

Gusio(거시오) Digital Watch

대응 보고서

Project Team

Team 2

Date

2019-06-10

Member

201511282 이재승

201511291 장유준

201511303 최원경

201511296 조현근

201511293 전상우

- System Testing

2. Brute Force Test

	Ref .	UseCase	detail	결과
1	R1.1	ShowTime	1분 경과 시간을 제대로 출력하는가.	Fail
2			1시간 경과 시간을 제대로 출력하는가.	Fail
3	R1.2	Set time	1시간 14분 22초 시각을 설정할 수 있는가	Fail
4			6월 31일이 설정 안되는가	Fail
5			일을 선택할때, 0에서 31로 내려가는가	Fail
6	R2.1	Stopwatch	스탑워치가 제대로 시작 하는가	Fail

7			1시간 경과 스톱워치가 현재 시각을 제대로 저장하는 가	Fail
8	R2.2	save records	기록 수가 초과시 삭제가 잘 되는가	Fail
9	R2.3	stop stopwatch	스톱워치가 제대로 정지하는 지	Fail
10	R2.5	reset stopwatch	스톱워치가 0으로 초기화 되는지	Fail
11	R3.1	show alarm	알람의 현재 상태가 표시되는지	Fail
12	R3.2	select alarm	저장된 알람을 보여주는지	Fail
13	R3.6	stop alarm	알람의 벨이 울리고 있을 때 종료할 수 있는 지	Fail
14	R4.2	set timer	3분 2초 대입	Fail
15	R4.3	start timer	타이머가 시작되는 지	Fail
16	R4.4	stop timer	타이머가 정지하는 지	Fail
17	R4.5	reset timer	타이머가 0으로 초기화되는 지	Fail
18	R4.6	ring timer	0초가 되면 버저가 울리는지	Fail
19	R5.1	show dice	주사위게임의 현재 상태가 표시되는 지	Fail
20	R5.2	set dice	주사위 개수 4 대입	Fail
21	R5.3	roll dice	2개 설정시, 제대로 값이 출력되는지	Fail
22	R6.1	show RR	룰렛 게임의 현재 상태가 표시되는 지	Fail
23	R6.2	set RR	룰렛 게임의 인원 수 6 대입	Fail
24	R6.3	play RR	룰렛 게임의 확률이 1/인원 수 인지	Fail
25	R7.1	show Menu	메뉴 버튼을 2초간 누르면 메뉴를 보여주는 간	Fail
26	R7.3	check Menu	시계 실행중 확인 버튼을 누르면 ON/OFF가 가능한지	Fail
27	R8.1	change Menu	menu 버튼을 4회 누를 때 반응	Fail

- UI 의 폰트 및 PNG 파일 전송간의 오류로 Test 를 하지 못해 All Fail 이 난 것으로 전달받음
- 폰트 및 PNG 재전송 결과 모든 Test 를 통과함.

3. Category Partitioning Test

	case	result
1	stopwatch menu : 0	fail
2	stopwatch menu : 99 hour >=	fail
3	time check : 99 hour >=	fail

4	alarm state : ringing	fail
5	alarm state : stop	fail
6	timer state : ringing	fail
7	timer state : stop	fail
8	dice game : 0	fail
9	RR game : 0	fail
10	1.1.0.1.0.0.0.0.0.0.0.)	fail
11	1.1.0.2.0.0.0.0.0.0.0.)	fail
12	1.2.0.1.0.0.0.0.0.0.0.)	fail
13	1.2.0.2.0.0.0.0.0.0.0.)	fail
14	1.3.0.1.0.0.0.0.0.0.0.)	fail
15	1.3.0.2.0.0.0.0.0.0.0.)	fail
16	1.4.0.1.0.0.0.0.0.0.0.)	fail
17	1.4.0.2.0.0.0.0.0.0.0.)	fail
18	2.1.0.0.0.0.0.0.0.0.0.)	fail
19	2.2.0.0.0.0.0.0.0.0.0.)	fail
20	2.3.0.0.0.0.0.0.0.0.0.)	fail
21	2.4.0.0.0.0.0.0.0.0.0.)	fail
22	3.1.2.0.0.0.0.0.0.0.0.)	fail
23	3.1.3.0.0.0.0.0.0.0.0.)	fail

