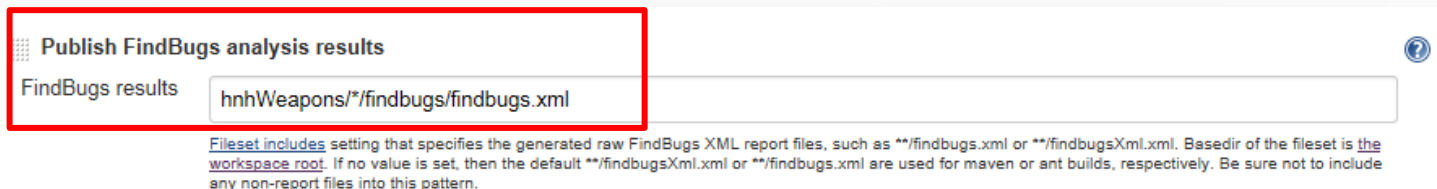


Jenkins job 구성 → Build → Invoke Ant의 Target

ant 설정에서 Findbugs 수행을 위해 build.xml에 추가한 target을 입력



빌드 후 조치 → Publish findbugs analysis results → 추출된 findbugs 결과 xml



PART 2 Static analysis Tools

Jenkins : Findbugs plugin

2. jenkins dashboard에서 Findbugs의 결과 확인

프로젝트의 왼쪽 메뉴에 Findbugs Warnings가 추가 되고, 클릭하면 dashboard 형태로 결과 확인

- [프로젝트로 돌아가기](#)
- [상태](#)
- [바뀐점](#)
- [Console Output](#)
- [빌드 정보 수정](#)
- [이 빌드를 삭제](#)
- [Tag this build](#)
- [FindBugs Warnings](#)
- [이전 빌드](#)

FindBugs Result

Warnings Trend

All Warnings	New this build	Fixed Warnings
34	34	0

Summary

Total	High Priority	Normal Priority	Low Priority
34	9	25	0

Details

Package	Total	Distribution
factories	3	
gui	10	
items	3	
items.defenses	1	
items.weapons	1	
player	6	
stores	10	
Total	34	


PART 2 Static analysis Tools


Jenkins : Findbugs plugin

2. jenkins dashboard에서 Findbugs의 결과 확인

프로젝트의 메인 화면에서 build #에 따른 findbugs warnings의 경향 확인 가능

Project hnhWeapons

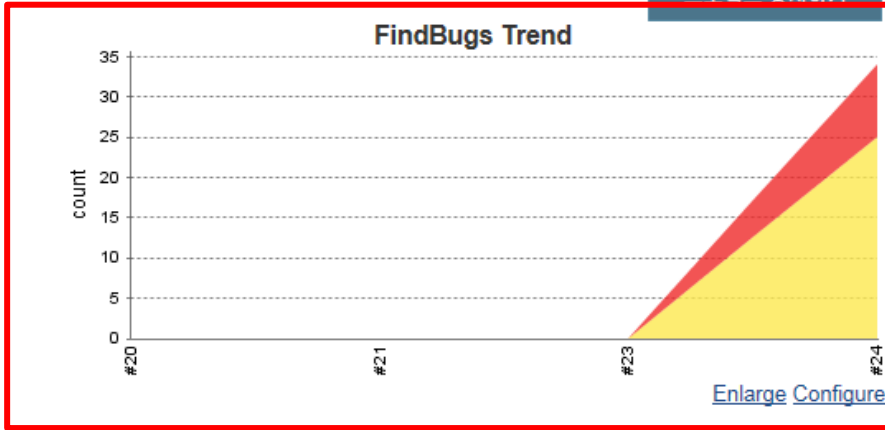
 [작업 공간](#)

 [최근 변경사항](#)

고정링크

- [Last build, \(#24\), 53 min 전](#)
- [Last stable build, \(#24\), 53 min 전](#)
- [Last successful build, \(#24\), 53 min 전](#)
- [Last failed build, \(#22\), 1 hr 1 min 전](#)
- [Last unsuccessful build, \(#22\), 1 hr 1 min 전](#)

FindBugs Trend



Build #	Count
#20	0
#21	0
#23	0
#24	35

[Enlarge](#) [Configure](#)

PART 3

System analysis Tool

PART 3 System analysis Tool

What is TestLink ?

- 요구사항으로 부터 테스트케이스를 유도해내서 관리할 수 있는 시스템통합 도구다.
- 테스트 도구는 아니고, 테스트 프로세스를 관리할 수 있다.
- 웹에서 사용하기 때문에, os구분은 없고 apm 환경이 필요하다.

주요 분석 목적 : 테스트 플랜& 케이스 관리, 문서작성



PART 3 System analysis Tool

What is TestLink ?

The screenshot displays the TestLink 1.9.11 web interface. At the top, the user is logged in as 'admin [admin]' with links for 'My Settings' and 'Logout'. The navigation bar includes 'Desktop', 'Requirements', 'Test Specification', 'Test Execution', 'Test Reports', 'Users/Roles', 'Events', and a search box. The current test project is 'foretestlink'. The sidebar on the left is organized into several categories:

- System**
 - Define Custom Fields
 - Issue Tracker Management
- Test Project**
 - Test Project Management
 - Assign User Roles
 - Assign Custom Fields
 - Keyword Management
 - Platform Management
 - Inventory
- Requirements**
 - Requirement Specification
 - Requirement Overview
 - Search Requirements
 - Search Requirement Specifications
 - Assign Requirements
 - Generate Requirement Specification Document
- Test Specification**
 - Test Specification
 - Search Test Cases
 - Test Cases created per User
- Documentation**

On the right side, the 'Current Test Plan' is set to 'testplanner'. Below this, there are three main sections:

- Test Plan**
 - Test Plan Management
 - Builds / Releases
 - Assign User Roles
 - Milestone Overview
- Test Execution**
 - Execute Tests
 - Test Cases Assigned to Me
 - Test Reports and Metrics
 - Metrics Dashboard
- Test Plan contents**
 - Add / Remove Platforms
 - Add / Remove Test Cases
 - Update Linked Test Case Versions
 - Show Test Cases Newest Versions
 - Assign Test Case Execution
 - Set Urgent Tests

PART 3 System analysis Tool

Redmine : Issue tracker

- Issue tracker와 연동할 수 있다.
- 레드마인이 설치된 상태기 때문에 redmine의 url을 등록 한다.

