

Presentation #5

# Static Analysis

2017 Software Modeling & Analysis

Team 3

한석호  
이진호  
나경수  
이원재

# Index



001/ 대상

002/ 사용 도구

003/ Team 1

004/ Team 2 (Client)

005/ Team 2 (Server)

## 1

## 대상

1	4, 3
2	4, 1
3	1, 2
4	2, 3
5	8, 7
6	8, 5
7	5, 6
8	6, 7

지정해 주신대로 1팀과 2팀의 코드를 분석

2팀은 Server와 Client를 나누어서 분석

## 2 사용 도구

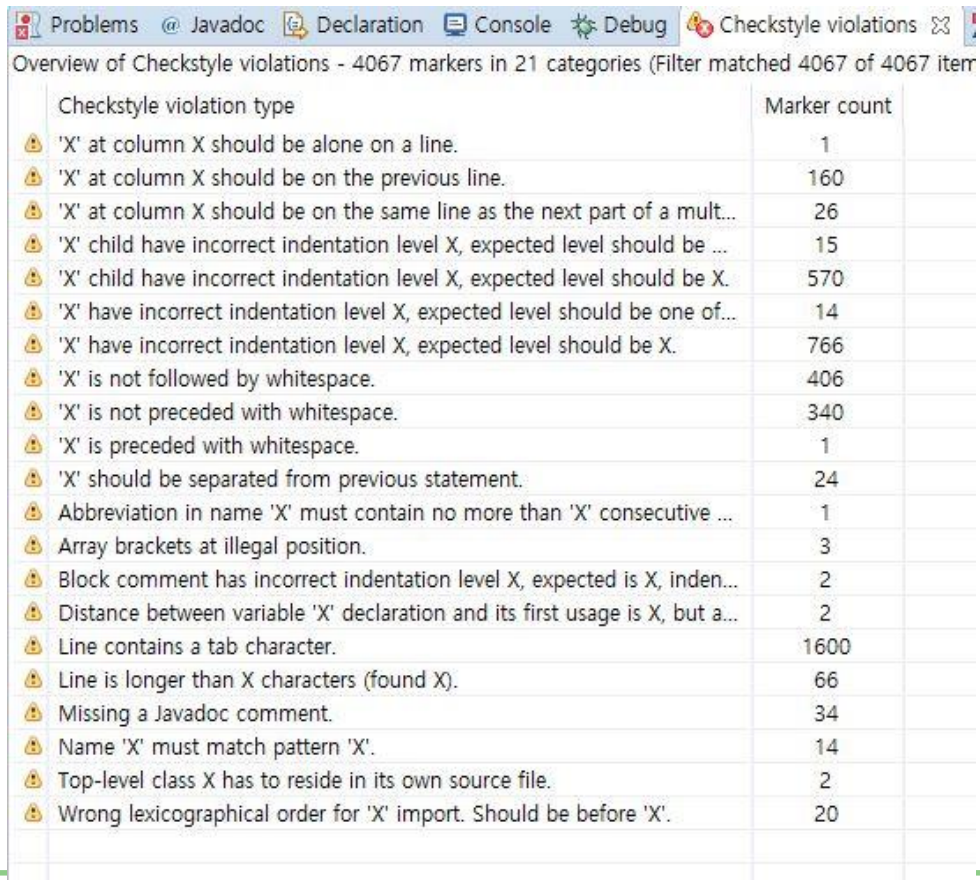
Level 1에서 CheckStyle

Level 2에서 Eclipse Metrics

Level 3에서 FindBugs

# 3 Team 1

## 1. CheckStyle



Checkstyle violation type	Marker count
'X' at column X should be alone on a line.	1
'X' at column X should be on the previous line.	160
'X' at column X should be on the same line as the next part of a mult...	26
'X' child have incorrect indentation level X, expected level should be ...	15
'X' child have incorrect indentation level X, expected level should be X.	570
'X' have incorrect indentation level X, expected level should be one of...	14
'X' have incorrect indentation level X, expected level should be X.	766
'X' is not followed by whitespace.	406
'X' is not preceded with whitespace.	340
'X' is preceded with whitespace.	1
'X' should be separated from previous statement.	24
Abbreviation in name 'X' must contain no more than 'X' consecutive ...	1
Array brackets at illegal position.	3
Block comment has incorrect indentation level X, expected is X, inden...	2
Distance between variable 'X' declaration and its first usage is X, but a...	2
Line contains a tab character.	1600
Line is longer than X characters (found X).	66
Missing a Javadoc comment.	34
Name 'X' must match pattern 'X'.	14
Top-level class X has to reside in its own source file.	2
Wrong lexicographical order for 'X' import. Should be before 'X'.	20

