

V&V프로젝트 - A6
static analysis, system testing

공민정, 이규은, 김태형, 최지현

- 
- 1. Feedback**
 - 2. System Testing**
 - 3. Static Analysis**
 - 4. Coverage**
 - 5. Overall**

INDEX

Use Case	8. Time Out
Actor	System
Purpose	버튼 입력이 없을 때, 그 동안 지난 시간을 계산한다.
Overview	버튼 입력이 없을 때, 그 동안 지난 시간을 계산하여 정해진 시간을 넘길 시 Time Mode로 전환한다.
Type	Hidden
Pre-Requisites	N/A
Typical Courses of Events	(U) : User, (S) : System 1. (S) : Mode 진입 시 Timeout 시간을 0초로 초기화한다. 2. (S) : Timeout 시간을 10ms마다 늘린다. 3. (S) : 버튼 입력을 받으면 Timeout 시간을 0초로 초기화한다. 4. (S) : Timeout 시간이 10초가 될 경우 해당 시간을 정지하고 Time Mode로 전환한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	(U) : User, (S) : System 1. (S) : Time Mode일 때와 Start Stopwatch, Set Timer, Start Timer, Set Alarm, Start Game일 시에는 Time Out이 일어나지 않는다.

Use Case	32. Select City
Actor	User
Purpose	보고싶은 도시를 선택한다.
Overview	보고싶은 도시를 선택하여 해당 도시의 시각을 계산한다.
Type	Evident
Pre-Requisites	Worldtime Mode이어야 한다.
Typical Courses of Events	(U) : User, (S) : System 1. (U) : D 버튼(다음 시간대의 도시 선택)을 누른다. 2. (S) : 선택된 도시를 다음 시간대의 도시로 바꾼다. 3. (S) : 바뀐 도시의 시각을 구하기 위해 우리나라 시각 기준 시각을 현재 시각에 더하고, Summertime이 활성화 되었을 경우 1시간을 추가로 더해서 계산한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

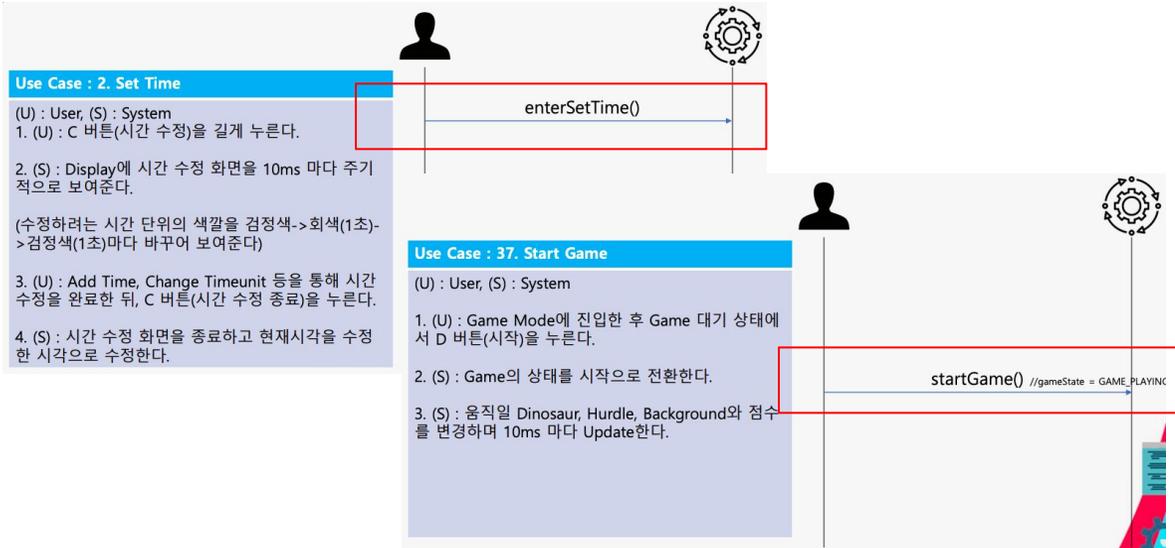
- Typical Courses of Events의 3번은 Alternative Courses of Events로 이동