The screenshot displays the TestLink web interface. At the top, the 'Issue Tracker Management' window is open, showing the 'Edit' configuration for a Redmine issue tracker. The configuration includes the name 'redmine', the type 'redmine (Interface: rest)', and the XML configuration for the issue tracker.

```
<issuetracker>
<uribase>http://localhost/redmine/</uribase>
<uriview>http://localhost/issues/</uriview>
</issuetracker>
```

The main TestLink interface shows the user 'admin [admin]' and the current test project 'forestlink'. Below the navigation bar, there is a table listing the configured issue trackers:

Issue Tracker	Type	Environment	delete
redmine	redmine (Interface: rest)	OK	

A 'Create' button is visible below the table. At the bottom, the 'Used on Test Project' section shows 'newtest'.

PART 3 System analysis Tool

How to use Testlink

1. 프로젝트 생성

이름, Prefix, 추가 항목, 이슈트래커 설정가능

Create from existing Test Project?

Name *

Prefix (used for Test case ID) *

Project description

body p

Enhanced features

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

Issue Tracker Integration

- Active

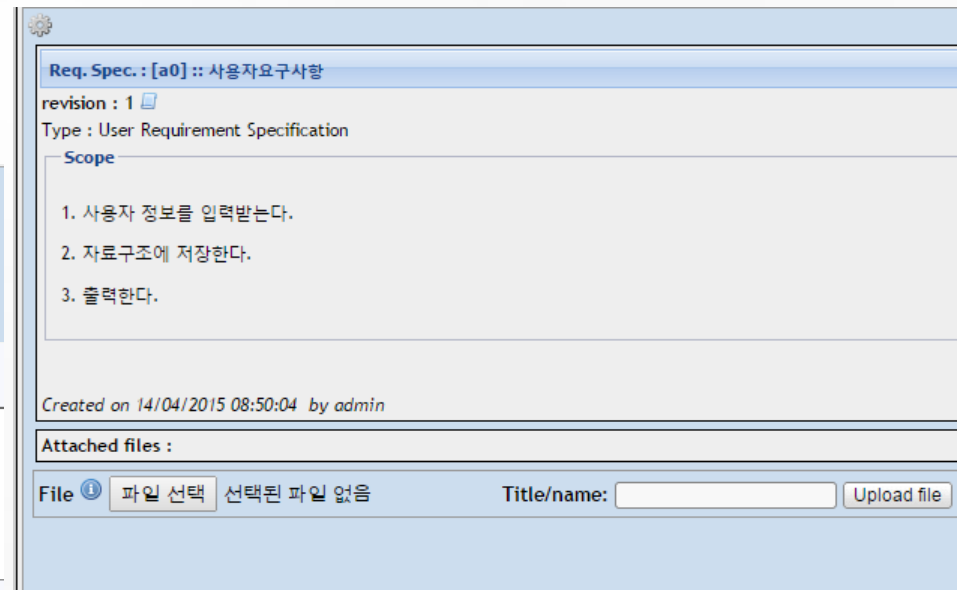
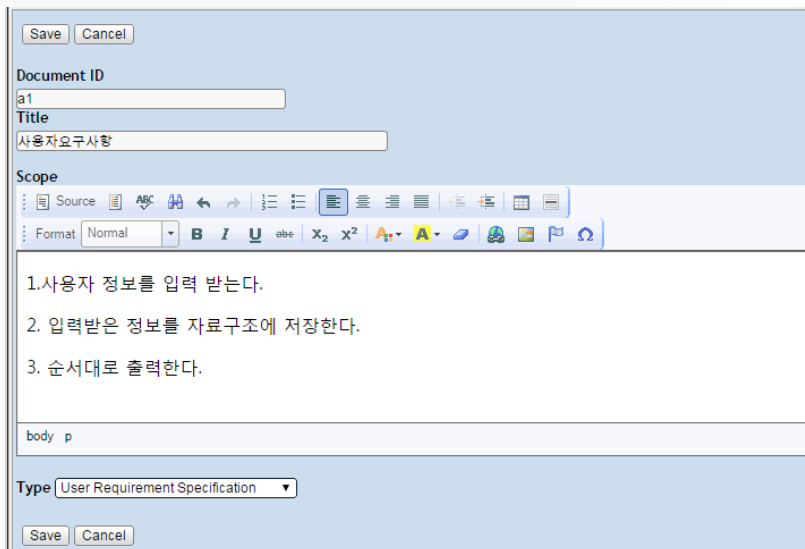
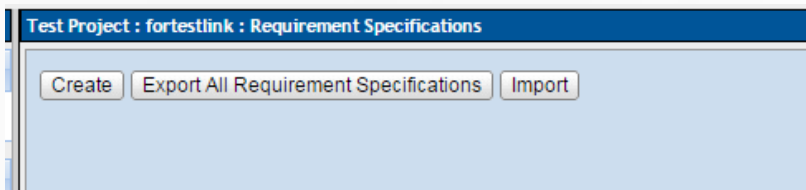
Issue Tracker

