Software Verification

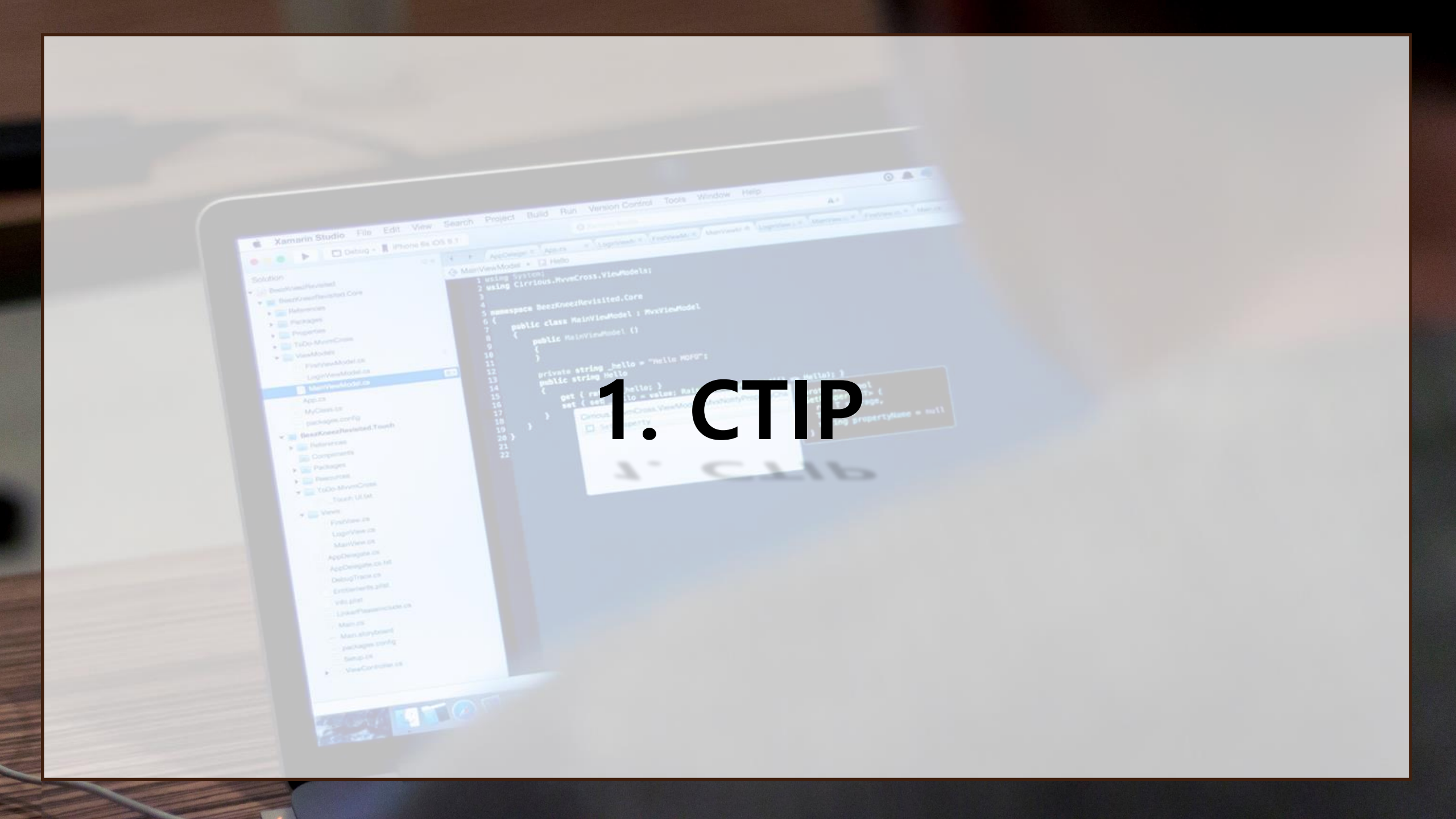
- Final -

Team 3

201011320 김용현
201111360 손준익
201111384 정국빈

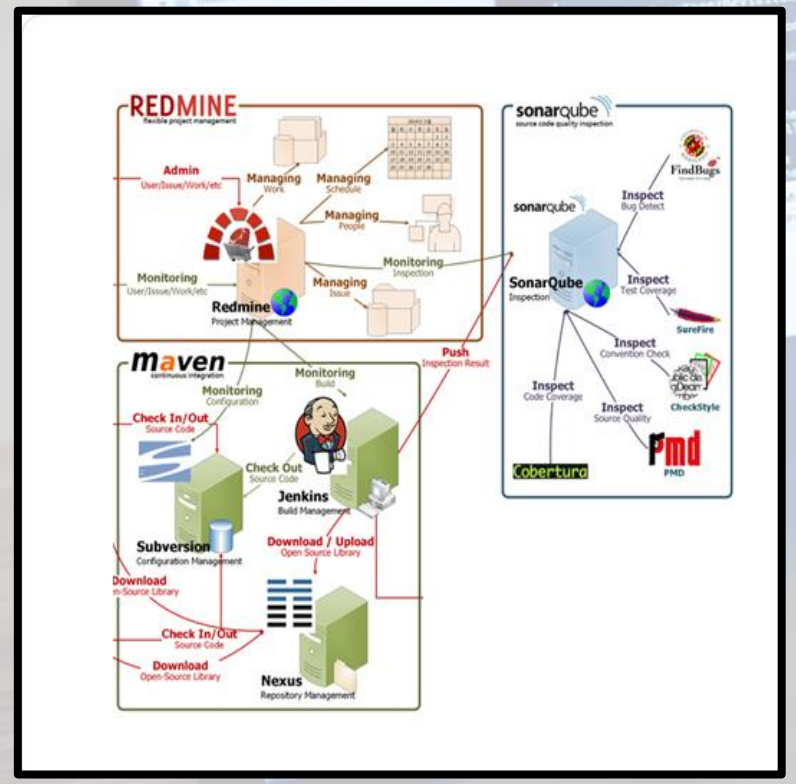
Index

1. CTIP
2. System Testing
3. Static Analysis
4. Improvement of S.V
5. Cooperation with S.M.A Team

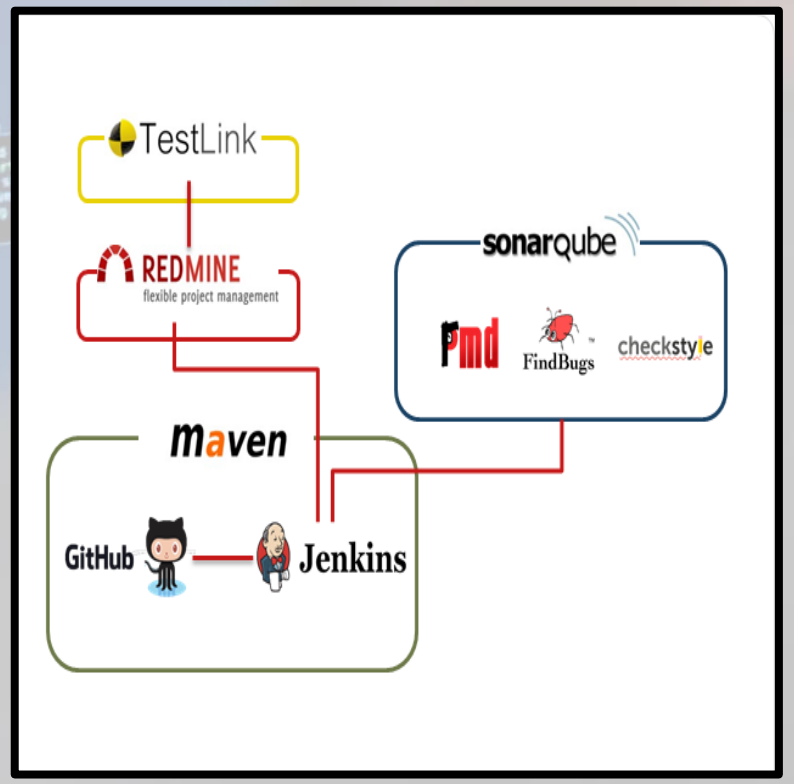

1. CTIP

* CTIP

<E-Government Framework>

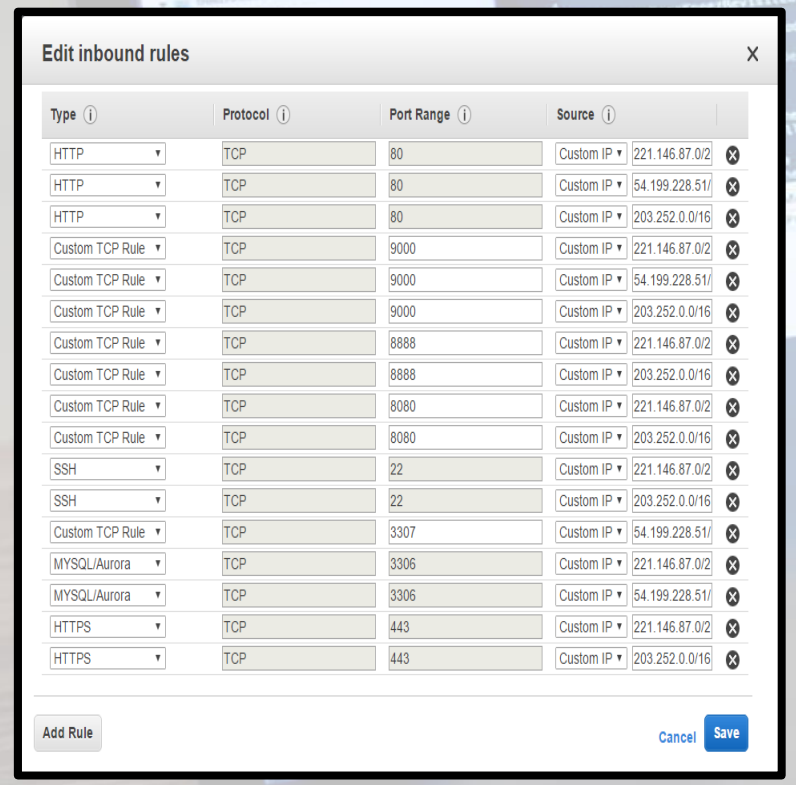


<Our CTIP>



* CTIP - Trouble Shooting

<AWS – security setting>