- Typical Courses of Events의 3번은 Alternative Courses of Events로 이동

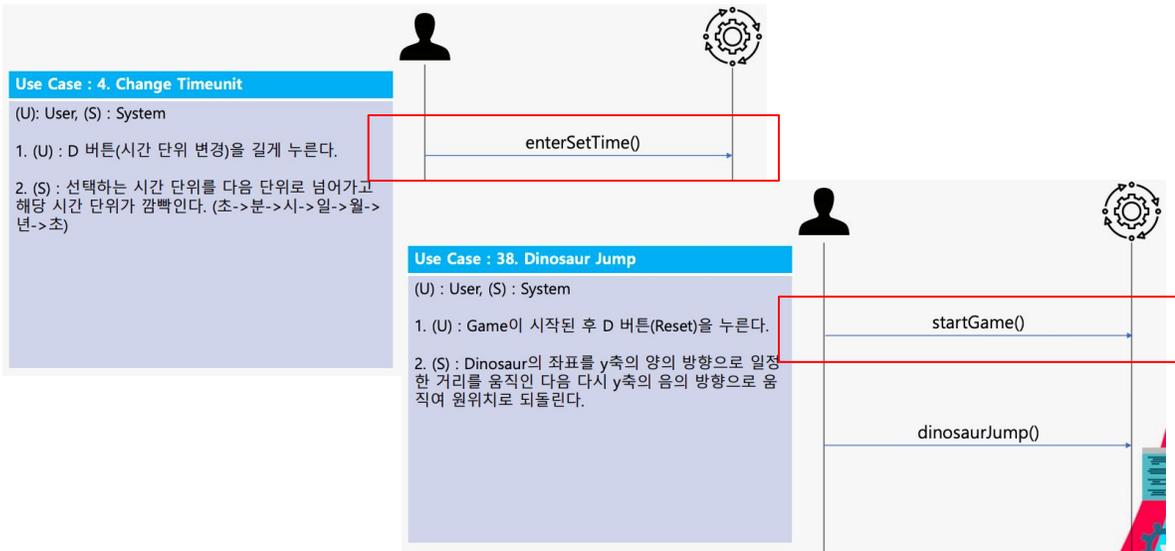
- 마지막 도시에서 D 버튼을 눌렀을 때 어떤 화면으로 이동하는지 명시

01

Feedback



TimeUnit을 변경할 때마다 enterSetOOO 호출



Dinosaur jump할 때마다 startGame 호출

02

System Testing - Category-Partition Testing

#	Mode	Actions	Checks	Key	Result
1	display time	change time format	Check valid time	1.1.1	P
2	display time	change time format	check counting time	1.1.2	P
3	display time	change time format	check changing time unit	1.1.3	P
4	display time	change time format	check <u>oclock</u>	1.1.4	P
5	display time	change time format	valid beep	1.1.11	P
6	display time	goto next function	check <u>oclock</u>	1.2.4	P
7	display time	goto next function	valid beep	1.2.11	P
8	set Time	set time	valid value change	2.3.9	P
9	set Time	set Time	valid beep	2.3.11	P
10	display timer	active timer	ring beep	3.4.5	P
11	display timer	active timer	stop beep	3.4.6	P
12	display timer	active timer	check counting down	3.4.7	P
13	display timer	active timer	valid beep	3.4.11	P
14	display timer	pause timer	check no counting down	3.5.8	P
15	display timer	pause timer	valid beep	3.5.11	P
16	display timer	reset timer	check no counting down	3.6.8	P
17	display timer	reset timer	valid beep	3.6.11	P
18	display timer	goto next function	valid beep	3.7.11	P
19	set timer	set timer	valid value change	4.8.9	P
20	set timer	set timer	valid beep	4.8.11	P
21	display stopwatch	active stopwatch	valid beep	5.9.11	P
22	display stopwatch	pause stopwatch	valid beep	5.10.11	P
23	display stopwatch	reset stopwatch	valid beep	5.11.11	P
24	display stopwatch	lap stopwatch	valid beep	5.12.11	P
25	display stopwatch	goto next function	valid beep	5.13.11	P
26	display alarm	see next alarm	check valid alarm list	6.14.10	P
27	display alarm	see next alarm	valid beep	6.14.11	P
28	display alarm	enable alarm	check valid alarm list	6.16.10	P
29	display alarm	enable alarm	valid beep	6.16.11	P
30	display alarm	enable alarm	stop beep	6.16.12	P
31	display alarm	enable alarm	show alarming	6.16.13	P
32	display alarm	change active alarm	check valid alarm list	6.17.10	P
33	display alarm	change active alarm	valid beep	6.17.11	P
34	display alarm	change active alarm	stop beep	6.17.12	P
35	display alarm	change active alarm	show alarming	6.17.13	P