# 3 Team 1

## 2. Eclipse Metrics

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
> McCabe Cyclomatic Complexity (avg/max per method)		1.962	3.461	28	/CptT1/src/File.java	openFile
> Number of Parameters (avg/max per method)		0.577	0.944	6	/CptT1/src/GUI_CPT.java	getTableCellRendererCo...
> Nested Block Depth (avg/max per method)		1.554	1.415	10	/CptT1/src/File.java	openFile
> Afferent Coupling (avg/max per packageFragment)		0	0	0	/CptT1/src	
> Efferent Coupling (avg/max per packageFragment)		1	0	1	/CptT1/src	
> Instability (avg/max per packageFragment)		1	0	1	/CptT1/src	
> Abstractness (avg/max per packageFragment)		0	0	0	/CptT1/src	
> Normalized Distance (avg/max per packageFragment)		0	0	0	/CptT1/src	
> Depth of Inheritance Tree (avg/max per type)		2.5	2.291	6	/CptT1/src/GUI_CPT.java	
> Weighted methods per Class (avg/max per type)	255	25.5	23.61	80	/CptT1/src/File.java	
> Number of Children (avg/max per type)	0	0	0	0	/CptT1/src/RepresentativeValue.java	
> Number of Overridden Methods (avg/max per type)	0	0	0	0	/CptT1/src/RepresentativeValue.java	
> Lack of Cohesion of Methods (avg/max per type)		0.64	0.357	0.963	/CptT1/src/GUI_CPT.java	
> Number of Attributes (avg/max per type)	97	9.7	8.556	27	/CptT1/src/File.java	
> Number of Static Attributes (avg/max per type)	0	0	0	0	/CptT1/src/RepresentativeValue.java	
> Number of Methods (avg/max per type)	129	12.9	11.344	39	/CptT1/src/File.java	
> Number of Static Methods (avg/max per type)	1	0.1	0.3	1	/CptT1/src/GUI_CPT.java	
> Specialization Index (avg/max per type)		0	0	0	/CptT1/src/RepresentativeValue.java	
> Number of Classes (avg/max per packageFragment)	10	10	0	10	/CptT1/src	
> Number of Interfaces (avg/max per packageFragment)	0	0	0	0	/CptT1/src	
> Number of Packages	1					
> Total Lines of Code	1451					
> Method Lines of Code (avg/max per method)	955	7.346	17.503	103	/CptT1/src/GUI_CPT.java	GUI_CPT

# 3 Team 1

## 2. Eclipse Metrics

```
public void openFile() {  
  
    if (f == 1) {  
        try {  
            FileInputStream inputStream = new FileInputStream(operands);  
            XSSFWorkbook workbook = new XSSFWorkbook(inputStream);  
  
            //sheet수 취득  
            int sheetCn = workbook.getNumberOfSheets();  
  
            for(int cn = 0; cn < sheetCn; cn++){  
  
                //0번째 sheet 정보 취득  
                XSSFSheet sheet = workbook.getSheetAt(cn);  
  
                //취득된 sheet에서 rows수 취득  
                rows = sheet.getPhysicalNumberOfRows();  
  
                //취득된 row에서 취득대상 cell수 취득  
                cells = sheet.getRow(cn).getPhysicalNumberOfCells(); //  
  
                for (int r = 1; r <= rows; r++) {  
                    row = sheet.getRow(r); // row 가져오기  
                    if (row != null) {  
                        for (int c = 1; c <= cells; c++) {  
                            cell = row.getCell(c);  
                            if (cell != null) {  
                                String value = cell.getStringCellValue();  
                                //value 출력  
                                System.out.println("value: " + value);  
                            }  
                        }  
                    }  
                }  
            }  
        }  
    }  
}
```

# 3 Team 1

## 2. Eclipse Metrics

```
Color back = new Color(0,0,0,0);
Color match = new Color(208,104,147);
public Component getTableCellRendererComponent(JTable table, Object value, boolean isSelected, boolean hasFocus, int row, int column) {
    super.getTableCellRendererComponent(table, value,
        isSelected, hasFocus, row, column);
    /* if 안에는 다른 클래스로부터 받아온 정보가 있어야 하니, if(row==1) 부분 대신 정보를 받아들 클래스를 이 leftTableCellRender에
스텐스로 생성하고, a.getvalue().equals 이런식으로 ... 해야할듯 합니다.
*/
    if(row==1)
    {
        setBackground(match);
    }
    ---
}
```