PART 3 System analysis Tool

How to use Testlink

2. 요구사항 만들기

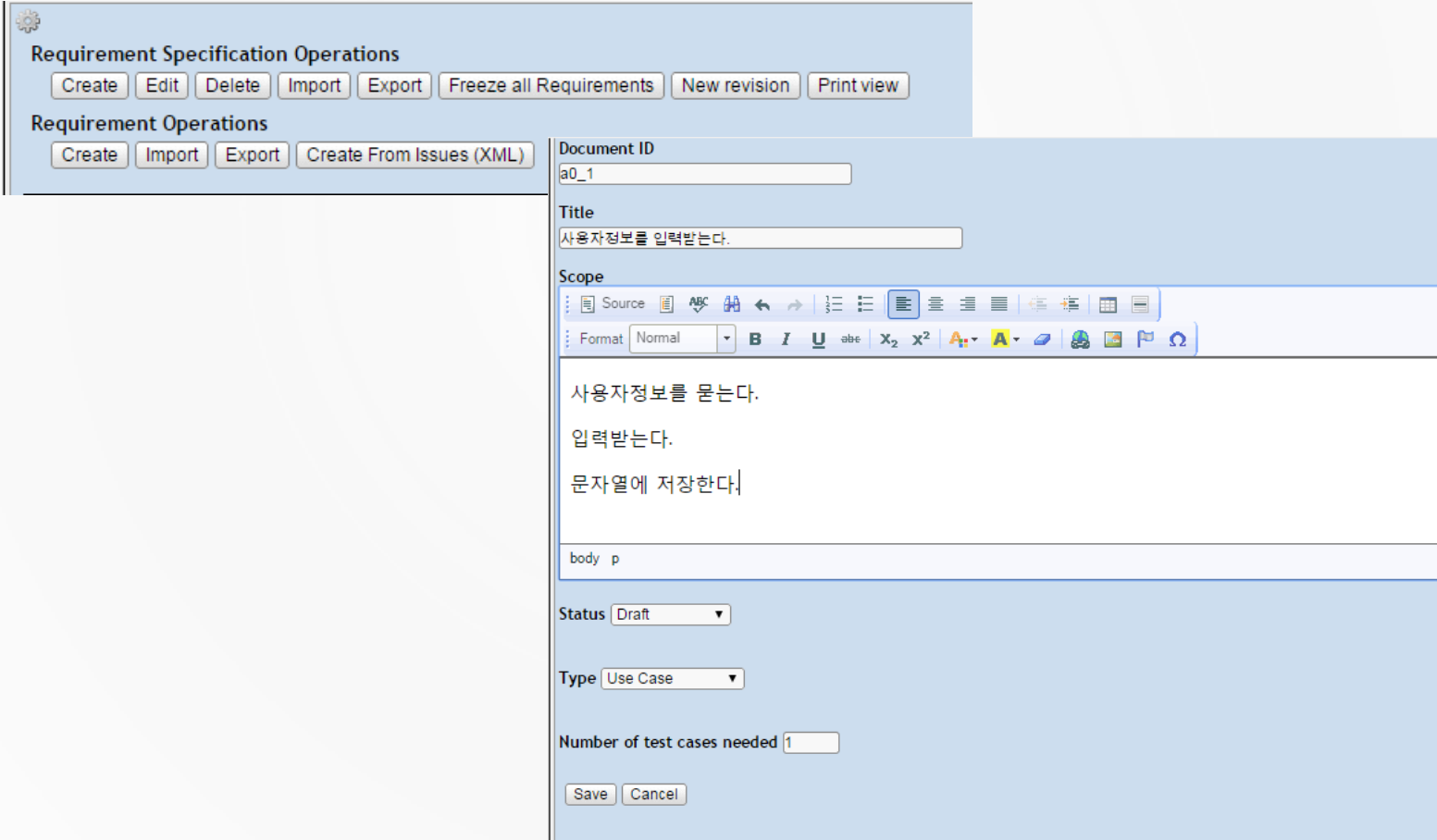
type도 설정 가능



PART 3 System analysis Tool

How to use Testlink

3. 요구사항에 대한 operation 만들기



The image shows the 'Requirement Specification Operations' dialog box in Testlink. It is divided into two main sections: 'Requirement Specification Operations' and 'Requirement Operations'.

Requirement Specification Operations:

- Buttons: Create, Edit, Delete, Import, Export, Freeze all Requirements, New revision, Print view

Requirement Operations:

- Buttons: Create, Import, Export, Create From Issues (XML)

Document ID: a0_1

Title: 사용자정보를 입력받는다.

Scope:

The scope section contains a rich text editor with the following content:

```

사용자정보를 묻는다.
입력받는다.
문자열에 저장한다.

```

The status bar below the editor shows: body p

Status: Draft

Type: Use Case

Number of test cases needed: 1


Buttons: Save, Cancel

PART 3 System analysis Tool

How to use Testlink

4. Test suite 만들기


Test Project : Example

 Test Suite Operations

Test Project Name
Example

Description
이해를 돕기위해 만드는 프로젝트 입니다.

Attached files :

File  선택된 파일 없음 Title/name:

PART 3 System analysis Tool

How to use Testlink

5. Specification

▼테스트 케이스 만들기. 스텝 밟을 수 있다.

The screenshot displays the Testlink interface for creating a test case. The main window is titled 'EX-1:이름 입력받기' and shows the following details:

- Version 1** (with a red bug icon and an info icon)
- Summary:** 이름을 입력 받는다.
- Preconditions:** 문자열 생성, scanner 생성.
- Input:** 입력받기
- Step actions table:**

#	Step actions	Expected Results		
1	문자열생성	문자열이 만들어짐	✖	⊕
2	스캐너 생성	스캐너 생성	✖	⊕
3	문자열에 입력받기.	입력된 정보가 저장됨.	✖	⊕
- Buttons:** Create step, Resequence Steps
- Status:** Draft (dropdown)
- Importance:** Medium (dropdown)
- Estimated exec. (min):** [input field]
- Save** button

On the left side, there is a sidebar for 'Test Suite : 정보입력받기' with a 'Details' section containing input fields for '이름', '나이', and '학번'. Below this, there is a 'Keywords' field set to 'None' and an 'Attached files' section with a 'File' button, a '파일 선택' button, and a '선택된 파일 없음' message. At the bottom, there is a 'Title/name:' input field and an 'Upload file' button.

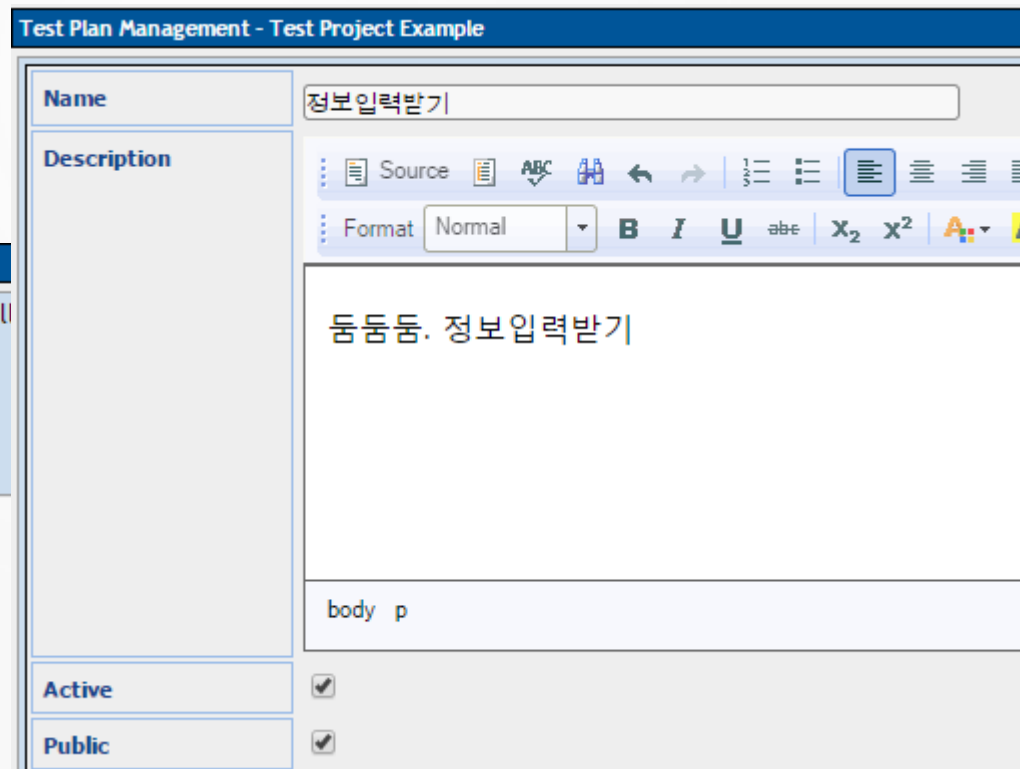
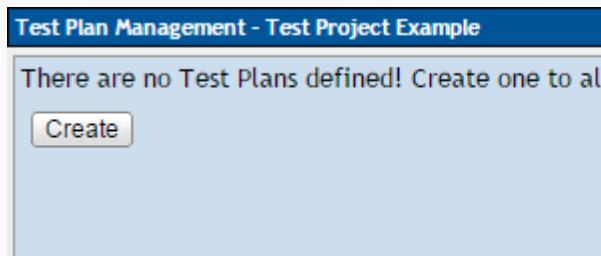
PART 3 System analysis Tool