36	set alarm	set alarm	valid value change	7.18.9	P
37	set alarm	set alarm	valid beep	7.18.11	P
38	set alarm	turn alarm on	valid value change	7.19.9	P
39	set alarm	turn alarm on	valid beep	7.19.11	P
40	set alarm	turn alarm off	valid value change	7.20.9	P
41	set alarm	turn alarm off	valid beep	7.20.11	P
42	globaltime	goto next function	valid beep	8.21.11	P
44	setSummerTime	<n/a>	valid beep	9.0.11	P
45	display game	start game	valid beep	10.23.11	P
46	play game	<n/a>	valid beep	11.0.11	P

46/46 x 100 = 100%

02

System Testing - Pairwise Testing

mode	running	beep	action	Pass/Fail
time	running	ringing	nextFunc	P
timer	running	stop	reset	P
stopwatch	stop	ringing	reset	P
stopwatch	running	ringing	record	P
stopwatch	stop	stop	off	P
alarm	running	stop	on	P
game	stop	stop	end	P
timer	stop	stop	nextFunc	P
stopwatch	running	ringing	nextFunc	P
game	running	ringing	on	P
alarm	running	stop	nextFunc	P
timer	running	stop	on	P
alarm	running	ringing	setTime	P
alarm	running	stop	list	P
stopwatch	stop	stop	on	P
timer	stop	stop	pause	P
timer	stop	stop	setTime	P
world time	running	ringing	nextFunc	P
game	stop	stop	play	P
world time	running	stop	off	P
world time	running	ringing	on	P

$$21/21 * 100 = 100\%$$

Test	Num	Description	P/F
Timekeeping	1-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
	1-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	1-3	윤년인 해에는 2월29일까지 있는지 제대로 확인한다.	P
	1-4	시간 설정시 시간단위가 깜빡이는지 확인한다.	P
	1-5	시간이 정상적으로 흘러가는지 확인한다.	P
Timer	2-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
	2-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	2-3	타이머 설정시 시간단위가 깜빡이는지 확인한다.	P
	2-4	설정된 시간에 beep이 울리는지 확인한다.	P
	2-5	다른 beep과 동일한 시간에 울릴 때, 울려야 할 beep의 개수만큼 울리는지 확인한다.	F
Alarm	3-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
	3-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	3-3	알람 설정시 시간단위가 깜빡이는지 확인한다.	P
	3-4	알람이 5개가 저장되지 않는지 확인한다.	P
	3-5	설정된 시간에 beep이 울리는지 확인한다.	P
	3-6	알람 여러 개가 같은 시간에 울릴 때, 개수만큼 울리는지 확인한다.	F
	3-7	알람 외 다른 beep과 동일한 시간에 울릴 때, 울려야 할 beep의 개수만큼 울리는지 확인한다.	F
Stopwatch	4-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
	4-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	4-3	Lap Time이 되는지 확인한다.	P
WorldTime	5-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F