# 3 Team 1

## 3. Find Bugs

Problems @ Javadoc Declaration Console Git Staging Synchronize History Bug Explorer

- ▼ CptT1 (15)
  - ▼ Of Concern (15)
    - ▼ Normal confidence (15)
      - ▼ May expose internal representation by returning reference to mutable object (5)
        - CaseController.getEcase() may expose internal representation by returning ecase [Of Concern(18), Normal confidence]
        - CaseController.getTestcase() may expose internal representation by returning testcase [Of Concern(18), Normal confidence]
        - NodeFactory.getValueNode() may expose internal representation by returning valuenode [Of Concern(18), Normal confidence]
        - RepresentativeValue.getIfconst() may expose internal representation by returning ifconst [Of Concern(18), Normal confidence]
        - RepresentativeValue.getProperty() may expose internal representation by returning property [Of Concern(18), Normal confidence]
      - ▼ May expose internal representation by incorporating reference to mutable object (2)
        - CaseController.setEcase(EssentialTestCase[]) may expose internal representation by storing an externally mutable object into ecase [Of Concern(18), Normal confidence]
        - CaseController.setTestcase(TestCase[]) may expose internal representation by storing an externally mutable object into testcase [Of Concern(18), Normal confidence]
      - ▼ Integral division result cast to double or float (2)
        - Integral division result cast to double or float in EssentialTestCase.getWeight() [Of Concern(17), Normal confidence]
        - Integral division result cast to double or float in TestCase.getWeight() [Of Concern(17), Normal confidence]
      - ▼ Method names should start with a lower case letter (1)
        - The method name GUI\_CPT.EcaseEditText() doesn't start with a lower case letter [Of Concern(16), Normal confidence]
      - ▼ Method may fail to clean up stream or resource on checked exception (1)
        - File.exportFile() may fail to clean up java.io.OutputStream on checked exception [Of Concern(20), Normal confidence]
      - ▼ Exception is caught when Exception is not thrown (1)
        - Exception is caught when Exception is not thrown in File.openFile() [Of Concern(17), Normal confidence]
      - ▼ Switch statement found where default case is missing (1)
        - Switch statement found in File.openFile() where default case is missing [Of Concern(19), Normal confidence]
    - ▼ Unread field (1)
      - Unread field: EssentialTestCase.eTestCase [Of Concern(18), Normal confidence]
    - ▼ Unused field (1)
      - Unused field: File.file [Of Concern(18), Normal confidence]

# 4 Team 2 (Client)

## 1. CheckStyle

Overview of Checkstyle violations - 5350 markers in 21 categories (Filter matched 5350 of 5350 ite

Checkstyle violation type	Marker count
⚠ 'X' at column X should have line break after.	1
⚠ 'X' child have incorrect indentation level X, expected level should be ...	2
⚠ 'X' child have incorrect indentation level X, expected level should be X.	1350
⚠ 'X' construct must use '{}'.s.	1
⚠ 'X' have incorrect indentation level X, expected level should be one of...	2
⚠ 'X' have incorrect indentation level X, expected level should be X.	950
⚠ 'X' is not followed by whitespace.	13
⚠ 'X' is not preceded with whitespace.	23
⚠ 'X' should be separated from previous import group by one line.	2
⚠ 'X' should be separated from previous statement.	32
⚠ Abbreviation in name 'X' must contain no more than 'X' consecutive ...	12
⚠ Array brackets at illegal position.	7
⚠ Comment has incorrect indentation level X, expected is X, indentation...	1
⚠ Distance between variable 'X' declaration and its first usage is X, but a...	19
⚠ Each variable declaration must be in its own statement.	35
⚠ Line contains a tab character.	2663
⚠ Line is longer than X characters (found X).	172
⚠ Missing a Javadoc comment.	46
⚠ Name 'X' must match pattern 'X'.	15
⚠ Only one statement per line allowed.	1
⚠ Wrong lexicographical order for 'X' import. Should be before 'X'.	3