How to use Testlink

6. Test Plan

실제 테스트 execution을 계획할 수 있다.

단위별로, 버전별로도 가능하며, 일정 조율도 가능할 듯 하다.

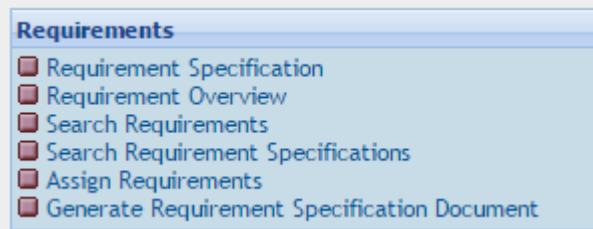


PART 3 System analysis Tool

How to use Testlink

6. Test Plan

Plan에 할당하기



Test Suite : 정보입력받기 ?

Requirements Bulk Assignment

Requirements Specification Document [a0] - 사용자요구사항 ▼

This operation will assign selected requirements to (maximum number of) 1 test cases belonging to test suites present on selected branch.

Available Requirements

<input checked="" type="checkbox"/>	Document ID	Requirement	Scope
<input checked="" type="checkbox"/>	a0_1	사용자정보를 입력받는다.	사용자정보를 묻는다. 입력받는다. 문자열에...

Assign

PART 3 System analysis Tool

How to use Testlink

6. Test Plan

Plan 생성

Build management - Test Plan : 정보입력받기

Create a new Build

Title	Build1
Description	<div style="border: 1px solid #ccc; padding: 5px;"> <p>첫시도입니다.</p> <p>body p</p> </div>
Active	<input checked="" type="checkbox"/>
Open	<input checked="" type="checkbox"/>
Release date	15/04/2015

A build is identified by its title. Each build is related to the active Test Plan.

Description should include: list of delivered packages, fixes or features, approvals, status, etc.

A build has two attributes:

Active / Inactive - defines whether the build can be used. Inactive builds are not listed on the execution and reports pages.

Open / Closed - Only for open builds, test results can be modified.

Create Cancel

PART 3 System analysis Tool

How to use Testlink

6. Test Plan

Plan 생성

Test Plan : 정보입력받기 - Add Test Cases to Test Plan

Assign to user on add on build Send mail notification to tester

Check/uncheck all Test cases for

정보입력받기

<input checked="" type="checkbox"/> Test Case	Version	Importance	
<input checked="" type="checkbox"/> EX-1 : 이름 입력받기	<input type="text" value="1"/>	Medium	<input type="text" value="1000"/>

Test Plan : 정보입력받기 - Add/Remove Test Cases to/from Test Plan

Assign to user on add on build Send mail notification to tester

Check/uncheck all Test cases for

정보입력받기

<input checked="" type="checkbox"/> Test Case	Version	Importance			
<input checked="" type="checkbox"/> EX-1 : 이름 입력받기	<input type="text" value="1"/>	Medium	<input type="text" value="1000"/>	<input type="checkbox"/>	<input type="text" value="14/04/2015"/>

PART 3 System analysis Tool

How to use Testlink

7. Test 실행

Test Results on Build Build1

Print Show complete execution history Import Results (XML)

- Test Plan 정보입력받기
- Build Build1
- Test Suite : 정보입력받기/

Latest execution (any build)

Not Run

Not Tested Yet

Test Case EX-1 :: Version : 1 :: 이름 입력받기
Assigned to : admin

Summary
이름을 입력 받는다.

Preconditions
문자열 생성
scanner 생성.
입력받기

PART 3 System analysis Tool

How to use Testlink

7. Test 실행

할당 받은 일을 Tester가 진행.

#	Step actions	Expected Results	Execution notes	Result
1	문자열생성	문자열이 만들어짐	할당했어요!	Passed
2	스캐너 생성	스캐너 생성	스캐너생성!	Passed
3	문자열에 입력받기.	입력된 정보가 저장됨.	저장하기!	Passed

Execution type : Manual
Estimated exec. duration (min) :

Latest execution (any build)

Date : 14/04/2015 09:06:14 - Tested by : admin - Build : Build1 - Status : Passed

Latest execution (current build) - Build : Build1

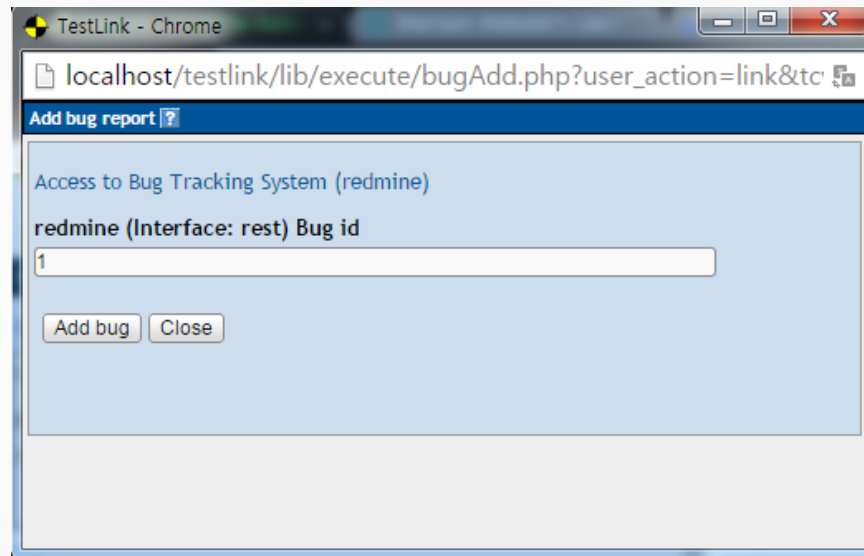
Date	Build	Tested by	Status	Exec (min)	Version	BUG management	Run mode
14/04/2015 09:06:14	Build1	admin	Passed		1		