Edit inbound rules

Type	Protocol	Port Range	Source	
HTTP	TCP	80	Custom IP	221.146.87.0/2
HTTP	TCP	80	Custom IP	54.199.228.51/
HTTP	TCP	80	Custom IP	203.252.0.0/16
Custom TCP Rule	TCP	9000	Custom IP	221.146.87.0/2
Custom TCP Rule	TCP	9000	Custom IP	54.199.228.51/
Custom TCP Rule	TCP	9000	Custom IP	203.252.0.0/16
Custom TCP Rule	TCP	8888	Custom IP	221.146.87.0/2
Custom TCP Rule	TCP	8888	Custom IP	203.252.0.0/16
Custom TCP Rule	TCP	8080	Custom IP	221.146.87.0/2
Custom TCP Rule	TCP	8080	Custom IP	203.252.0.0/16
SSH	TCP	22	Custom IP	221.146.87.0/2
SSH	TCP	22	Custom IP	203.252.0.0/16
Custom TCP Rule	TCP	3307	Custom IP	54.199.228.51/
MYSQL/Aurora	TCP	3306	Custom IP	221.146.87.0/2
MYSQL/Aurora	TCP	3306	Custom IP	54.199.228.51/
HTTPS	TCP	443	Custom IP	221.146.87.0/2
HTTPS	TCP	443	Custom IP	203.252.0.0/16

Add Rule

Cancel Save

<Maven – pom.xml>

pom - 메모장

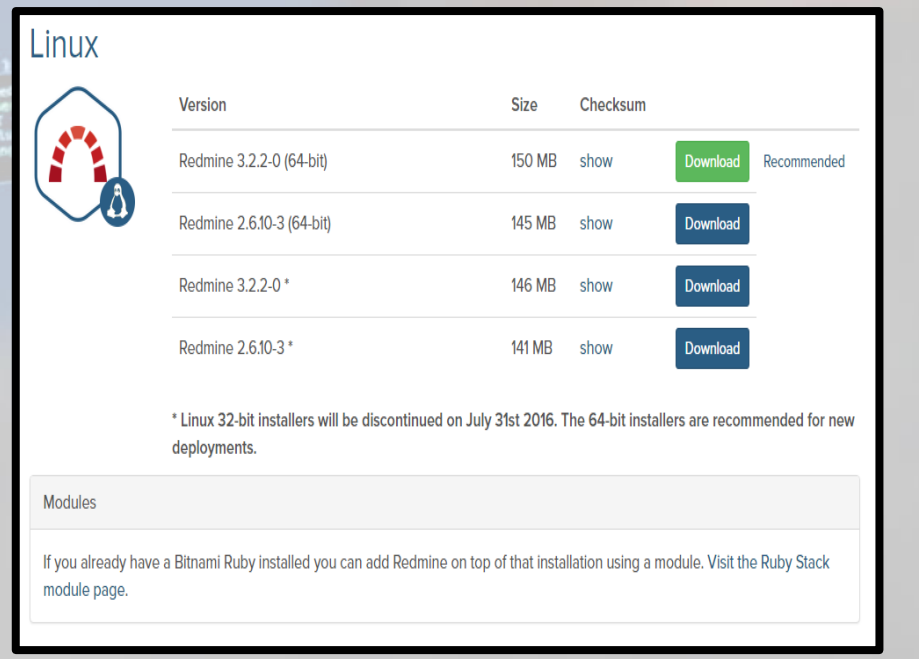
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

