

Final Presentation

System Test, Static Analysis Test
SW V&V, CTIP, ... (ANT, SVN, Mantis)

200611490 오 준
200913988 주영진
201011318 김슬기
201011358 임국현



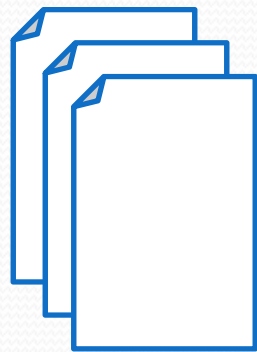
Contents

- Category-partition & Pairwise testing
- Clover
- Sonar Analysis
- Ant & Cruise Control
- SVN
- Mantis

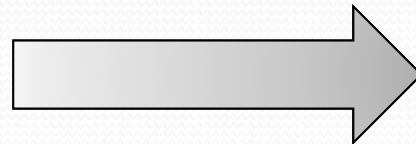


Category-partition & Pairwise Testing

Category-partition & Pairwise Testing



OSP Doc_team2



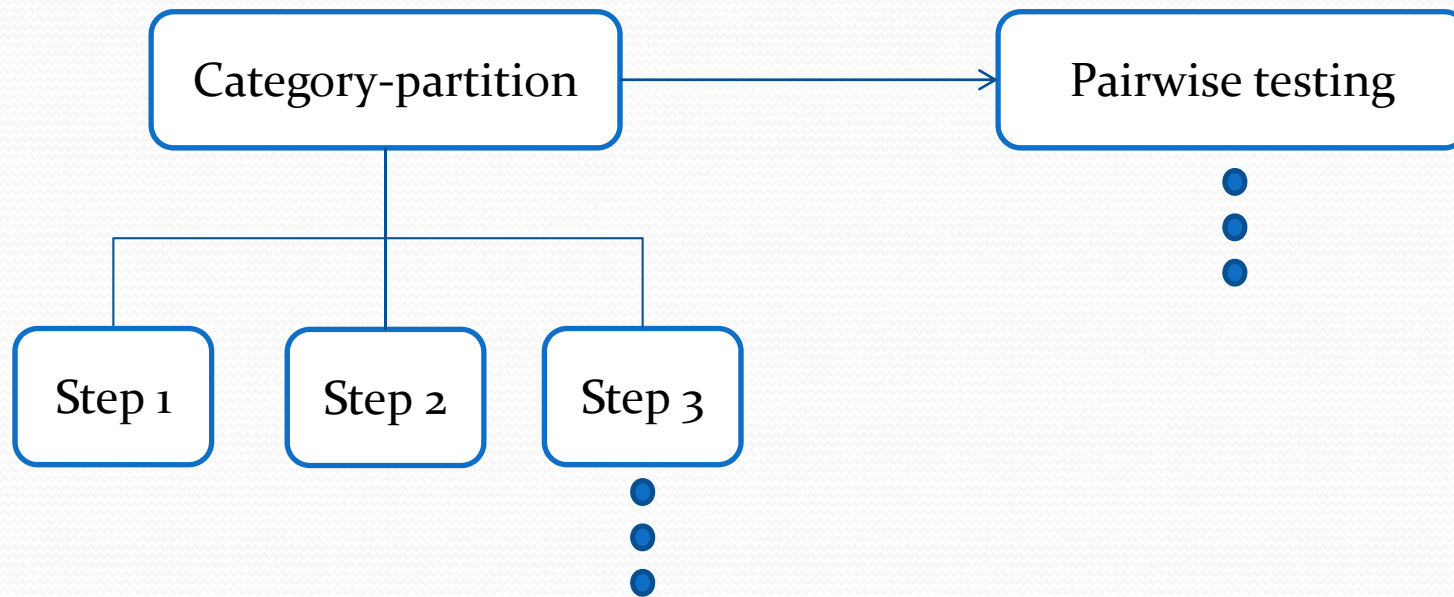
Category-partition & Pairwise Testing

- Testable features



Category-partition & Pairwise Testing

- Step



Category-partition & Pairwise Testing

- Step 1 : Identify Independently Testable Features & Parameter Characteristics

Testable Features	Input Parameters
파일 불러오기	파일 이름 및 경로
	파일 타입
	이미지 파일의 Width
	이미지 파일의 Height
	컷 사이즈