# 4 Team 2 (Client)

## 2. Eclipse Metrics

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
> McCabe Cyclomatic Complexity (avg/max pe		1.806	2.268	24	/VersionCPT/src/controller/AdminVer.java	makeTmpFile
> Number of Parameters (avg/max per method)		0.671	0.893	6	/VersionCPT/src/controller/AdminVer.java	makeTmpFile
> Nested Block Depth (avg/max per method)		1.518	1.053	7	/VersionCPT/src/controller/AdminVer.java	makeTmpFile
> Afferent Coupling (avg/max per packageFrag		2	1.414	4	/VersionCPT/src/model	
> Efferent Coupling (avg/max per packageFrag		1.667	1.7	4	/VersionCPT/src/controller	
> Instability (avg/max per packageFragment)		0.433	0.33	0.8	/VersionCPT/src/controller	
> Abstractness (avg/max per packageFragment)		0	0	0	/VersionCPT/src/controller	
> Normalized Distance (avg/max per packageFr		0.567	0.33	1	/VersionCPT/src/model	
> Depth of Inheritance Tree (avg/max per type)		3.296	1.997	6	/VersionCPT/src/view/MainUI.java	
> Weighted methods per Class (avg/max per ty	307	11.37	14.327	77	/VersionCPT/src/controller/AdminVer.java	
> Number of Children (avg/max per type)	0	0	0	0	/VersionCPT/src/controller/AdminProject.java	
> Number of Overridden Methods (avg/max pe	0	0	0	0	/VersionCPT/src/controller/AdminProject.java	
> Lack of Cohesion of Methods (avg/max per t		0.738	0.303	1.111	/VersionCPT/src/view/MainUI.java	
> Number of Attributes (avg/max per type)	195	7.222	4.323	18	/VersionCPT/src/view/MainUI.java	
> Number of Static Attributes (avg/max per ty	6	0.222	1.133	6	/VersionCPT/src/view/MainUI.java	
> Number of Methods (avg/max per type)	167	6.185	6.689	30	/VersionCPT/src/model/Version.java	
> Number of Static Methods (avg/max per type	3	0.111	0.416	2	/VersionCPT/src/view/MainUI.java	
> Specialization Index (avg/max per type)		0	0	0	/VersionCPT/src/controller/AdminProject.java	
> Number of Classes (avg/max per packageFrac	27	9	4.32	15	/VersionCPT/src/view	
> Number of Interfaces (avg/max per packageF	0	0	0	0	/VersionCPT/src/controller	
> Number of Packages	3					
> Total Lines of Code	2481					
> Method Lines of Code (avg/max per method)	1832	10.776	24.229	140	/VersionCPT/src/view/MainUI.java	Version

# 4 Team 2 (Client)

## 2. Eclipse Metrics

```
private void makeTmpFile(int[] pointerArr, LinkedList<Choice>[] choiceListArr, int totalSize, int idx, Version ver,
    BufferedWriter bw) {

    while (pointerArr[totalSize - idx] < choiceListArr[totalSize - idx].size()) {
        if (idx == 1) {

            AdminCase c = new AdminCase();
            boolean flag = true;

            ArrayList<String> propList = new ArrayList<String>();
            ArrayList<String> ifList = new ArrayList<String>();
            ArrayList<String> elseList = new ArrayList<String>();
            HashMap<String, Integer> tempHash = new HashMap<>();

            for (int j = 0; j < totalSize; j++) {

                c.getChoiceList().add(choiceListArr[j].get(pointerArr[j]));
                if (choiceListArr[j].get(pointerArr[j]).isProp()) {
                    Iterator<String> iter = choiceListArr[j].get(pointerArr[j]).getPropList().iterator();
                    while (iter.hasNext()) {

                        String str = iter.next();
                        propList.add(str);
```



# 4 Team 2 (Client)

## 3. Find Bugs

Problems @ Javadoc Declaration Console Debug Bug Explorer Bug Explorer Dependency Graph View Layered Packa