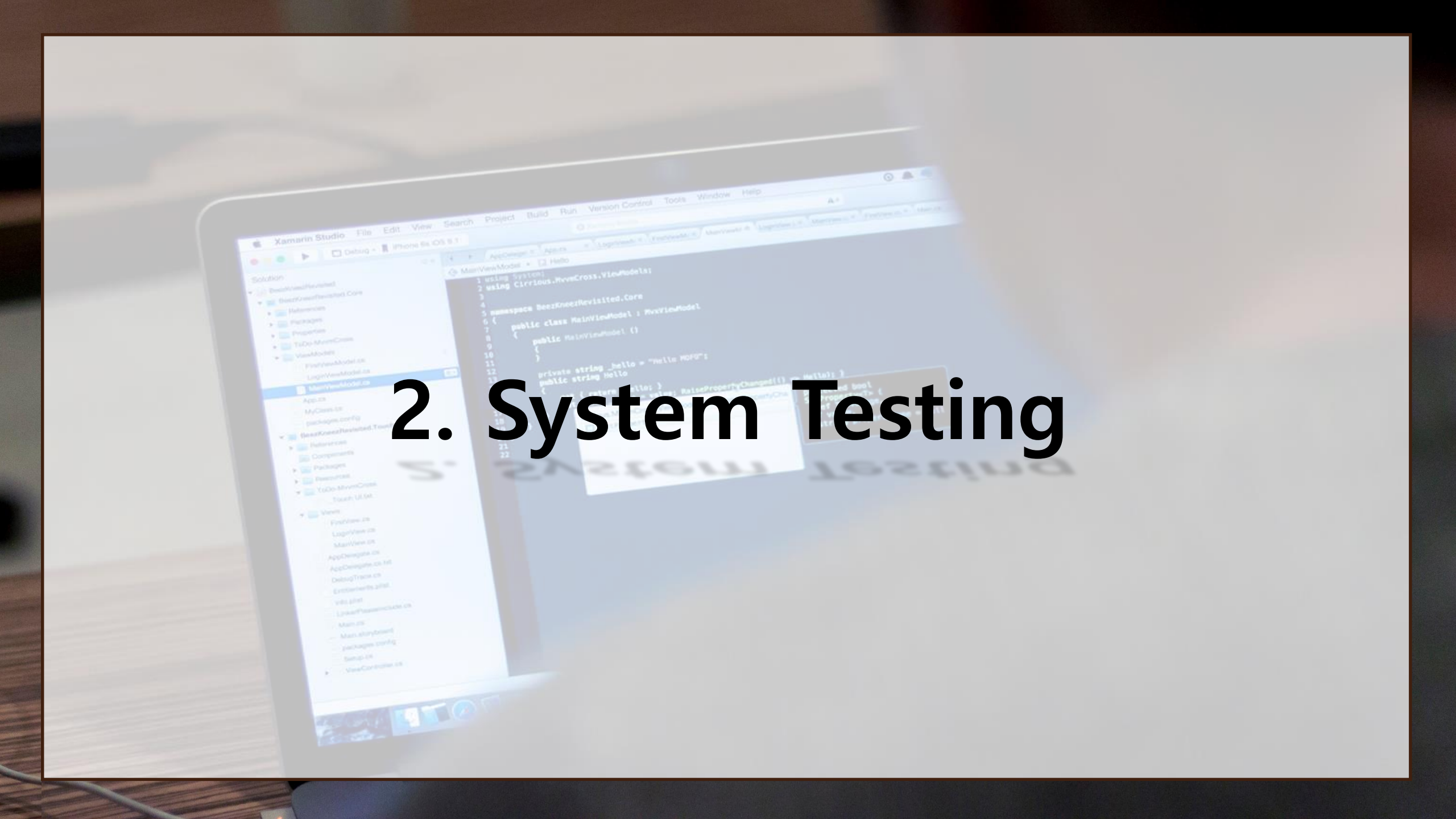
```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>src</groupId>
  <artifactId>test</artifactId>
  <version>0.0.2-SNAPSHOT</version>
  <packaging>jar</packaging>

  <name>test</name>
  <url>http://maven.apache.org</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>

  <scm>
  <connection>scm:git:https://github.com/aksjm2/2016sv.git</connection>
  </scm>

  <build>
  <plugins>
  <plugin>
      <groupId>org.apache.maven.plugins</groupId>
      <artifactId>maven-jar-plugin</artifactId>
      <version>2.4</version>
      <configuration>
        <archive>
          <manifest>
            <addClasspath>true</addClasspath>
            <classpathPrefix>classes/</classpathPrefix>
            <mainClass>src.test.MainSystem</mainClass>
          </manifest>
        </archive>
      </configuration>
    </plugin>
<plugin>
    <groupId>org.apache.maven.plugins</groupId>
```