	5-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	5-3	Summertime 해제 시 기본시간으로 바뀌는지 확인한다.	P
Game	6-1	지정한 버튼을 눌렀을 때만 게임이 시작되는지 확인한다.	P
	6-2	무작위 버튼을 눌렀을 때 에러가 나지 않는지 확인한다.	P
	6-3	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
NextMode	7-1	모든 버튼을 눌렀을 때 잘 작동하는지 확인한다.	F
O'clock	8-1	다른 beep과 동일한 시간에 울릴 때, 울려야 할 beep의 개수만큼 울리는지 확인한다.	F

1. 버튼의 Short Press와 Long Press가 구분되지 않고 동일한 기능이 수행되는 문제
2. beep이 동일한 시간에 누적되었을 때, 하나의 beep만 울리는 문제

39.1% -> $17/28 \times 100 = 60.7\%$

03

Static Analysis - Overall

12 Bugs	Reliability C	
46 Vulnerabilities	Security B	
23 Security Hotspots 0.0% Reviewed	Security Review E	
93d Debt	3.2k Code Smells	Maintainability A

Bug	12
Vulnerability	46
Code Smell	3.2k

0.0%
Coverage on [1.4k](#) Lines to cover
Unit Tests

4.1%
Duplications on [2.2k](#) Lines
Duplicated Blocks **8**

▼ Severity			
Blocker	0	Minor	9
Critical	0	Info	0
Major	3		

<Bug>

▼ Severity			
Blocker	0	Minor	46
Critical	0	Info	0
Major	0		

<Vulnerability>

Blocker	4	Minor	2.2k
Critical	48	Info	138
Major	764		

<Codesmell>

03

Static Analysis - Bugs (Major)

```
try{
    Thread.sleep(msSleep, nanoSleep);
} catch (InterruptedException e){
    e.printStackTrace();
}
lastTime = System.nanoTime();
}
```

Either re-interrupt this method or rethrow the "InterruptedException". Why is this an issue? 11 days ago ▾ L100 🔗

🐛 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ Not assigned ▾ 15min effort Comment 🔍 cwe, error-handling, multi-threading ▾

InterruptedException handling 시 logging 이외에 thread의 interrupted state 분명히 명시할 것
ex. Thread.currentThread.interrupt()

```
stopwatch = new Stopwatch();
```

Unread field: clock.ModeManager.beep Why is this an issue? 10 days ago ▾ L

🐛 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ Not assigned ▾ 30min effort Comment 🔍 perform

메소드 내에서 사용하지 않는 불필요한 코드

03

Static Analysis - Code Smell (Blocker)

```
18 private static final int JUMPING = 1;
private BufferedImage jumping;
```

Rename field "jumping" to prevent any misunderstanding/clash with field "JUMPING" defined on line 18. Why is this an issue? 10 days ago ▾ L32 🔗

🚫 Code Smell ▾ 🚫 Blocker ▾ 🔵 Open ▾ Not assigned ▾ 10min effort Comment 🗑️ confusing ▾

JUMPING, jumping 변수명 혼동 가능성 있음.

```
public class WatchSystem extends JPanel implements MouseListener, KeyListener, Runnable {
    public ModeManager Watch;

    private BufferedImage background;

    private long isLongpress;
    private long timeOut;

    private Font font;
```

"background" is the name of a field in "Component". Why is this an issue? 8 days ago ▾ L24 🔗

🚫 Code Smell ▾ 🚫 Blocker ▾ 🔵 Open ▾ Not assigned ▾ 5min effort Comment 🗑️ confusing ▾

"font" is the name of a field in "Component". Why is this an issue? 8 days ago ▾ L28 🔗