24	3.2.2.0.0.0.0.0.0.0.)	fail
25	3.2.3.0.0.0.0.0.0.0.)	fail
26	3.3.2.0.0.0.0.0.0.0.)	fail
27	3.3.3.0.0.0.0.0.0.0.)	fail
28	3.4.2.0.0.0.0.0.0.0.)	fail
29	3.4.3.0.0.0.0.0.0.0.)	fail
30	4.1.0.0.1.0.0.0.0.0.)	fail
31	4.1.0.0.2.0.0.0.0.0	fail
32	4.1.0.0.3.0.0.0.0.0.)	fail
33	4.2.0.0.1.0.0.0.0.0.)	fail
34	4.2.0.0.2.0.0.0.0.0.)	fail
35	4.2.0.0.3.0.0.0.0.0.)	fail
36	4.3.0.0.1.0.0.0.0.0.)	fail
37	4.3.0.0.2.0.0.0.0.0	fail
38	4.3.0.0.3.0.0.0.0.0.)	fail

39	4.4.0.0.1.0.0.0.0.0.)	fail
40	4.4.0.0.2.0.0.0.0.0.)	fail
41	4.4.0.0.3.0.0.0.0.0.)	fail
42	5.1.0.0.0.0.1.0.0.0.)	fail
43	5.1.0.0.0.0.2.0.0.0.)	fail
44	5.1.0.0.0.0.3.0.0.0.)	fail
45	5.2.0.0.0.0.1.0.0.0.)	fail
46	5.2.0.0.0.0.2.0.0.0.)	fail
47	5.2.0.0.0.0.3.0.0.0.)	fail
48	5.3.0.0.0.0.1.0.0.0.	fail
49	5.3.0.0.0.0.2.0.0.0.)	fail
50	5.3.0.0.0.0.3.0.0.0.)	fail
51	5.4.0.0.0.0.1.0.0.0.)	fail
52	5.4.0.0.0.0.2.0.0.0.)	fail
53	5.4.0.0.0.0.3.0.0.0.	fail
54	6.1.0.0.0.0.0.0.2.0.)	fail
55	6.1.0.0.0.0.0.0.3.0.)	fail
56	6.2.0.0.0.0.0.0.2.0.)	fail
57	6.2.0.0.0.0.0.0.3.0.)	fail
58	6.3.0.0.0.0.0.0.2.0	fail
59	6.3.0.0.0.0.0.0.3.0	fail
60	6.4.0.0.0.0.0.0.2.0.)	fail
61	6.4.0.0.0.0.0.0.3.0.)	fail
62	7.1.0.0.0.0.0.0.0.2	fail

63	7.1.0.0.0.0.0.0.0.3.)	fail
64	7.2.0.0.0.0.0.0.0.2	fail
65	7.2.0.0.0.0.0.0.0.3.)	fail
66	7.3.0.0.0.0.0.0.0.2.)	fail
67	7.3.0.0.0.0.0.0.0.3.)	fail
68	7.4.0.0.0.0.0.0.0.2.)	fail
69	7.4.0.0.0.0.0.0.0.3.)	fail
70	8.1.0.0.0.0.0.0.0.0.)	fail
71	8.2.0.0.0.0.0.0.0.0	fail
72	8.3.0.0.0.0.0.0.0.0.)	fail
73	8.4.0.0.0.0.0.0.0.0.)	fail

- UI 의 폰트 및 PNG 파일 전송간의 오류로 Test 를 하지 못해 All Fail 이 난 것으로 전달받음
- 폰트 및 PNG 재전송 결과 모든 Test 를 통과함.