Category-partition & Pairwise Testing

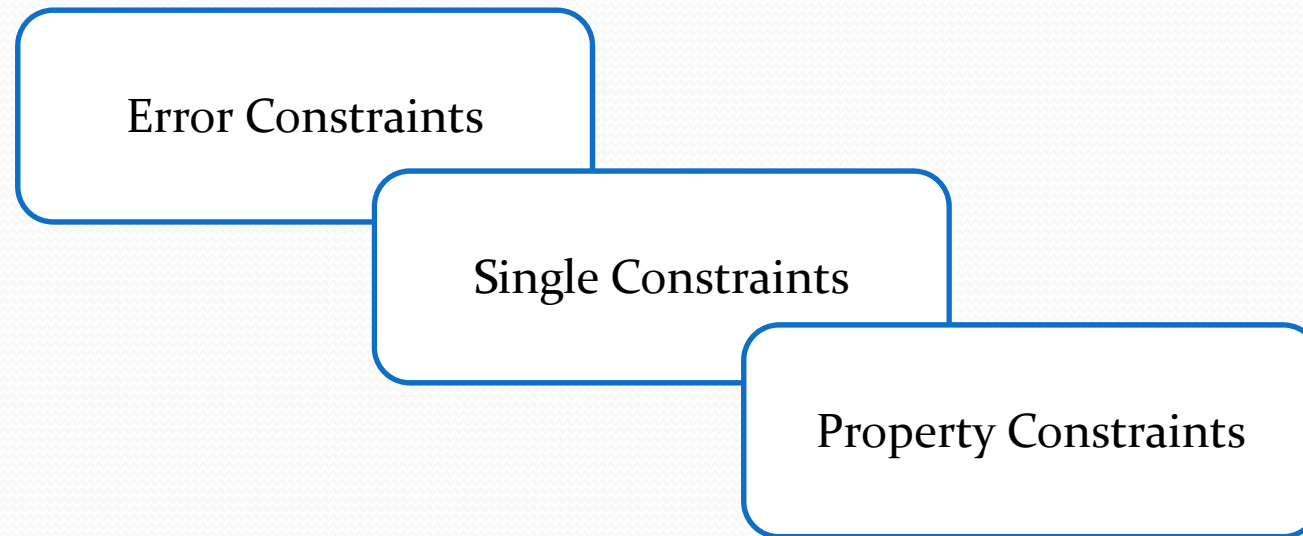
- Step 2 : Identify Representative Values

Input Parameters	Test Values
파일 이름 및 경로	empty
	존재하는 파일 이름 및 경로
	존재하지 않는 파일 이름 및 경로
파일 타입	bmp
	png
	others
이미지 파일의 Width	0
	100
	others



Category-partition & Pairwise Testing

- Step 3 : Generate Test Case Specifications



Category-partition & Pairwise Testing

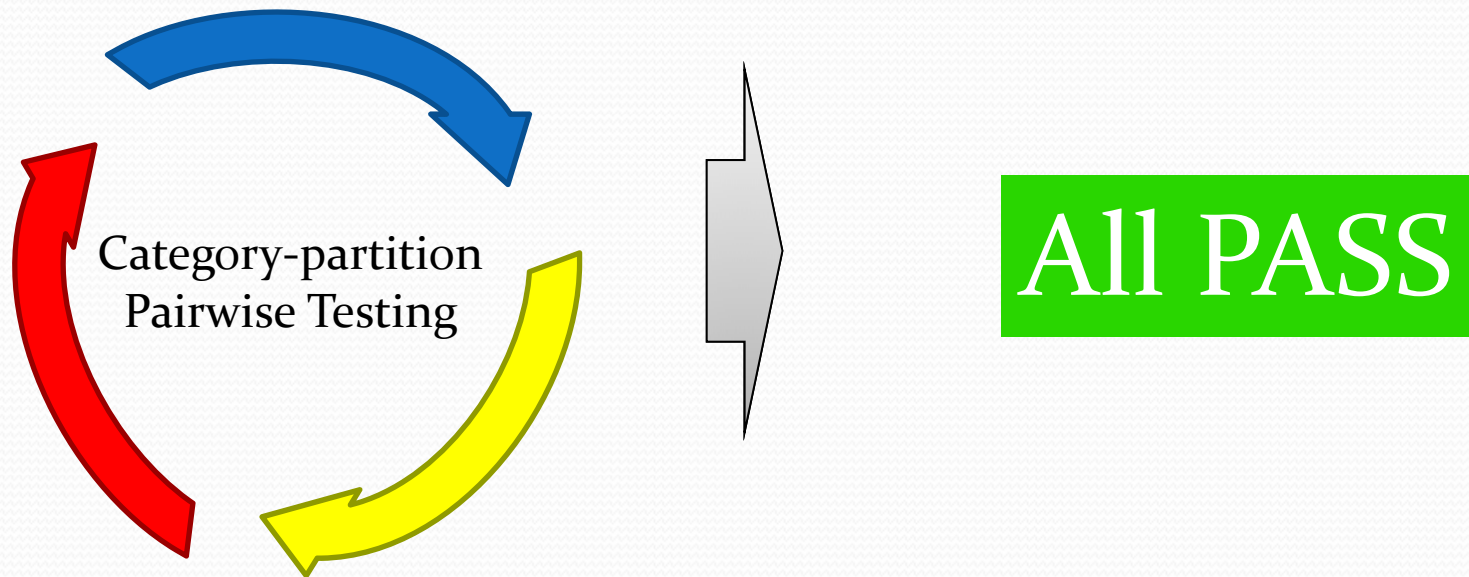
- Pairwise Testing

case	image_width	image_height	total_cut_size	pairings	pass/fail
1	100	100	1	3	pass
2	100	others	others	3	pass
3	others				
4	others				

var1	var2	value1	value2	appearances	cases
image_width	image_height	100	100	1	1
image_width	image_height	100	others	1	2
image_width	image_height	others	100	1	3
image_width	image_height	others	others	1	4
image_width	total_cut_size	100	1	1	1
image_width	total_cut_size	100	others	1	2
image_width	total_cut_size	others	1	1	4
image_width	total_cut_size	others	others	1	3
image_height	total_cut_size	100	1	1	1
image_height	total_cut_size	100	others	1	3
image_height	total_cut_size	others	1	1	4
image_height	total_cut_size	others	others	1	2