🚫 Code Smell ▾ 🚫 Blocker ▾ 🔵 Open ▾ Not assigned ▾ 5min effort Comment 🗑️ confusing ▾

JPanel에 background component와 font component 존재함

03

Static Analysis - Code Smell (Critical)

```
timeUnit[Unit]++;
```

Make the enclosing method "static" or remove this set. Why is this an issue? 52 minutes ago L39

Code Smell Critical Open Not assigned 20min effort Comment multi-threading

```
if (timeUnit[0] == 60){  
    timeUnit[0] = 0;  
}
```

Make the enclosing method "static" or remove this set. Why is this an issue? 52 minutes ago L41

Code Smell Critical Open Not assigned 20min effort Comment multi-threading

```
}
```

```
if (timeUnit[1] == 24){  
    timeUnit[1] = 0;  
}
```

Make the enclosing method "static" or remove this set. Why is this an issue? 52 minutes ago L44

Code Smell Critical Open Not assigned 20min effort Comment multi-threading


```
private Thread timeThread;
```

Make "timeThread" transient or serializable. Why is this an issue? 8 days ago L10

Code Smell Critical Open Not assigned 30min effort Comment cwe, serialization

static으로 선언된 timeUnit을 non-static method에서 사용

-> static 변수를 static이 아닌 변수로 바꾸기

timeThread를 serialize하게 할지 말지 명확히 할 것

03

Static Analysis - Code Smell (Critical)

```
switch (state){  
  case NORMAL_RUN:  
    g.drawImage(normalRunAnim.getFrame(), (int) posX, (int) posY, null);  
    break;  
  case JUMPING:  
    g.drawImage(jumping, (int) posX, (int) posY, null);  
    break;  
  case DEATH:  
    g.drawImage(deathImage, (int) posX, (int) posY, null);  
    break;  
}
```

Add a default case to this switch. Why is this an issue? 10 days ago L58

Code Smell Critical Open Not assigned 5min effort Comment cert, cwe

가급적 Default문 작성하기

```
beepSound = Applet.newAudioClip(Resource.class.getClassLoader().getResource("resources/beep.wav"));
```

In J2EE, getClassLoader() might not work as expected. Use Thread.currentThread().getContextClassLoader() instead. Why is this an issue? 5 days ago L15

Code Smell Critical Open Not assigned 15min effort Comment No tags

getClassLoader()는 잘 작동이 안될 수 있으므로
Thread.currentThread().getContextClassLoader()
사용 권장

03

Static Analysis - Code Smell (Critical)

```
top = Resource.getFont( 1 "resources/scoreboard.ttf", 60);
```

Define a constant instead of duplicating this literal "resources/scoreboard.ttf" 3 times. 5 days ago ▾ L52 🔗
Why is this an issue?
🗑️ Code Smell ▾ 🚨 Critical ▾ 🔓 Open ▾ Not assigned ▾ 8min effort Comment 🗨️ design ▾

```
main = Resource.getFont( 2 "resources/scoreboard.ttf", 170);  
sub = Resource.getFont( 3 "resources/scoreboard.ttf", 60);  
} catch (Exception e) {  
    e.printStackTrace();  
}
```

반복되는 **string**은 변수에 저장할 것

```
}catch(IllegalAccessException e){  
  
}
```

Avoid empty catch blocks Why is this an issue? 2 days ago ▾ L133 🔗
🗑️ Code Smell ▾ 🚨 Critical ▾ 🔓 Open ▾ Not assigned ▾ 10min effort Comment 🗨️ error-handling ▾

Catch 블록 비우지 말것

03

Static Analysis - Code Smell (Major)

```
g.drawString(String.format("%8s",data[0].substring(5, 8)), 100 , 388);
```

Potential violation of Law of Demeter (method chain calls) Why is this an issue?