- VersionCPT (12)
  - Troubling (8)
    - High confidence (6)
      - Non-transient non-serializable instance field in serializable class (6)
        - Class controller.Client defines non-transient non-serializable instance field br [Troubling(14), High confidence]
        - Class controller.Client defines non-transient non-serializable instance field ois [Troubling(14), High confidence]
        - Class controller.Client defines non-transient non-serializable instance field oos [Troubling(14), High confidence]
        - Class controller.Client defines non-transient non-serializable instance field pw [Troubling(14), High confidence]
        - Class controller.Client defines non-transient non-serializable instance field sock [Troubling(14), High confidence]
        - Class model.Version defines non-transient non-serializable instance field ctgList [Troubling(14), High confidence]
      - Normal confidence (2)
        - Unwritten field (2)
          - Unwritten field: view.MainUI\$OpinionListPanel.verNum [Troubling(12), Normal confidence]
          - Unwritten field: view.MainUI\$Version.date [Troubling(12), Normal confidence]
      - Of Concern (4)
        - High confidence (4)
          - Dead store to local variable (1)
            - Dead store to ui in controller.Main.main(String[]) [Of Concern(15), High confidence]
          - Write to static field from instance method (3)
            - Write to static field view.MainUI.c from instance method new view.MainUI() [Of Concern(15), High confidence]
            - Write to static field view.MainUI.cards from instance method new view.MainUI() [Of Concern(15), High confidence]
            - Write to static field view.MainUI.mc from instance method new view.MainUI() [Of Concern(15), High confidence]

# 5 Team 2 (Server)

## 1. CheckStyle

Overview of Checkstyle violations - 1819 markers in 18 categories (Filter matched 1819 of 1819 iterations)

Checkstyle violation type	Marker count
'X' child have incorrect indentation level X, expected level should be X.	432
'X' construct must use '{}'. [Warning]	5
'X' have incorrect indentation level X, expected level should be X.	430
'X' is not followed by whitespace.	1
'X' is not preceded with whitespace.	5
'X' should be separated from previous import group by one line.	2
'X' should be separated from previous statement.	3
Abbreviation in name 'X' must contain no more than 'X' consecutive ...	24
Array brackets at illegal position.	2
Block comment has incorrect indentation level X, expected is X, inden...	2
Comment has incorrect indentation level X, expected is X, indentation...	1
Distance between variable 'X' declaration and its first usage is X, but a...	3
Each variable declaration must be in its own statement.	1
Empty catch block.	4
Line contains a tab character.	851
Line is longer than X characters (found X).	5
Missing a Javadoc comment.	36
Name 'X' must match pattern 'X'.	12

# 5 Team 2 (Server)

Problems @ Javadoc Declaration Console Debug Checkstyle violations Checkstyle violations chart Bug Explorer Dependency Graph View Layered

Metric	Total	Mean	Std. D...	Maxim...	Resource causing Maximum	Method
> McCabe Cyclomatic Complexity (avg/max pe		1.736	2.582	25	/VersionCPT/src/controller/ServerThread.java	analyzeMSG
> Number of Parameters (avg/max per method)		0.655	0.878	5	/VersionCPT/src/controller/AdminVer.java	chkAuth
> Nested Block Depth (avg/max per method)		1.373	0.796	5	/VersionCPT/src/controller/AdminVer.java	chkAuth
> Afferent Coupling (avg/max per packageFrag		1.5	1.5	3	/VersionCPT/src/model	
> Efferent Coupling (avg/max per packageFrag		1.5	1.5	3	/VersionCPT/src/controller	
> Instability (avg/max per packageFragment)		0.5	0.5	1	/VersionCPT/src/controller	
> Abstractness (avg/max per packageFragment)		0	0	0	/VersionCPT/src/controller	
> Normalized Distance (avg/max per packageFr		0.5	0.5	1	/VersionCPT/src/model	
> Depth of Inheritance Tree (avg/max per type)		1.091	0.287	2	/VersionCPT/src/controller/ServerThread.java	
> Weighted methods per Class (avg/max per ty	191	17.364	8.315	30	/VersionCPT/src/controller/ServerThread.java	
> Number of Children (avg/max per type)	0	0	0	0	/VersionCPT/src/controller/AdminProject.java	
> Number of Overridden Methods (avg/max pe	1	0.091	0.287	1	/VersionCPT/src/controller/ServerThread.java	
> Lack of Cohesion of Methods (avg/max per t		0.596	0.261	0.894	/VersionCPT/src/model/Version.java	
> Number of Attributes (avg/max per type)	54	4.909	2.678	11	/VersionCPT/src/model/Choice.java	
> Number of Static Attributes (avg/max per ty	0	0	0	0	/VersionCPT/src/controller/AdminProject.java	
> Number of Methods (avg/max per type)	108	9.818	6.422	24	/VersionCPT/src/model/Version.java	
> Number of Static Methods (avg/max per type)	2	0.182	0.575	2	/VersionCPT/src/controller/MainServer.java	
> Specialization Index (avg/max per type)		0.061	0.192	0.667	/VersionCPT/src/controller/ServerThread.java	
> Number of Classes (avg/max per packageFrag	11	5.5	0.5	6	/VersionCPT/src/controller	
> Number of Interfaces (avg/max per packageF	0	0	0	0	/VersionCPT/src/controller	
> Number of Packages	2					
> Total Lines of Code	959					
> Method Lines of Code (avg/max per method)	591	5.373	13.318	124	/VersionCPT/src/controller/ServerThread.java	analyzeMSG