Category-partition & Pairwise Testing

- Result





Clover

Clover



Clover

- JUnit test case













- Test Coverage

































Clover

Elem	Cov%	Av Me Cpx	Cpx
Painter	 66.7%	1.7	179.0
(default package)	 66.7%	1.7	179.0
Area.java	 58.8%	1.0	4.0
AreaTool.java	 57.1%	1.4	14.0
Brush.java	 100.0%	1.0	1.0
Cut.java	 85.7%	1.0	3.0
DrawingTool.java	 100.0%	1.0	1.0
Interface.java	 58.7%	1.6	120.0
InterfaceTest.java	 100.0%	2.4	22.0
Project.java	 47.1%	3.5	14.0

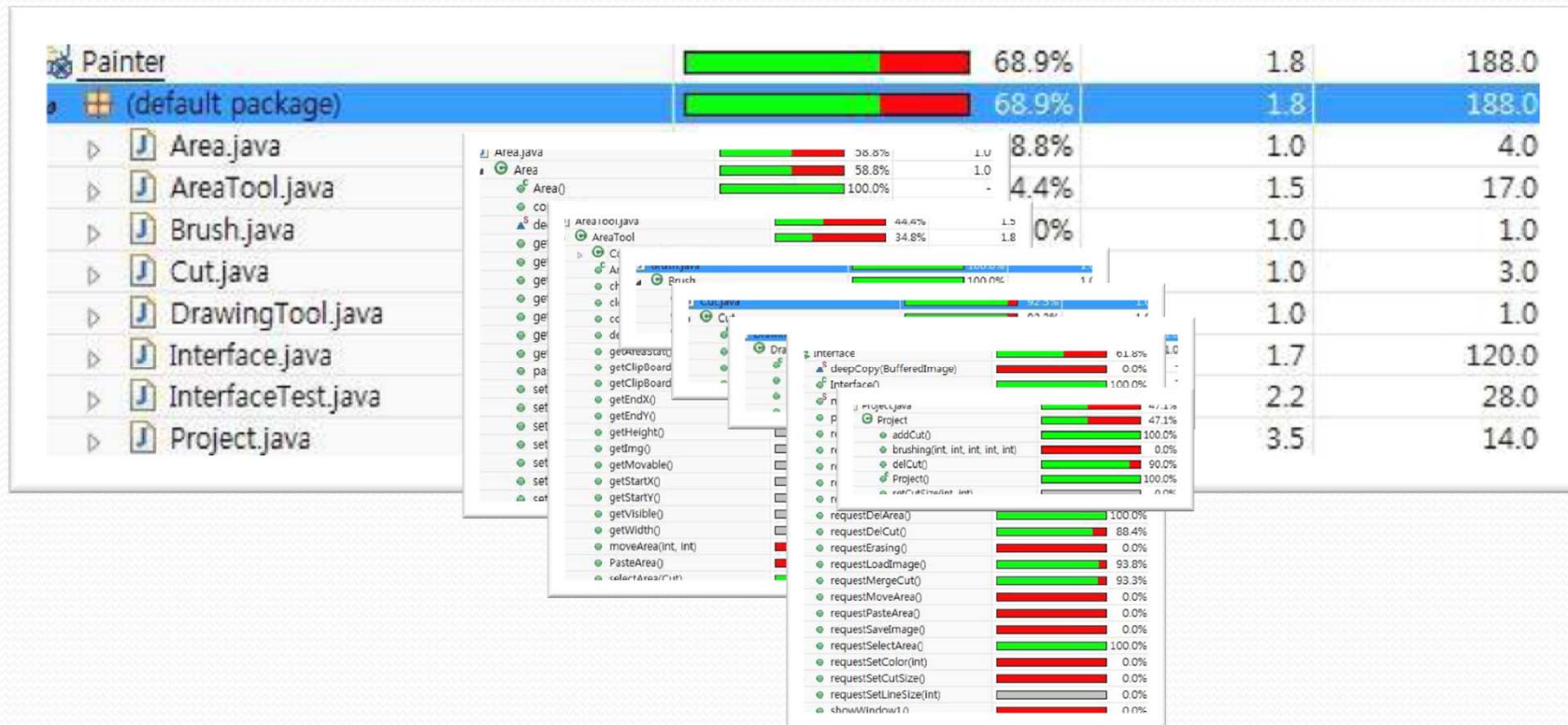
Painter	 68.9%	1.8	188.0
(default package)	 68.9%	1.8	188.0
Area.java	 58.8%	1.0	4.0
AreaTool.java	 44.4%	1.5	17.0
Brush.java	 100.0%	1.0	1.0
Cut.java	 92.3%	1.0	3.0
DrawingTool.java	 100.0%	1.0	1.0
Interface.java	 61.8%	1.7	120.0
InterfaceTest.java	 93.7%	2.2	28.0
Project.java	 47.1%	3.5	14.0