Test Case EX-1 :: Version : 1 :: 이름 입력받기
Assigned to : admin

PART 3 System analysis Tool

How to use Testlink







8. 이슈 등록



Latest execution (any build)

Date : 14/04/2015 09:06:14 - Tested by : admin - Build : Build1 - Status : Passed

Latest execution (current build) - Build : Build1

Date	Build	Tested by	Status	Exec (min)	Version	BUG management	Run mode
14/04/2015 09:06:14	Build1	admin	Passed		1	 	  
Build		Relevant bugs					
Build1		1 : [해결] : 아 어떻게 되려나 					

PART 3 System analysis Tool

How to use Testlink

9. Metric Dashboard

Show metrics only for active test plans

Test Project Progress

Progress: 100% [1/1]

Not Run: 0% [0/1]

Passed: 100% [1/1]

Failed: 0% [0/1]

Blocked: 0% [0/1]

The project progress is calculated by summing up all executions for a specific execution result for all platforms within all test plans. This sum is then set into relation to the sum of all active test cases for all platforms within all test plans.

Test Plan Progress (ATTENTION: Only Test Plans with ACTIVE BUILDS are considered)

Metrics Dashboard

Expand/Collapse Groups Show all Columns Reset to Default State Refresh Reset Filters MultiSort

Test Plan	Active Test Cases	Not Run [%]	Passed [%]	Failed [%]	Blocked [%]	Progress [%]
정보입력받기 - Overall Progress: 100%	1	0	100	0	0	100

*This Report shows the progress for the whole project and all (or only active) test plans defined within project.
 "Overall Progress" for test plans is calculated based upon the progress of each platform (if in use).
 "Test Project Progress" is calculated based upon the progress of all (or only active) test plans.*

Generated by TestLink on 14/04/2015 09:07:51

PART 3 System analysis Tool

How to use Testlink

10. Test 결과 보고

Reports and Metrics

Report Format: HTML Print

Test Plan: 정보입력받기

Show inactive test plans

- Test Plan Report
- Test Report
 - Test Report on build
- General Test Plan Metrics
- Results by Tester per Build
 - Test Case Assignment Overview
- Test result matrix
- Failed Test Cases
 - Blocked Test Cases
 - Not run Test Cases
 - Test Cases without Tester Assignment
- Charts
 - Requirements based Report
 - Bugs per Test Case
 - Test Cases with Custom Fields set on Execution
 - Test Plan with Custom Field info
 - Test Cases not assigned to Any Test Plan

Description of Test Reports and Metrics

Reports are related to a Test Plan (defined in top of navigator). This Test Plan could differ from the current Test Plan for execution. You can also select a Report format:

- Normal - report is displayed in web page
- OpenOffice Writer - report imported to OpenOffice Writer
- OpenOffice Calc - report imported to OpenOffice Calc
- MS Excel - report imported to Microsoft Excel
- HTML Email - report is emailed to user's email address
- Charts - report include graphs (flash technology)

The print button activates printing of a report only (without navigation).

There are several separate reports to choose from, their purpose and function are explained below.

Test Plan

The document 'Test Plan' has options to define a content and a document structure.

Test Report

The document 'Test Report' has options to define a content and document structure. It includes Test cases together with test results.

General Test Plan Metrics

This page shows you only the most current status of a Test plan by test suite, owner, and keyword. The most 'current status' is determined by the most recent build test cases were executed on. For instance, if a test case was executed over multiple builds, only the latest result is taken into account.

'Last Test Result' is a concept used in many reports, and is determined as follows:

- The order in which builds are added to a Test Plan determines which build is most recent. The results from the most recent build will take precedence over older builds. For example, if you mark a test as 'fail' in build 1, and mark it as 'pass' in build 2, it's latest result will be 'pass'.
- If a test case is executed multiple times on the same build, the most recent execution will take precedence. For example, if build 3 is released to your team and tester 1 marks it as 'pass' at 2PM, and tester 2 marks it as 'fail' at 3PM - it will appear as 'fail'.
- Test cases listed as 'not run' against a build are not taken into account. For example, if you mark a case as 'pass' in build 1, and don't execute it in build 2, it's last result will be considered as 'pass'.

PART 3 System analysis Tool

How to use Testlink

11. 요구사항 문서

Test Plan: 정보입력받기

Example (1)

a0:사용자요구사항 (1)

1. a0 : 사용자요구사항

a0_1 : 사용자정보를 입력받는다.

1. Requirements Spec.: a0 : 사용자요구사항

<u>revision</u>	1
<u>Author</u>	admin
<u>Type</u>	User Requirement Specification
<u>Total count of requirements (Coverage)</u>	N/A (1/0)

1. 사용자 정보를 입력받는다.

2. 자료구조에 저장한다.

3. 출력한다.

Requirement: a0_1 : 사용자정보를 입력받는다.

<u>Version :</u>	1
<u>revision:</u>	1
<u>Author:</u>	admin
<u>Status</u>	Draft
<u>Type</u>	Use Case
<u>Coverage</u>	100% (1/1)

사용자정보를 묻는다.

입력받는다.

문자열에 저장한다.