## 2. Eclipse Metrics

# 5 Team 2 (Server)

## 2. Eclipse Metrics

```
public void analyzeMSG(String msg) {
    // 받은 메시지 분석
    String projectName, verNum, ID, PW, newPW, opinionNum;
    Object obj = null;
    LinkedList tmpList;

    switch (msg) {
        case "/LOGIN":
            // ID, PW
            ID = server.rcvFromClient();
            PW = server.rcvFromClient();

            if (adminU.login(ID, PW)) {
                server.sendACK();
            } else {
                server.sendREJ();
            }
            break;

        case "/CHPW":
            // PW, newPW
            PW = server.rcvFromClient();
            newPW = server.rcvFromClient();

            if (adminU.chgPW(PW, newPW)) {
                server.sendACK();
            } else {
                server.sendREJ();
            }
        }
    }
}
```



# 5 Team 2 (Server)

## 3. Find Bugs

- 
- VersionCPT (5)
    - Scariest (1)
      - Normal confidence (1)
        - Self comparison of field with itself (1)
          - Self comparison of AdminVer.tmpVer with itself in controller.AdminVer.delOpinion() [Scariest(3), Normal confidence]
    - Scary (1)
      - High confidence (1)
        - An apparent infinite loop (1)
          - There is an apparent infinite loop in new controller.MainServer() [Scary(8), High confidence]
    - Troubling (3)
      - Normal confidence (3)
        - Dead store to local variable that shadows field (1)
          - Dead store to tmpVer rather than field with same name in controller.AdminVer.delOpinion() [Troubling(11), Normal confidence]
        - Value is null and guaranteed to be dereferenced on exception path (1)
          - obj is null guaranteed to be dereferenced in controller.ServerThread.analyzeMSG(String) on exception path [Troubling(11), Normal confidence]
        - Constructor invokes Thread.start() (1)
          - new controller.MainServer() invokes controller.ServerThread.start() [Troubling(14), Normal confidence]

```
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI" rank="18" priority="2" type="EI_EXPOSE_REP">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI" rank="18" priority="2" type="EI_EXPOSE_REP">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI2" rank="18" priority="2" type="EI_EXPOSE_REP2">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI2" rank="18" priority="2" type="EI_EXPOSE_REP2">
+ <BugInstance first="1" category="STYLE" abbrev="ICAST" rank="17" priority="2" type="ICAST_IDIV_CAST_TO_DOUBLE">
+ <BugInstance first="1" category="PERFORMANCE" abbrev="UrF" rank="18" priority="2" type="URF_UNREAD_FIELD">
+ <BugInstance first="1" category="EXPERIMENTAL" abbrev="OBL" rank="20" priority="2" type="OBL_UNSATISFIED_OBLIGATION_EXCEPTION_EDGE">
+ <BugInstance first="1" category="STYLE" abbrev="REC" rank="17" priority="2" type="REC_CATCH_EXCEPTION">
+ <BugInstance first="1" category="STYLE" abbrev="SF" rank="19" priority="2" type="SF_SWITCH_NO_DEFAULT">
+ <BugInstance first="1" category="PERFORMANCE" abbrev="UuF" rank="18" priority="2" type="UUF_UNUSED_FIELD">
+ <BugInstance first="1" category="BAD_PRACTICE" abbrev="Nm" rank="16" priority="2" type="NM_METHOD_NAMING_CONVENTION">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI" rank="18" priority="2" type="EI_EXPOSE_REP">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI" rank="18" priority="2" type="EI_EXPOSE_REP">
+ <BugInstance first="1" category="MALICIOUS_CODE" abbrev="EI" rank="18" priority="2" type="EI_EXPOSE_REP">
+ <BugInstance first="1" category="STYLE" abbrev="ICAST" rank="17" priority="2" type="ICAST_IDIV_CAST_TO_DOUBLE">
```

The image features a white background with a green border. In the top-left and bottom-right corners, there are green triangles pointing towards the center, creating a frame-like effect.

THANK YOU