Clover

AreaTool.java		44.4%	1.5
AreaTool		34.8%	1.8
CopyImagetoClipboard		100.0%	1.0
AreaTool()		100.0%	-
checkAreaOn(int, int)		0.0%	-
clearArea()		100.0%	-
copyArea(Cut, int, int)		100.0%	-
delArea(Cut)		0.0%	-
getAreaStat()		0.0%	-
getClipboard(Area)		0.0%	-
getClipboard()		0.0%	-
getEndX()		0.0%	-
getEndY()		0.0%	-
getHeight()		0.0%	-
getImg()		0.0%	-
getMovable()		0.0%	-
getStartX()		0.0%	-
getStartY()		0.0%	-
getVisible()		0.0%	-
getWidth()		0.0%	-
moveArea(int, int)		0.0%	-
PasteArea()		0.0%	-
selectArea(Cut)		100.0%	-
Project.java		47.1%	
Project		47.1%	
addCut()		100.0%	
brushing(int, int, int, int, int)		0.0%	
delCut()		90.0%	
Project()		100.0%	
setCutSize(int, int)		0.0%	

Clover





Static Analysis

Static Analysis

- Sonar result (Static Analysis)

Lines of code 1,676 2,024 lines 996 statements 8 files	Classes 10 1 packages 84 methods 14 accessors	Violations 407 Rules compliance 57.3%	<table><tr><td></td><td>Blocker</td><td>0</td><td></td></tr><tr><td></td><td>Critical</td><td>0</td><td></td></tr><tr><td></td><td>Major</td><td>156</td><td><div style="width: 100%;"></div></td></tr><tr><td></td><td>Minor</td><td>247</td><td><div style="width: 100%;"></div></td></tr><tr><td></td><td>Info</td><td>4</td><td></td></tr></table>		Blocker	0			Critical	0			Major	156	<div style="width: 100%;"></div>		Minor	247	<div style="width: 100%;"></div>		Info	4					
	Blocker	0																									
	Critical	0																									
	Major	156	<div style="width: 100%;"></div>																								
	Minor	247	<div style="width: 100%;"></div>																								
	Info	4																									
Comments 6.7% 121 lines 0.6% docu. API 154 undocu. API	Duplications 8.9% 180 lines 10 blocks 3 files	Package tangle index 0.0% > 0 cycles	Dependencies to cut 0 between packages 0 between files																								
Complexity 3.2 /method 26.5 /class 33.1 /file Total: 265	<table border="1"><caption>Complexity Distribution</caption><thead><tr><th>Complexity Level</th><th>Methods</th><th>Files</th></tr></thead><tbody><tr><td>1</td><td>50</td><td>0</td></tr><tr><td>2</td><td>15</td><td>0</td></tr><tr><td>4</td><td>5</td><td>0</td></tr><tr><td>6</td><td>2</td><td>0</td></tr><tr><td>8</td><td>1</td><td>0</td></tr><tr><td>10</td><td>1</td><td>0</td></tr><tr><td>12</td><td>1</td><td>0</td></tr></tbody></table>	Complexity Level	Methods	Files	1	50	0	2	15	0	4	5	0	6	2	0	8	1	0	10	1	0	12	1	0	Unit tests coverage 0.0% 0.0% line coverage 0.0% branch coverage	Unit test success 0 tests
Complexity Level	Methods	Files																									
1	50	0																									
2	15	0																									
4	5	0																									
6	2	0																									
8	1	0																									
10	1	0																									
12	1	0																									

Static Analysis

- Violations

Hotspots

Reviews

Time Machine

TOOLS

Components

Violations Drilldown

Design

Documentation

Libraries

Clouds

Compare



Most violated rules Any severity [More](#)

▼ Magic Number	219	<div style="width: 100%;"></div>
▲ Local Variable Name	50	<div style="width: 50%;"></div>
▲ Visibility Modifier	35	<div style="width: 35%;"></div>
▼ Trailing Comment	27	<div style="width: 27%;"></div>
▲ Simplify Boolean Expression	14	<div style="width: 14%;"></div>

Most violated resources [More](#)

Interface	▲ 0	▲ 0	▲ 105	▼ 124	▼ 4
Project	▲ 0	▲ 0	▲ 16	▼ 64	▼ 0
InterfaceTest	▲ 0	▲ 0	▲ 13	▼ 42	▼ 0
AreaTool	▲ 0	▲ 0	▲ 11	▼ 4	▼ 0
Cut	▲ 0	▲ 0	▲ 7	▼ 7	▼ 0