<u>Related Test Cases</u>	1: 이름 입력받기
---------------------------	------------

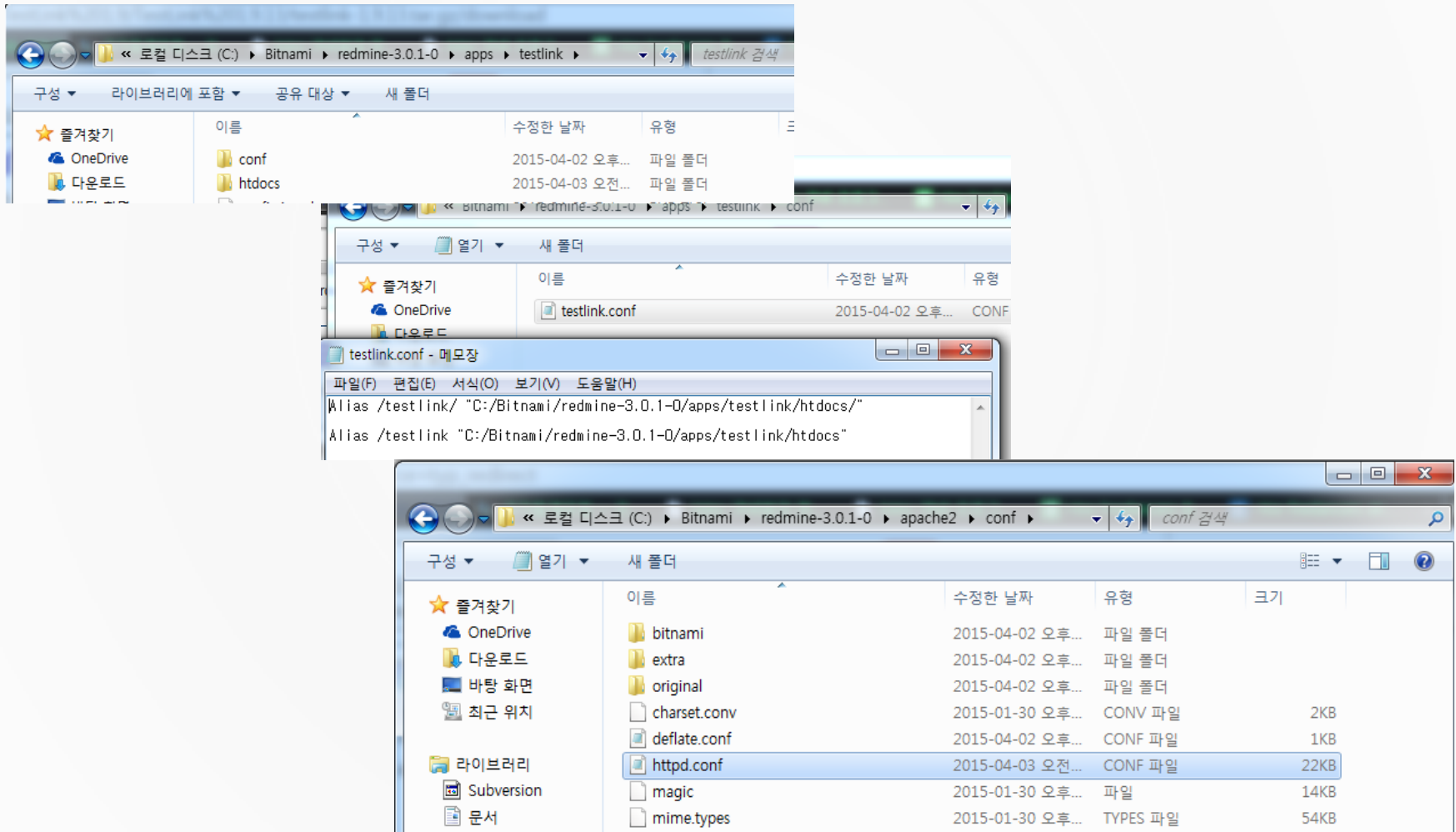
PART 3 System analysis Tool

How to install Testlink

- 홈페이지에서 testlink-1.9.13.targz 다운로드
- apps 폴더에 testlink 생성. 그 안에 conf와 htdocs 생성.
- Htdocs에 압축해제
- Conf폴더에 testlink.conf 파일 생성
- Bitnami->redmine->apache2->conf-> httpd.conf파일 변경
- startRedmine에서 mysql 실행 -> table 생성 -> 권한부여
- Config.inc.php파일 변경
- Testlink 설치...

PART 3 System analysis Tool

How to install Testlink



PART 3 System analysis Tool

How to install Testlink

- 따라서 작성한다.

```
#  
# Deny access to the entirety of your server's filesystem. You must  
# explicitly permit access to web content directories in other  
# <Directory> blocks below.  
#  
<Directory />  
    AllowOverride none  
    Require all granted  
</Directory>
```

```
#  
# "C:/Bitnami/redmine-3.0.1-0/apache2/cgi-bin" should be changed to whatever you  
# CGI directory exists, if you have that configured.  
#  
<Directory "C:/Bitnami/redmine-3.0.1-0/apache2/cgi-bin">  
    AllowOverride None  
    Options None  
    Require all granted  
</Directory>
```

```
Include "C:/Bitnami/redmine-3.0.1-0/apps/testlink/conf/testlink.conf"  
  
<Directory "C:/Bitnami/redmine-3.0.1-0/apps/testlink">  
    AllowOverride None  
    Options None  
    Require all granted  
</Directory>
```

PART 3 System analysis Tool

How to install Testlink

- Mysql 설정

Here are the steps needed for a successful installation:

- Unpack the release file into directory inside your document root
For installing the DB you can either choose the command line tools available in your MySQL installation or any MySQL Database Client
- Create a new empty MySQL database.
mysql> create database testlink;
- Install the sql into the newly created database.
mysql -u <user> -p<password> <dbname> < <testlinkdir>/install/sql/testlink_create_tables.sql
e.g.: "mysql -u testlink -ppass testlink < /var/www/html/testlink/install/sql/testlink_create_tables.sql"
- Edit <testlinkdir>/config.inc.php to match your configuration. See [Configuration](#) for more.
- On Linux or UNIX you must change the permissions of the templates_c directory to be writable by the webserver. From the TestLink
chmod 777 gui/templates_c
- Log into TestLink! Default credentials are:
user: admin; pass: admin
Changing this password is a good security practice.
- Report any issues or feedback to [TestLink](#) page or [testlink.org forums](#) within the TestLink 1.5 section.