4. Pairwise Test

	case	result
1	1.1.0.1.0.0.0.0.0.0.)	fail
2	1.1.0.2.0.0.0.0.0.0.)	fail
3	1.2.0.1.0.0.0.0.0.0.)	fail
4	1.2.0.2.0.0.0.0.0.0.)	fail
5	1.3.0.1.0.0.0.0.0.0.)	fail
6	1.3.0.2.0.0.0.0.0.0.	fail
7	1.4.0.1.0.0.0.0.0.0.)	fail
8	1.4.0.2.0.0.0.0.0.0.	fail
9	2.1.0.0.0.0.0.0.0.0	fail
10	2.2.0.0.0.0.0.0.0.0.)	fail
11	2.3.0.0.0.0.0.0.0.0.)	fail

12	2.4.0.0.0.0.0.0.0.0.)	fail
13	3.1.2.0.0.0.0.0.0.0.)	fail
14	3.1.3.0.0.0.0.0.0.0.)	fail
15	3.2.2.0.0.0.0.0.0.0.)	fail
16	3.2.3.0.0.0.0.0.0.0.)	fail
17	3.3.2.0.0.0.0.0.0.0.)	fail
18	3.3.3.0.0.0.0.0.0.0.)	fail
19	3.4.2.0.0.0.0.0.0.0.)	fail
20	3.4.3.0.0.0.0.0.0.0.)	fail
21	4.1.0.0.1.0.0.0.0.0.)	fail
22	4.1.0.0.2.0.0.0.0.0	fail
23	4.1.0.0.3.0.0.0.0.0.)	fail
24	4.2.0.0.1.0.0.0.0.0.)	fail
25	4.2.0.0.2.0.0.0.0.0.)	fail
26	4.2.0.0.3.0.0.0.0.0.)	fail
27	4.3.0.0.1.0.0.0.0.0.)	fail
28	4.3.0.0.2.0.0.0.0.0	fail
29	4.3.0.0.3.0.0.0.0.0.)	fail
30	4.4.0.0.1.0.0.0.0.0.)	fail
31	4.4.0.0.2.0.0.0.0.0.)	fail
32	4.4.0.0.3.0.0.0.0.0.)	fail

33	5.1.0.0.0.0.1.0.0.0.)	fail
34	5.1.0.0.0.0.2.0.0.0.)	fail
35	5.1.0.0.0.0.3.0.0.0.)	fail
36	5.2.0.0.0.0.1.0.0.0.)	fail
37	5.2.0.0.0.0.2.0.0.0.)	fail
38	5.2.0.0.0.0.3.0.0.0.)	fail
39	5.3.0.0.0.0.1.0.0.0.	fail
40	5.3.0.0.0.0.2.0.0.0.)	fail
41	5.3.0.0.0.0.3.0.0.0.)	fail
42	5.4.0.0.0.0.1.0.0.0.)	fail
43	5.4.0.0.0.0.2.0.0.0.)	fail
44	5.4.0.0.0.0.3.0.0.0.	fail
45	6.1.0.0.0.0.0.0.2.0.)	fail
46	6.1.0.0.0.0.0.0.3.0.)	fail
47	6.2.0.0.0.0.0.0.2.0.)	fail
48	6.2.0.0.0.0.0.0.3.0.)	fail
49	6.3.0.0.0.0.0.0.2.0	fail
50	6.3.0.0.0.0.0.0.3.0	fail

- UI 의 폰트 및 PNG 파일 전송간의 오류로 Test 를 하지 못해 All Fail 이 난 것으로 전달받음
- 폰트 및 PNG 재전송 결과 모든 Test 를 통과함.