Hotspots by Complexity [More](#)

Interface	145	<div style="width: 100%;"></div>
AreaTool	41	<div style="width: 28%;"></div>
Project	33	<div style="width: 23%;"></div>
InterfaceTest	23	<div style="width: 16%;"></div>
Area	12	<div style="width: 8%;"></div>

Hotspots by Uncovered lines [More](#)

Interface	766	<div style="width: 100%;"></div>
InterfaceTest	193	<div style="width: 25%;"></div>
AreaTool	125	<div style="width: 16%;"></div>
Project	115	<div style="width: 15%;"></div>
Area	82	<div style="width: 11%;"></div>

Hotspots by Duplicated lines [More](#)

InterfaceTest	97	<div style="width: 100%;"></div>
Project	48	<div style="width: 50%;"></div>
Interface	35	<div style="width: 35%;"></div>

Hotspots by Complexity /method [More](#)

Project	6.6	<div style="width: 100%;"></div>
Interface	6.3	<div style="width: 95%;"></div>
InterfaceTest	2.3	<div style="width: 35%;"></div>
Area	1.5	<div style="width: 23%;"></div>
AreaTool	1.5	<div style="width: 23%;"></div>

Static Analysis

- Violations List

All

Rule	Count
Local Variable Name	50
Visibility Modifier	3
Simplify Boolean Expression	14
Member Name	1
Avoid Commented-Out Lines Of Code	10
If Else Stmts Must Use Braces	9
Anon Inner Length	8
If Stmts Must Use Braces	4
System Println	3
Local Final Variable Name	2
Cyclomatic Complexity	2
Method Name	2
Hidden Field	1
Naming - Suspicious Constant Field Name	1
Access Method Count	1
Unused Private Field	1
Unused Private Method	1
Avoid Print Stack Trace	1
Sum	154



Class

The 'Class' section displays multiple overlapping tables, each representing a different class and its associated violation counts. The tables are arranged in a way that shows the distribution of violations across various classes, with some tables overlapping others. The visible tables include:

- Class: [unclear]**
 - Local Variable Name: 3
 - Member Name: 2
 - Method Name: 1
- Class: [unclear]**
 - Visibility Modifier: 1
 - Simplify Boolean Expression: 1
 - Avoid Commented-Out Lines Of Code: 1
 - If Else Stmts Must Use Braces: 1
 - Cyclomatic Complexity: 1
- Class: [unclear]**
 - Local Variable Name: 1
 - Visibility Modifier: 1
 - Member Name: 1
 - System Println: 1
 - Hidden Field: 1
 - Naming - Suspicious Constant Field Name: 1
 - Unused Private Field: 1
- Class: [unclear]**
 - Visibility Modifier: 1
 - Method Name: 1
- Class: DrawingTool**
 - Visibility Modifier: 1
- Class: Brush**
 - Visibility Modifier: 1

Static Analysis

- Violations List

All

Rule	Count
Local Variable Name	50
Visibility Modifier	3
Simplify Boolean Expression	14
Member Name	1
Avoid Commented-Out Lines Of Code	10
If Else Stmts Must Use Braces	9
Anon Inner Length	8
If Stmts Must Use Braces	4
System Println	3
Local Final Variable Name	2
Cyclomatic Complexity	2
Method Name	2
Hidden Field	2
Naming - Suspicious Constant Field Name	1
Access Method Count	1
Unused Private Field	1
Unused Private Method	1
Avoid Print Stack Trace	1
Sum	154

Kinds

The 'Kinds' section displays several overlapping screenshots of violation messages from a static analysis tool. A large blue arrow points from the 'All' table towards these screenshots. The screenshots show various error messages such as:

- Line Num: 46, Class: public int SelectedFoot, Name 'SelectedFoot' must match pattern '^([a-zA-Z0-9])\$'
- Line Num: 52, Class: public int EnabledColumn, Name 'EnabledColumn' must match pattern '^([a-zA-Z0-9])\$'
- Line Num: 59, Class: public JPanel CanvasPanel, Name 'CanvasPanel' must match pattern '^([a-zA-Z0-9])\$'
- Line Num: 61, Class: public JPanel CutListPanel
- Line Num: 211, Class: if (ButtonList.getSelected() == true) { Expression can be simplified.
- Line Num: 684, Class: final ToggleButton btnNewButton_2 = new ToggleButton(Name btnNewButton_2 must match pattern '^([a-zA-Z0-9])\$'
- Line Num: 1059, Class: if (CanvasList.isEmpty() == false) { Expression can be simplified.
- Line Num: 1061, Class: if (CanvasList.get().isVisible() == true) { Expression can be simplified.
- Line Num: 132, Class: Project if (cuts.get().cutstat == true) { Expression can be simplified.
- Line Num: 151, Class: Project if (cuts.get().cutstat == true) { Expression can be simplified.
- Line Num: 194, Class: InterfaceTest if (of.cj.cuts.get().cutstat == true) { Expression can be simplified.
- Line Num: 32, Class: Cut public void LoadCut(String path) throws IOException Name 'LoadCut' must match pattern '^([a-zA-Z0-9])\$'
- Line Num: 65, Class: public ButtonGroup LoadGroup = new ButtonGroup() Variable 'LoadGroup' must be private and have accessor methods.
- Line Num: 66, Class: public ButtonGroup ColorGroup = new ButtonGroup() Variable 'ColorGroup' must be private and have accessor methods.
- Line Num: 67, Class: public Toggle t1 Variable 't1' must be private and have accessor methods.
- Line Num: 68, Class: public Cursor brush Variable 'brush' must be private and have accessor methods.
- Line Num: 69, Class: public Cursor eraser Variable 'eraser' must be private and have accessor methods.
- Line Num: 697, Class: int i Variable 'i' must be private and have accessor methods.
- Line Num: 497, Class: int j Variable 'j' must be private and have accessor methods.
- Line Num: 73, Class: public List<Cut> cuts Variable 'cuts' must be private and have accessor methods.
- Line Num: 28, Class: public int bef_x Variable 'bef_x' must be private and have accessor methods.
- Line Num: 27, Class: public int bef_y Variable 'bef_y' must be private and have accessor methods.
- Line Num: 28, Class: public int track Variable 'track' must be private and have accessor methods.
- Line Num: 32, Class: ArrayList<ColorGroup> color Variable 'color' must be private and have accessor methods.
- Line Num: 25, Class: Cut
- Line Num: 27, Class: Cut
- Line Num: 28, Class: Cut
- Line Num: 28, Class: Cut
- Line Num: 36, Class: DrawingTool public void draw Variable 'draw' must be private and have accessor methods.
- Line Num: 37, Class: DrawingTool public Brush eraser Variable 'eraser' must be private and have accessor methods.
- Line Num: 38, Class: DrawingTool public int track Variable 'track' must be private and have accessor methods.

Static Analysis

- JArchitect

It's better



Static Analysis

```
import android.app.Activity;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;

public class StatusBarNotify extends Activity {
    private static final int HELLO_ID = 1;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

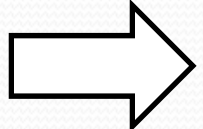
        // noti피케이션메니저에 대한 리퍼런스 얻기
        String ns = Context.NOTIFICATION_SERVICE;
        NotificationManager mNotificationManager =
            (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);

        // noti피케이션 인스턴스 만들기
        int icon = R.drawable.ic_launcher;
        CharSequence tickerText = "hello";
        long when = System.currentTimeMillis();
        Notification notification = new Notification(icon, tickerText, when);

        // noti피케이션 메시지와 인텐트를 정의
        PendingIntent contentIntent = PendingIntent.getActivity(this, 0,
            new Intent(this, StatusBarNotify.class),
            PendingIntent.FLAG_ACTIVITY_NEW_TASK);
        notification.setLatestEventInfo(this, "My notification",
            contentIntent);

        // noti피케이션메니저에게 noti피케이션 전달
        mNotificationManager.notify(HELLO_ID, notification);
    }
}
```

Bad code



```
import android.app.Activity;
import android.app.Notification;
import android.app.NotificationManager;
import android.app.PendingIntent;
import android.content.Context;
import android.content.Intent;
import android.os.Bundle;

public class StatusBarNotify extends Activity {
    private static final int HELLO_ID = 1;

    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        // noti피케이션메니저에 대한 리퍼런스 얻기
        String ns = Context.NOTIFICATION_SERVICE;
        NotificationManager mNotificationManager =
            (NotificationManager) getSystemService(Context.NOTIFICATION_SERVICE);

        // noti피케이션 인스턴스 만들기
        int icon = R.drawable.ic_launcher;
        CharSequence tickerText = "hello";
        long when = System.currentTimeMillis();
        Notification notification = new Notification(icon, tickerText, when);

        // noti피케이션 메시지와 인텐트를 정의
        PendingIntent contentIntent = PendingIntent.getActivity(this, 0,
            new Intent(this, StatusBarNotify.class),
            PendingIntent.FLAG_ACTIVITY_NEW_TASK);
        notification.setLatestEventInfo(this, "My notification",
            contentIntent);

        // noti피케이션메니저에게 noti피케이션 전달
        mNotificationManager.notify(HELLO_ID, notification);
    }
}
```