* CTIP - Trouble Shooting

<REDMINE, TestLink – install problem>

<Bitnami Package>





2. System Testing

* System Testing

<Category Partition Test>

Category	Values	Number
Open File 확장자	.c	1000
	.etc	1001
Open File 수	0	1100
	1	1101
	n	1102
파일 이름 중복	이름이 중복되는 파일이 존재할 때	1200
	이름이 중복되는 파일이 존재하지 않을 때	1201
Select File 수	0	1300
	1	1301
	n	1302
Delete	Delete를 수행한다.	1400
Clear	파일 목록에 파일이 존재할 때	1401
	파일 목록에 파일이 존재하지 않을 때	1500
Compare	Compare를 수행한다.	1600

- We took a lot of effort to put a hidden use case in category test
- It seems to divide the categories well.
- wrongly dividing category may cause some problems.
- It was difficult to reduce the number of cases by applying constraints.

* System Testing

<Pairwise Test>

PictMaster v5.7.3 December 10, 2013

Item number		Item name		Date	
Sub-item numbe		Sub-item name		Creator	

Build Edit Settings

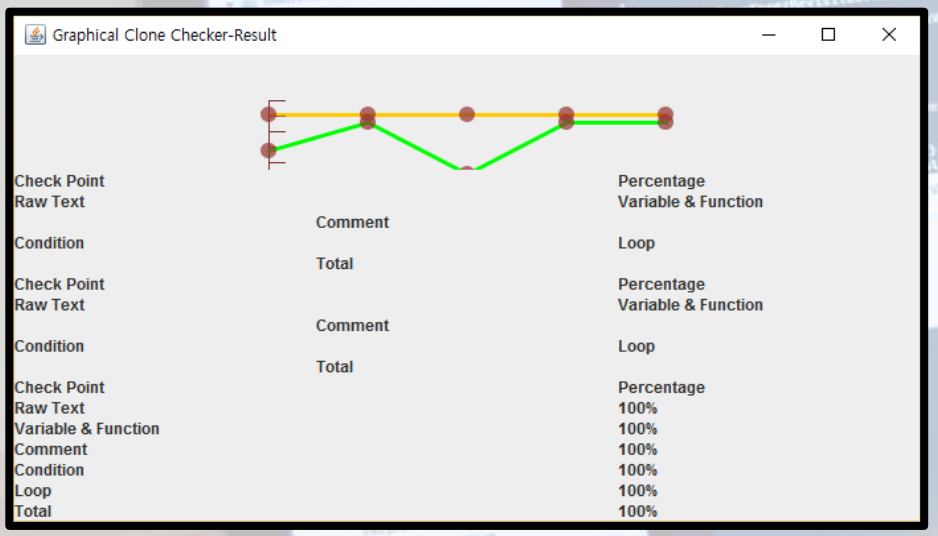
© IWATSU System & Software Co., Ltd. Licensed under the Open Software License version 2.1

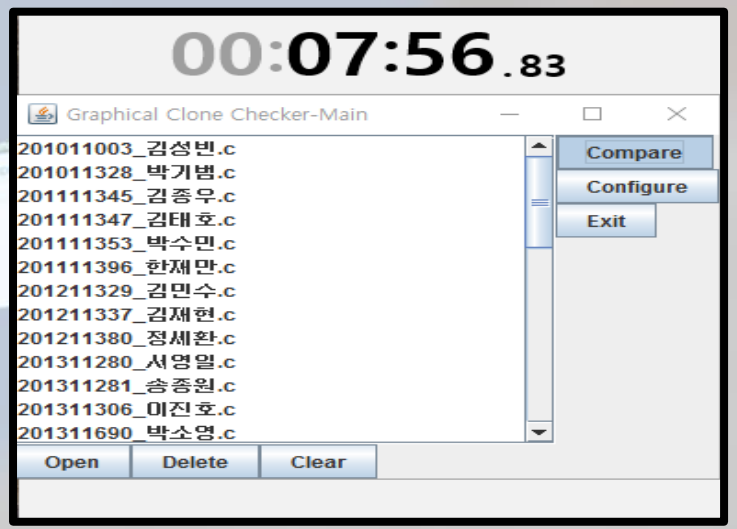
Parameters	value hierarchy
open file extension	c, etc
open file number	0, 1, n
filename samesame	sse, ssne
select file number	0,1,n
clear	fle, flne
delete button	delete execute
compare button	compare execute
configure	lessthan0, 0, 50, 100, 100over, malformed
select pair number	0, 1, n
display pair number	n