- Static Analysis

1) PMD

9 개의 Vulnerability 발견

The screenshot displays five instances of a PMD vulnerability: "Returning 'inform' may expose an internal array." and "Returning 'dis' may expose an internal array." Each instance is associated with a specific Java file and is classified as a "Vulnerability" with a "Critical" severity. The findings are listed as follows:

- src/main/java/Function/Alarm_Manager.java**: Returning 'inform' may expose an internal array. Vulnerability Critical
- src/main/java/Function/Dice.java**: Returning 'dis' may expose an internal array. Vulnerability Critical
- src/main/java/Function/RR.java**: Returning 'dis' may expose an internal array. Vulnerability Critical
- src/main/java/Function/Stop_Watch.java**: Returning 'dis' may expose an internal array. Vulnerability Critical

Response	<pre> public int [] getDis(){ dis[3]=disNum; dis[6]=isSet; int [] arr=new int []{dis[0],dis[1],dis[2],dis[3],dis[4],dis[5],dis[6]}; return arr; } public int [] mode_getter(){ int [] arr=new int []{mode[0],mode[1],mode[2],mode[3],mode[4],mode[5]}; return arr; } </pre> <p>Class 내의 배열을 그대로 return 하는 과정에 생긴 것으로 판단, 새로운 배열을 만들어 값을 그대로 복사한 뒤 return 하여 해결</p>
----------	---

2 개의 Critical 한 Code Smell 발견

```

41
42     public void mouseClicked(MouseEvent e){
43     }
44     public void mousePressed(MouseEvent e) {
45         if(e.isMetaDown()){
46
47         }
48
49         else
50         {
51             btn_press=System.currentTimeMillis();
52         }
53     }
54     public void mouseReleased(MouseEvent e) {
55         JButton b = (JButton)e.getSource();
56         if(e.isMetaDown()){
57
58         }
59     }

```

Response	<pre> public void mousePressed(MouseEvent e) { if(!e.isMetaDown()){ btn_press=System.currentTimeMillis(); } } </pre> <p>If 문 내의 내용없는 중괄호로 인한 오류로 판단, if 의 조건을 수정함으로 해결</p>
----------	--

2) findBugs

1 개의 버그 발견

The screenshot shows an IDE window with a bug report on the left and code on the right. The bug report is titled "new Function.Time() invokes Thread.start()" and is categorized as a "Major" bug. The code on the right shows a Java class with a method that creates a thread and starts it.

```
14 private int setFlow;
15 public Time(){
16     system_clock = new System_Clock(1,1,0,0,0,0);
17     Thread t = new Thread(system_clock, "내부클럭");
18     t.start();
19
19     month = 1;
20     date = 1;
```

Response	<pre>private void runThread(){ Thread t = new Thread(system_clock, name: "내부클럭"); t.start(); }</pre> <p>생성자에서의 스레드 객체 생성이 문제인 것으로 판단, 스레드 객체 생성 함수를 만들어 해결</p>
----------	--

2 개의 Major 한 Code Smell 발견

The screenshot shows an IDE window with two code smell reports on the left and code on the right. The code smell reports are titled "The method name System_Support.System_Clock.Flag() doesn't start with a lower case letter" and are categorized as "Major" code smells. The code on the right shows a Java class with a method named Flag() that sets a flag.

```
29 mlimit[9] = 30;
30 mlimit[10] = 31;
31 mlimit[11] = 30;
32 mlimit[12] = 31;
33 }
34
35 public void Flag() {
36     flag = 0;
37 }
38
39 @Override
40 public void run() {
```

Response	<pre>public void setFlag() { flag = 0; }</pre> <p>함수의 이름이 명사로 선언되어 생긴 오류로 판단, 동사형으로 바꿔주어 해결</p>
----------	---

- Review

이재승 : 많이 힘들기도 했지만 객체지향에 대해 많은 것을 알 수 있었고, 개발을 할 때 설계 과정이 매우 중요하다는 걸 뼈저리게 느꼈습니다. 많은 것을 얻어 갈 수 있어 후회없는(ㄹㅇ?) 수업이었습니다.

전상우 : 설계가 직접 구현되는 과정을 보며 스릴과 짜릿함을 느낄 수 있었습니다.

장유준 : UP 개발의 이해를 통해 소프트웨어 설계의 단계를 깊게 공부했고 객체지향 언어를 한층 깊게 이해했습니다.

조현근 : 프로젝트 진행 시 단계별 진행의 중요성에 대해 알게 되었습니다

최원경 : 여러명에서 만드는 큰 프로젝트일수록 설계가 중요하다는 걸 느꼈습니다.