PART 3 System analysis Tool

How to install Testlink

● Mysql 설정

▼ table 생성

```
Bitnami Redmine Stack Environment - mysql -u root -p
0)
C:\Bitnami\redmine-3.0.1-0>mysql -u root -p
Enter password:
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using
0)
C:\Bitnami\redmine-3.0.1-0>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.5.42 MySQL Community Server (GPL)

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> create database testlink;
Query OK, 1 row affected (0.00 sec)

mysql>
```

```
mysql> mysql -u testlink -ppassword testlink C:\Bitnami\redmine-3.0.1-0\apps\testlink\htdocs\install\sql\mysql\testlink_create_tables.sql
ERROR:
Unknown command 'WB'.
ERROR 1102 (42000): Incorrect database name 'edmine-3.0.1-0appstestlinkhtdocsinstallsqlmysqltestlink_create_tables.sql'
mysql> mysql -u testlink -ppassword testlink C:\Bitnami\redmine-3.0.1-0\apps\testlink\htdocs\install\sql\mysql\testlink_create_tables.sql

Copyright (c) 2000, 2015, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input
statement.

mysql> grant all privileges on *.* to david@localhost identified by 'wdjgks123';
Query OK, 0 rows affected (0.00 sec)

mysql>
```

▲ 아이디, 권한 생성

PART 3 System analysis Tool

How to install Testlink

- Htdocs폴더 → config.inc.php 수정

```

config.inc.php - 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)
<?php * TestLink Open Source Project - http://testlink.sourceforge.net/ *
*/$tICfg = new stdClass();$tICfg->api = new stdClass();$tICfg->document_gener
REFIX') );define('DB_TABLE_PREFIX','');/** The root dir for the testl
>= 4.1) 'ISO-8859-1' or another Charset could be set for * backward compatabili
->charts_font_size = 8;/**
-----
/* [LOGGING] */
receive message on screen. */$tICfg->config_check_warning_mode = 'SILENT';/*
/**
only if SMTP server requires authentication */$smtp_username = ''; # userge
ment' => false);/** $tICfg->authentication['domain'] = array('DB','LDAP');
lwing configuration parameters are used to build ldap filter and ldap attri
venname';$tICfg->authentication['ldap_surname_field'] = 'sn';/** Enable/di
{ }~]+)*" . " @(?:[a-z0-9](?:[a-z0-9]+[a-z0-9])?#.)+[a-z0-9](?:[a-z0-9-]
-----
e'; // 'inline'$tICfg->gui->op_area_display->req_spec_container = 'none'; // 'i
->gui->round_corners = new stdClass();$tICfg->gui->round_corners->exec_history
ut size of test case search by id on navigation bar. * This value will be add
://docs.fckeditor.net/ for more information about ToolbarSets) * TestLink
itor['all'] = array( 'type' => 'none'); * //This configuration is useful only i
tion'] = array( 'type' => 'none');*/$tICfg->gui->text_editor = array();$tICf
on' => 2, 'testPlanContents' => 3);
or External ID in tree on * test suite level using reorder button */$tICfg
fg->reportsCfg->start_date_offset = (7*24*60*60); // one week/** minutes part i
/

```

PART 4

Make up for CTIP

PART 4 Make up for CTIP

Jenkins : Mail setting

Jenkins Location

Jenkins URL

System Admin e-mail address

- ▲ jenkins 서버 관리자 email 설정
- ▼ smtp설정을 통해 사용자 mail 환경 구축

E-mail로 알려줌

SMTP 서버

Default user e-mail suffix

Use SMTP Authentication

사용자명

비밀번호

SSL 사용

SMTP Port

Reply-To Address

Charset

Test configuration by sending test e-mail

PART 4 Make up for CTIP

Jenkins : Mail setting

빌드 후 조치

E-mail Notification ?

Recipients

Whitespace-separated list of recipient addresses. May reference build parameters like \$PARAM. E-mail will be sent when a build fails, becomes unstable or returns to stable.

Send e-mail for every unstable build

Send separate e-mails to individuals who broke the build ?

삭제

프로젝트 관리 → 설정 → 빌드 후 조치 → E-mail Notification

빌드에 오류가 난 경우, 또는 오류가 해결 된 경우 → 등록된 메일로 빌드 내용 발송

<input type="checkbox"/>	☆	▶	나	Jenkins build is back to normal : Project1 #7 - See < http://192.168.204.1:8080/job/Project1/7/changes >	19:38
<input type="checkbox"/>	☆	▶	나	Build failed in Jenkins: Project1 #6 - See < http://192.168.204.1:8080/job/Project1/6/changes > Changes: [user1	19:35

PART 4 Make up for CTIP

Redmine : Mail setting

- Bitnami로 redmine 설치시 관리자 계정의 email 설정을 할 수 있다.
- SMTP 설정 → gmail 선택 후 ID/PW 입력

The image displays three overlapping screenshots of the Redmine installation setup wizard, illustrating the configuration process for email notifications.

Left Screenshot: Create Admin account
This window shows the creation of a Bitnami Redmine Stack admin user. The fields are filled with the following information:
- Your real name: luckii
- Email Address: jinaham11@gmail.com
- Login: luckii
- Password: [masked]
- Please confirm your password: [masked]

Middle Screenshot: Configure SMTP Settings
This window prompts the user to configure mail support. The checkbox "Do you want to configure mail support?" is checked. The "Default email provider" is set to "GMail".
- Do you want to configure mail support?
- Default email provider: GMail

Right Screenshot: Configure SMTP Settings
This window shows the configuration of Gmail as the email provider. It includes a warning about storing data in configuration files and a recommendation to use personal account credentials.
- Gmail address: jinaham11@gmail.com
- Gmail password: [masked]
- Re-enter: [masked]

PART 4 Make up for CTIP

Redmine : Mail setting

- User 가입 시에, email을 작성하도록 되어 있어 모든 user가 email 계정을 등록한다.

등록

로그인 * luckii

비밀번호 * ●●●●●●
최소한 8 글자 이상이어야 합니다.

비밀번호 확인 * ●●●●●●

이름 * 진아

성 * 함

메일 * jinaham11@gmail.com

언어 Korean (한국어)

Users + New user

Filters

Status: active (1) User: Apply Clear

Login	First name	Last name	Email	Administrator	Created	Last connection
luckii	luckii	Admin	jinaham11@gmail.com	✓	04/14/2015 08:10 PM	04/14/2015 08:14 PM

(1-1/1)

PART 4 Make up for CTIP

Redmine : Mail setting

- 관리자 → setting → email modifications → 일감 등록이 있을 때마다 email 발송

Settings

General Display Authentication Projects Issue tracking **Email notifications** Incoming emails Repositories

Emission email address

Blind carbon copy recipients (bcc)

Plain text mail (no HTML)

Default notification option

Select actions for which email notifications should be sent.