Good code



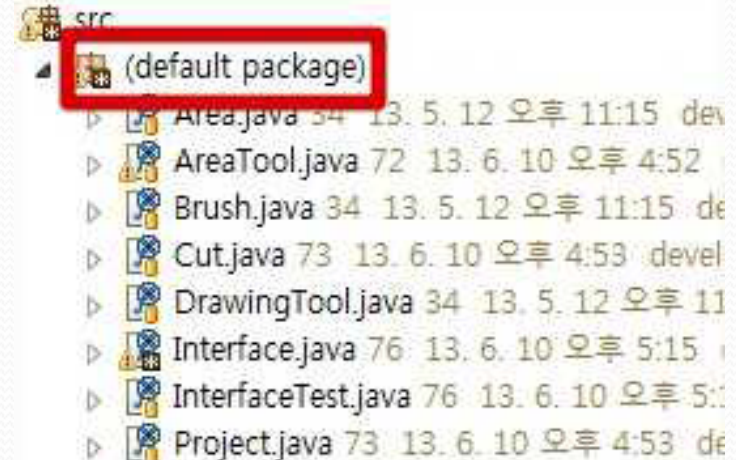
Ant & Cruise Control

Ant & Cruise Control

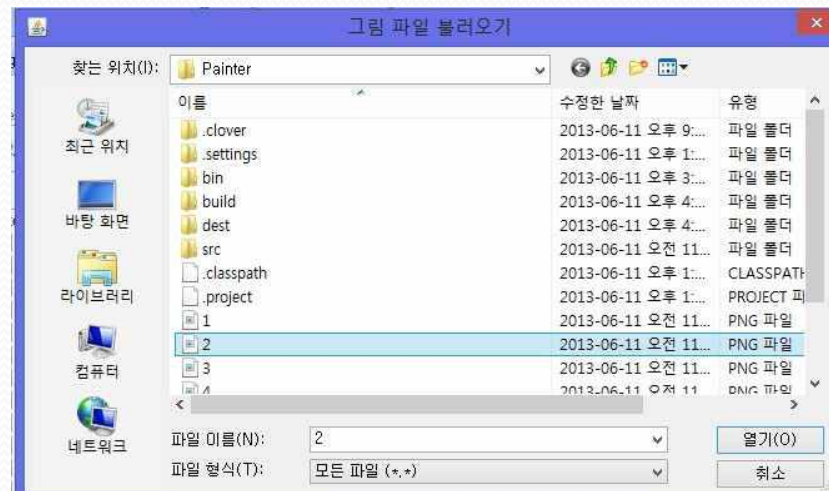
Case 1



Case 2



Case 3

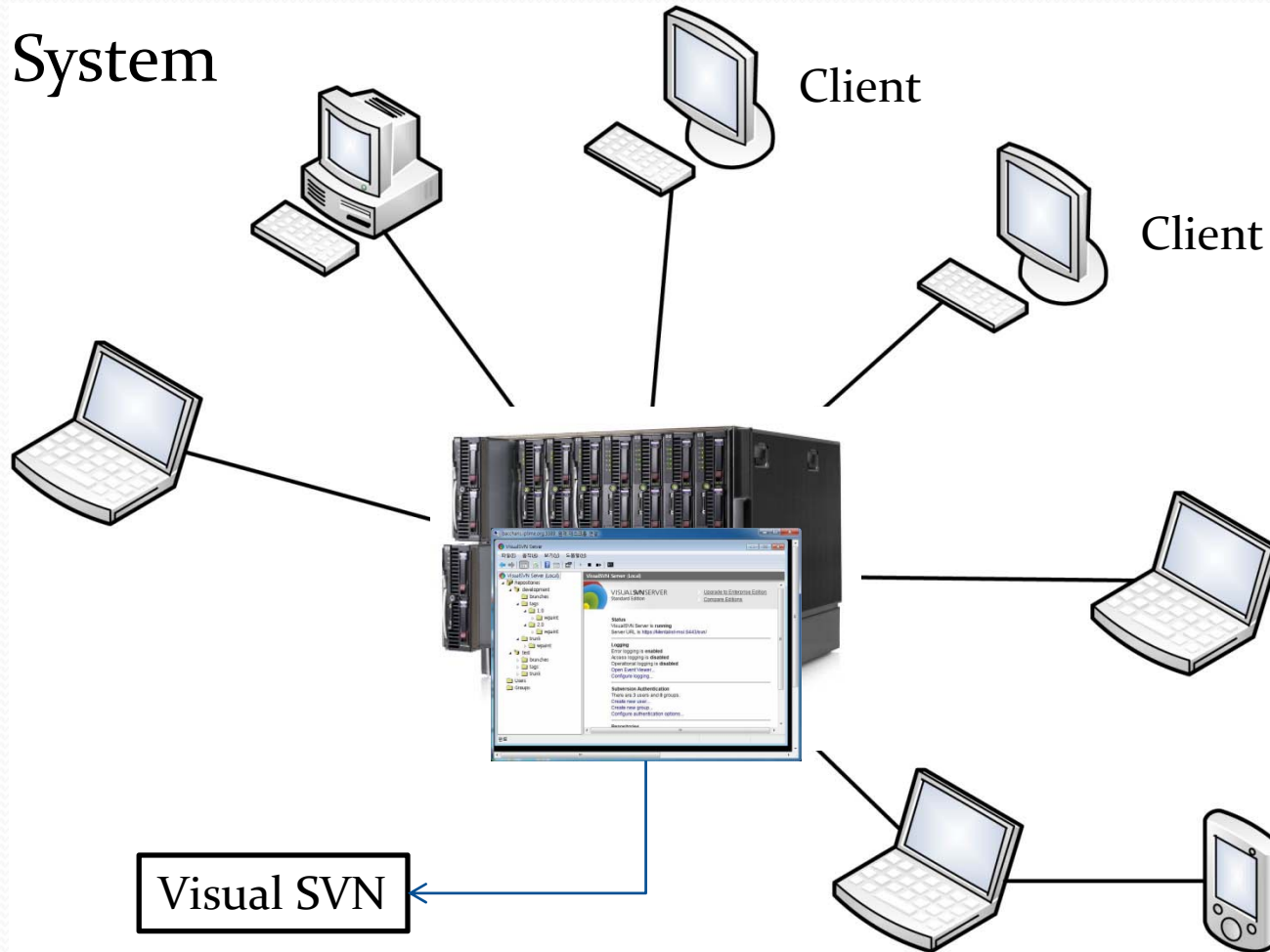




SVN

SVN

- SVN System



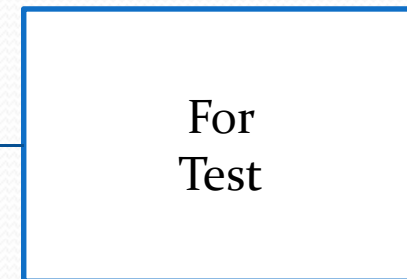
SVN

- Problem & solve

3학년



4학년



SVN





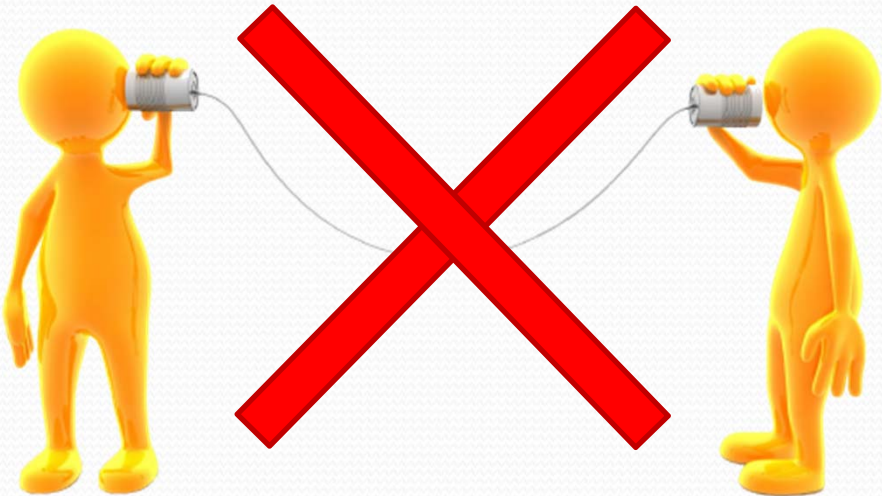
Mantis

Mantis

The screenshot shows the Mantis Bug Tracker interface. The top navigation bar includes links for '찾아보기', '내 페이지', '이슈 보기', '이슈 보고하기', '변경 기록', '종료됨', '휴먼', '관리', '계정 관리', and '로그아웃'. The main content area is divided into two columns. The left column displays a list of bugs with columns for ID, description, reporter, and date. The right column shows the details of a selected bug, including its title, description, and various fields like 'Assign To', 'Severity', and 'Priority'.