- It was difficult to use pict tool because it is made of Excel macro

* System Testing

<Brute Force Test>

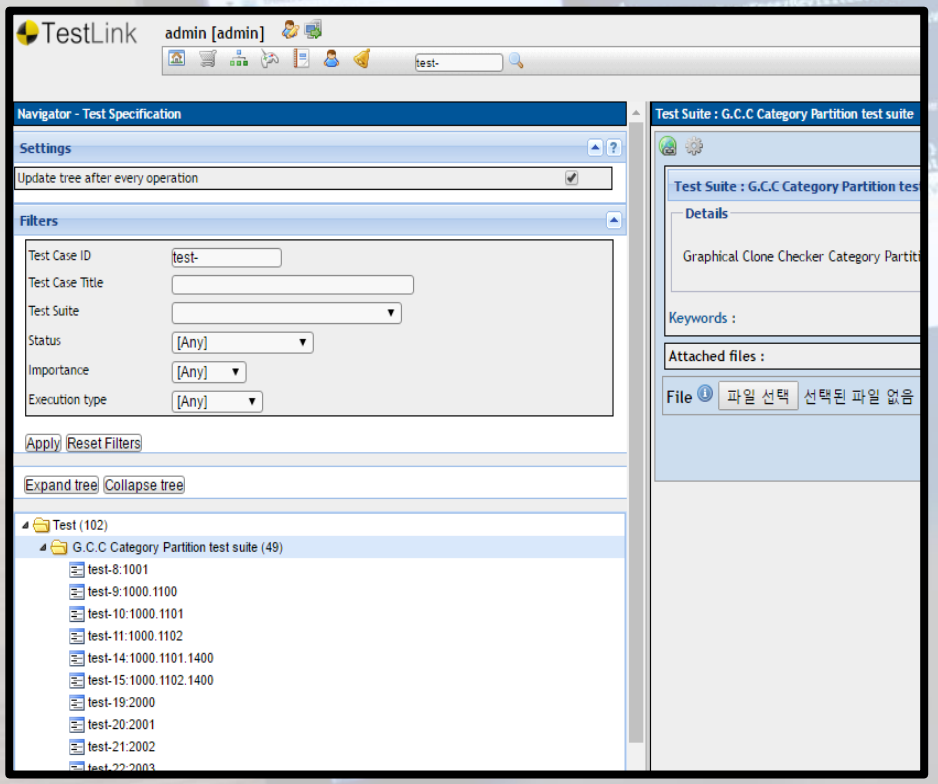




- It is not possible to test non functional requirements
  using category partition and Pairwise test so we use brute force test