3 days ago L234

Code Smell Major Open Not assigned 30min effort Comment

No tags

클래스의 필드이거나 메소드, 파라미터로 받은 메소드와 같이 가까운 것만 호출하고 그 외 2,3차는 호출하지 말 것

```
private WatchSystem watchSystem;
```

Found non-transient, non-static member. Please mark as transient or provide accessors.

9 days ago L12

Why is this an issue?

Code Smell Major Open Not assigned 30min effort Comment

No tags

데이터를 표현하는 Bean 클래스이거나 Bean으로부터 참조된 클래스는 직렬화 (serializable)가 필요함

```
}catch (Exception e){  
    e.printStackTrace();
```

Avoid printStackTrace(); use a logger call instead. Why is this an issue?

11 days ago L17

Code Smell Major Open Not assigned 30min effort Comment

error-handling

예외 처리시 printStackTrace 보다는 Logger.log() 권장

03

Static Analysis - Code Smell (Major)

```
private WatchSystem watchSystem;
```

Private field 'watchSystem' could be made final; it is only initialized in the declaration or constructor. 9 days ago L12

Why is this an issue?

Code Smell Major Open Not assigned 10min effort Comment No tags

```
public void updateStopw(){  
    if (this.isPaused == false)
```

Change this instance-reference to a static reference. Why is this an issue? 8 days ago L60

Code Smell Major Open Not assigned 5min effort Comment pitfall

```
private class ImageBackground{  
    float posX;
```

Variable 'posX' must be private and have accessor methods. Why is this an issue? 11 days ago L87

Code Smell Major Open Not assigned 10min effort Comment No tags

```
    BufferedImage image;
```

Variable 'image' must be private and have accessor methods. Why is this an issue? 11 days ago L88

Code Smell Major Open Not assigned 10min effort Comment No tags

생성자에서 한 번만 초기화되고 이후에는 바뀌지 않으므로 **final** 추가 권장

static 변수는 바꿀 때마다 한 클래스의 여러 인스턴스들에서 동시에 변경되므로 **static**을 제외하는 것이 좋음.

주요 변수들은 **private**으로 선언하고 접근할 수 있는 **getter, setter**를 생성

03

Static Analysis - Code Smell (Major)

```
try {
    beepSound = Applet.newAudioClip(Resource.class.getClassLoader().getResource("resources/beep.wav"));
} catch (Exception e){
```

Avoid catching generic exceptions such as NullPointerException, RuntimeException, Exception in try-catch block Why is this an issue? 6 days ago ▾ L16 🔗

🔗 Code Smell ▾ 🔴 Major ▾ ○ Open ▾ Not assigned ▾ 15min effort Comment 🔗 error-handling ▾

```
    e.printStackTrace();
}
```

```
try {
    beepSound = Applet.newAudioClip(Resource.class.getClassLoader().getResource("resources/beep.wav"));
} catch (Exception e){
```

Catching 'Exception' is not allowed. Why is this an issue? 6 days ago ▾ L16 🔗

🔗 Code Smell ▾ 🔴 Major ▾ ○ Open ▾ Not assigned ▾ 15min effort Comment 🔗 error-handling ▾

```
    e.printStackTrace();
}
```

Exception으로 처리하지 말고 NullPointerException 등 정확하게 특정하여 catch할 것

Exception, RuntimeException 등으로 catch하지 말 것

```
private class ImageBackground{
```

Class should define a constructor. Why is this an issue? 11 days ago ▾ L86 🔗

🔗 Code Smell ▾ 🔴 Major ▾ ○ Open ▾ Not assigned ▾ 30min effort Comment 🔗 No tags ▾

```
private class ImageBackground{
    float posX;
    BufferedImage image;
}
```

Each class should declare at least one constructor Why is this an issue? 11 days ago ▾ L89 🔗

🔗 Code Smell ▾ 🔴 Major ▾ ○ Open ▾ Not assigned ▾ 15min effort Comment 🔗 No tags ▾