This screenshot displays a grid of bug lists from the Mantis Bug Tracker. The grid is organized into four main sections: '내가 보고함' (1-10/12), '내가 보고됨' (1-10/11), '최근에 변경됨' (1-10/32), and '내가 모니터링 중' (1-1/1). Each section contains a list of bugs with columns for ID, description, reporter, and date. The bugs are color-coded, with green indicating 'Resolved' and red indicating 'Unresolved'. The interface also shows the user's profile information at the top, including the name 'Administrator' and the email 'helixworld'.

Mantis



Mantis



읽음 5273

내게쓰기
수신확인

417

클립보... 156

142

265

메일 받는사람

검색결과 47건 **mantis**

답장 전체답장 전달 안읽음 *삭제 스펀신고 이동 추가기능 목록 <이전 다음 > 단축키

그리고 기타 수행화면

(0000014) minchulkim (보고가능) - 2013-06-10 15:58
<http://baccharis.iptime.org:8080/mantis/view.php?id=28#c14>

작동 방식이 이미 선택되어 있는 영역 삭제 -> 클립보드에 있는 이미지 복사 했는데

클립보드에 있는 이미지를 바로 복사하지 않고 클립보드의 이미지의 가로, 세로 높이가 둘다 0 이상일 경우에만 선택되어 있는 영역 삭제 -> 클립보드에 있는 이미지 복사를 하고

클립보드의 이미지의 가로, 세로 높이중 하나라도 0이면 동작하지 않도록 하였습니다.

이슈 역사
수정일 사용자 이름 필드
수정



Q & A



Thank you