* System Testing

<Testlink>

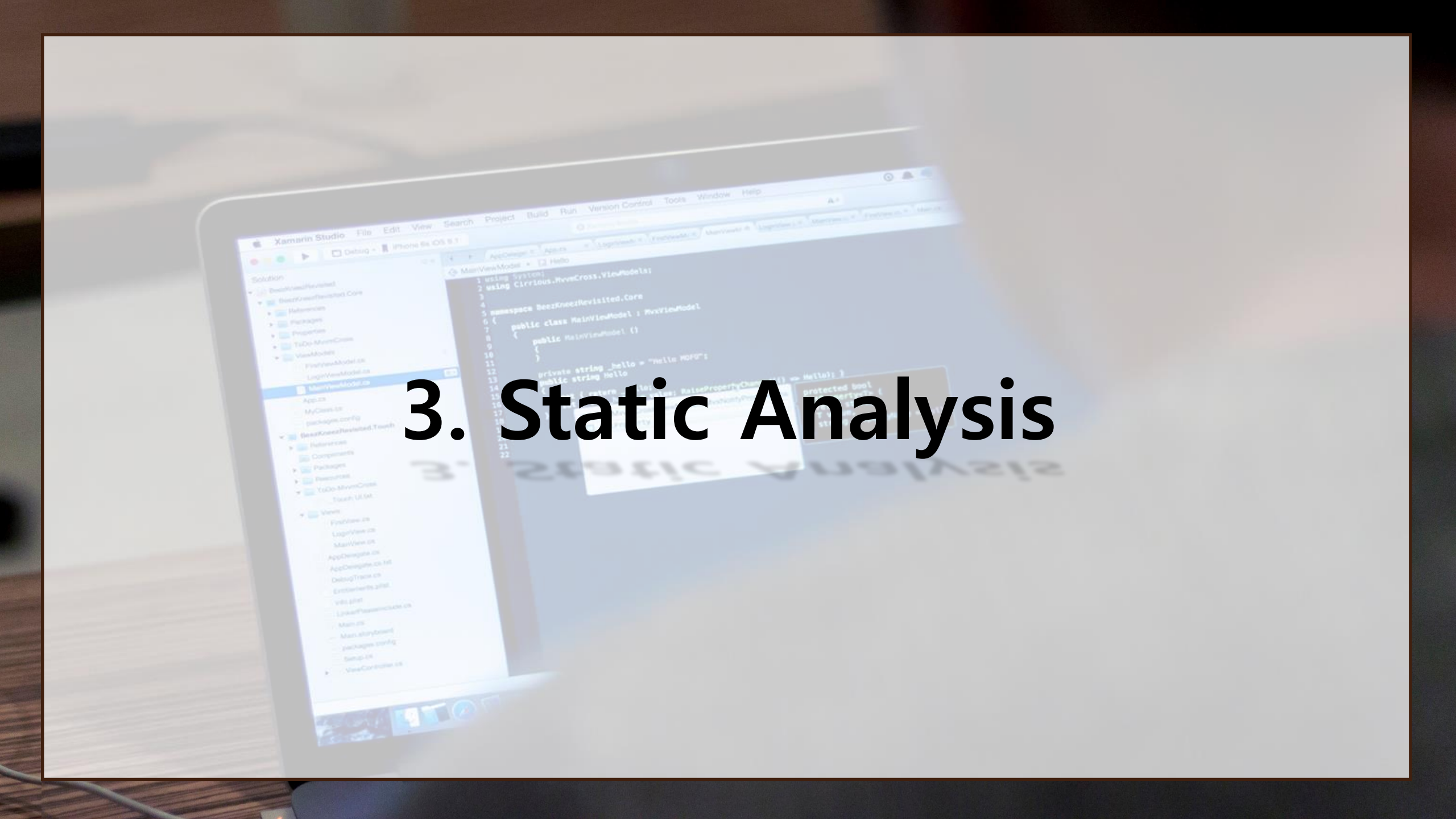


- It is useful for test case management

- Because we can link test case & requirement, It is easy to know each test case is linked to what requirement