- Issue added
- Issue updated
 - Note added
 - Status updated
 - Assignee updated
 - Priority updated
- News added
- Comment added to a news
- Document added
- File added
- Message added
- Wiki page added
- Wiki page updated

[Check all](#) | [Uncheck all](#)

PART 4 Make up for CTIP

Redmine : Mail setting

- 일감 등록 시에 담당자로 특정 user를 지목할 경우, 해당 user에게 email 발송

Issues

▼ Filters

Status Add filter

► Options

✓ Apply Clear Save

✓ #	Tracker	Status	Priority	Subject	Assignee	Updated
<input type="checkbox"/> 1	Bug	New	Normal	issue1	luckii Admin	04/14/2015 08:22 PM

[T4_project - Bug #1] (New) issue1 받은편지함 x

jinaham11@gmail.com 20:22 (0분 전) ☆

숨은 참조: 나에게

Issue #1 has been reported by luckii Admin.

Bug #1: issue1

- Author: luckii Admin
- Status: New
- Priority: Normal
- Assignee: luckii Admin
- Category:
- Target version:

이슈 1번 테스트

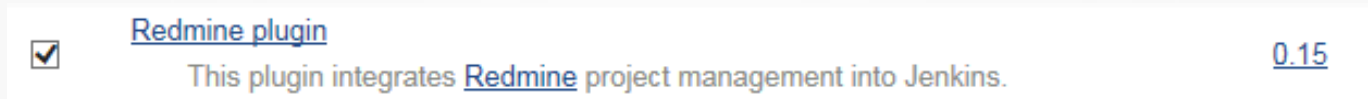
You have received this notification because you have either subscribed to it, or are involved in it.
To change your notification preferences, please click here: <http://hostname/my/account>

PART 4 Make up for CTIP

Jenkins : Redmine plugin

1. Redmine plugin 설치

Jenkins setting → plugin manager → 설치가능 → redmine plugin 체크 및 설치



2. Redmine 주소 설정

Jenkins setting → Redmine → 이름, url 및 version 입력

Redmine

Redmine websites	Name	Base url	Version number	
	T4_Redmine	http://192.168.204.1:3000/redmine	3.0.1	?

PART 4 Make up for CTIP

Jenkins : Redmine plugin

3. 프로젝트와 Redmine 연결

프로젝트 → 구성 → assign redmine project 체크 → 등록된 redmine 선택
→ redmine에 등록 되어 있는 프로젝트의 이름 입력

The screenshot shows the Jenkins configuration page for 'Project1' under the '환경설정' (Settings) tab. The left sidebar contains navigation options like '대시보드로 돌아가기', '상태', '변경사항', '작업공간', 'Build Now', 'Project 삭제', '구성', and 'Subversion Polling Log'. Below the sidebar is a 'Build History' table with columns for build number and timestamp.

The main configuration area includes fields for '이름' (Name) set to 'Project1' and '설명' (Description). A red box highlights the 'Assign Redmine project' section, which includes:

- Assign Redmine project
- Redmine website: T4_Redmine (http://192.168.204.1:3000/)
- Redmine project name: T4_project

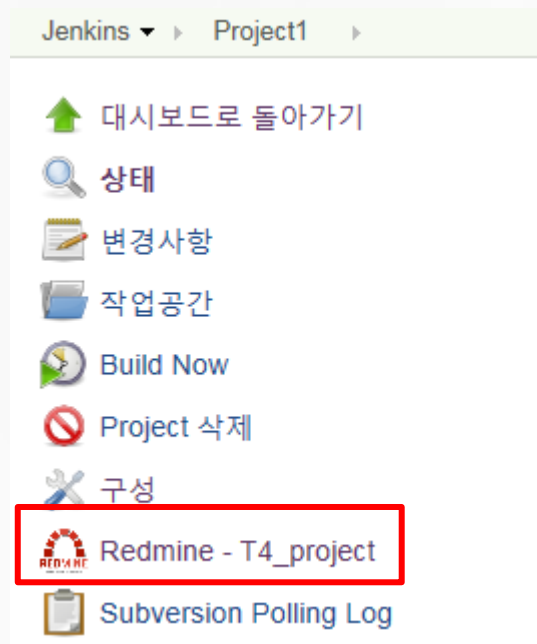
Other options include '오래된 빌드 삭제', '이 빌드는 매개변수가 있습니다', '빌드 안함', and '필요한 경우 concurrent 빌드 실행'. A '고급 프로젝트 옵션' (Advanced project options) section is also visible at the bottom.

PART 4 Make up for CTIP

Jenkins : Redmine plugin

3. 프로젝트와 Redmine 연결

프로젝트 메뉴에 redmine - T4 생성 → 클릭하면 단순히 redmine 링크로 넘어감.
 사실 상 연동되는 부분은 없다고 판단! → 메일 설정으로 빌드 이슈 관리 결정



빌드 실패 시 jenkins 관리자에게 메일 발송
 → 메일 확인 후 redmine에 이슈 등록
 → 지정된 담당자에게 메일 발송

PART 5

Summary

PART 5 Summary

Total Environment

	이름	버전
운영체제	Windows	7 (64bit)
JAVA		JDK 1.8.0_40
개발 도구	Eclipse	4.4.2 (Luna)
유닛 테스트 도구	Junit	4.0
빌드 도구	Ant	1.9.2
빌드 서버	Jenkins	1.609
버전 관리	Visual SVN Server	3.3.1
이슈 트래커	Redmine	3.0.1
정적 분석 도구	Checkstyle	6.5
	PMD	5.3.0
	JDepend	2.9.1
	Findbugs	3.0.1
시스템 테스트 도구	TestLink	1.9.13

✧ References

Downloads

- findbugs : <http://sourceforge.net/projects/findbugs/files/findbugs/3.0.1/findbugs-3.0.1.zip/download>
- pmd plugin : <http://sourceforge.net/projects/pmd/files/pmd-edipse/update-site/>
- <http://sourceforge.net/projects/pmd/files/pmd/5.3.0/pmd-bin-5.3.0.zip/download>
- testlink : <http://sourceforge.net/projects/testlink/files/TestLink%201.9/TestLink%201.9.13/testlink-1.9.13.targz/download>
- jdepend edipse plugin : <http://andrei.gmxhome.de/edipse/>

References

- <http://skyul.tistory.com/336> // find bugs
- <http://science.slc.edu/~jmarshall/courses/2002/fall/cs30/Lectures/week08/Computation.html>
//turing machine
- <http://blog.daum.net/darkandy/129> //jdepend
- <http://kitle.tistory.com/51> //testlink 설치
- <http://www.treselle.com/blog/static-code-analysis-jenkins/> // jenkins 정적 분석 툴 연동

THANK YOU
ANY QUESTIONS?