- Testlink provide just link to redmine so we cannot find advantage

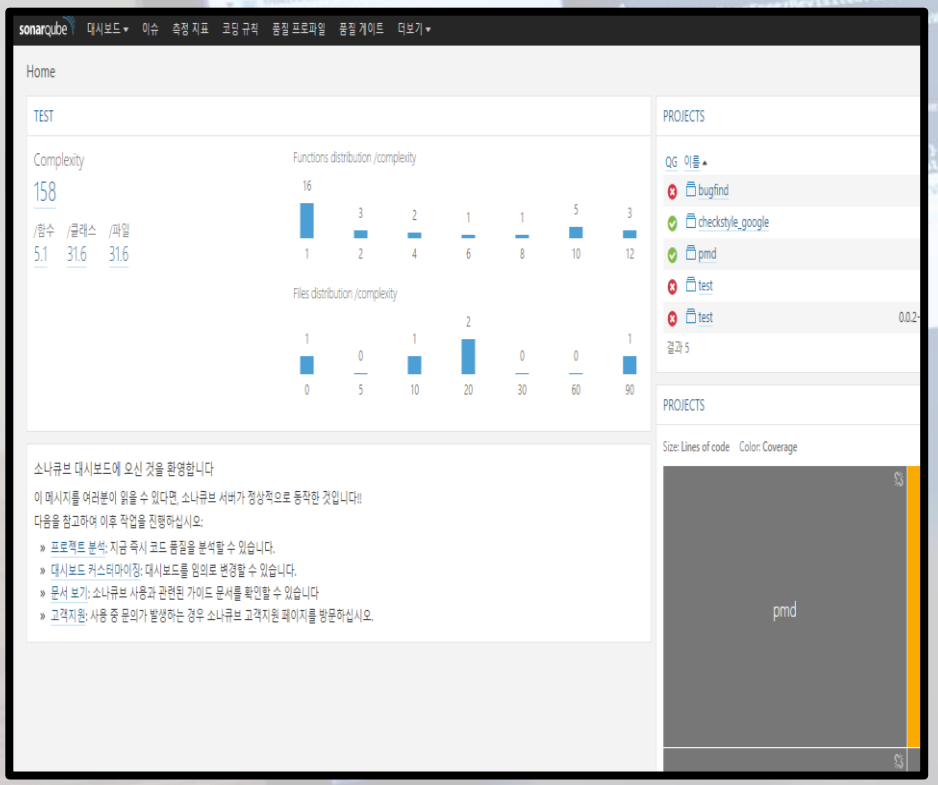
- U.I and U.X is bad



3. Static Analysis

* Static Analysis

<SonarQube>



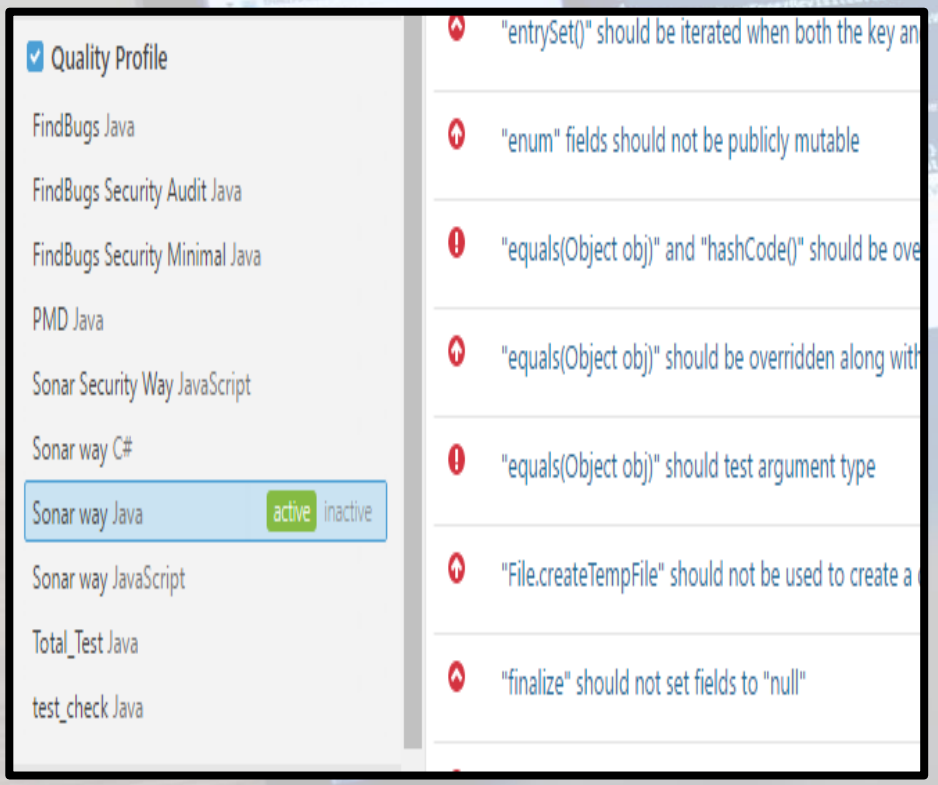
- It was easy to use because SonarQube provide korean language pack

- It was easy to apply rules of checkstyle, pmd, findbug

- U.I. and U.X. is good

* Static Analysis

<CheckStyle, PMD, FindBug>



- when we are coding,
  we need to consideration about rules.

- Because we didn't know detail of rules,
  we apply many rules(PMD, FindBug)

4. Improvement of S.V

* Improvement of S.V

<Improvement of S.V>

- Would be useful if the school provide a server.
- Would be helpful if we practice category partition method at practice time.
- Would be better to check CTIP environment in presentation.
- English presentation is very hard so every presentation must be in English.

forever . . . . :-)

5. Cooperation with S.M.A Team

* Cooperation with S.M.A. team

<Cooperation with S.M.A. team>

- After first System Test, S.M.A team's response is poor.

- It is sad that we didn't often meet S.M.A. team

- Last year, we were S.M.A team and cooperate S.V team.
  this year we knew S.V teams difficulty because we experience S.V.

Thx

Thank You !