생성자를 만들 것

03

Static Analysis - Code Smell (Major)

```
private static int timeUnit[];
```

Possible unsafe assignment to a non-final static field in a constructor. Why is this an issue? 4 days ago L11

Code Smell Major Open Not assigned 10min effort Comment No tags

```
public Alarm() {  
    alarms = new LinkedList<Calendar>();  
    toggle = new LinkedList<Boolean>();  
    timeUnit = new int[2];  
}
```

Remove this assignment of "timeUnit". Why is this an issue? yesterday L19 1

Code Smell Major Open Not assigned 20min effort Comment No tags

static 변수를 생성자에서 선언하면 모든 인스턴스들이 공유하는 **static** 변수의 값이 인스턴스를 만들 때마다 달라지므로 **static**이 아닌 변수로 바꿀 것

60% classes, 31% lines covered in package 'clock'

Element	Class, %	Method, %	Line, %
Alarm	100% (1/1)	84% (11/13)	91% (61/67)
Background	33% (1/3)	16% (1/6)	12% (6/47)
Buzzer	100% (1/1)	50% (1/2)	62% (5/8)
Cactus	0% (0/1)	0% (0/5)	0% (0/22)
Dinosaur	100% (1/1)	10% (1/10)	25% (13/52)
Game	100% (1/1)	9% (1/11)	7% (5/69)
Hurdle	0% (0/1)	0% (0/6)	0% (0/34)
ModeManager	0% (0/1)	0% (0/14)	0% (0/195)
Stopwatch	100% (1/1)	72% (8/11)	52% (32/61)
Time	100% (1/1)	70% (7/10)	85% (68/80)
Timer	100% (1/1)	69% (9/13)	61% (49/80)
WatchSystem	0% (0/1)	0% (0/13)	0% (0/72)
Worldtime	100% (1/1)	40% (4/10)	55% (20/36)

WatchSystem, Modemanager은 Unit Test Code가 전혀 없기에 0%.

04

Coverage

▼ ! clock	335 ms
▶ ✓ AlarmTest	29 ms
▶ ! GameTest	181 ms
▶ ✓ ModeManagerTest	8 ms
▶ ✓ StopwatchTest	42 ms
▶ ✓ TimerTest	51 ms
▶ ✓ TimeTest	10 ms
▶ ✓ WatchSystemTest	6 ms
▶ ✓ WorldtimeTest	8 ms

Unit Test 1 Failed

-> 접근할 수 없는 private 멤버에 접근하여 IllegalAccessException 발생

! Tests failed: 1, passed: 56 of 57 tests – 328 ms

```
java.lang.IllegalAccessException: Class clock.WorldtimeTest can not access a member of class clock.Worldtime with modifiers "private" <1 internal call>
    at java.lang.reflect.AccessibleObject.slowCheckMemberAccess(AccessibleObject.java:296)
    at java.lang.reflect.AccessibleObject.checkAccess(AccessibleObject.java:288)
    at java.lang.reflect.Field.get(Field.java:390)
    at clock.WorldtimeTest.requestWorldtime(WorldtimeTest.java:24) <15 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1257) <5 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1257) <17 internal calls>

java.lang.IllegalAccessException: Class clock.WorldtimeTest can not access a member of class clock.Worldtime with modifiers "private" <1 internal call>
    at java.lang.reflect.AccessibleObject.slowCheckMemberAccess(AccessibleObject.java:296)
    at java.lang.reflect.AccessibleObject.checkAccess(AccessibleObject.java:288)
    at java.lang.reflect.Field.get(Field.java:390)
    at clock.WorldtimeTest.caWorldTime(WorldtimeTest.java:123) <15 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1257) <5 internal calls>
    at java.util.ArrayList.forEach(ArrayList.java:1257) <17 internal calls>
```

04

Coverage

Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
test2	49.4 %	3,059	3,134	6,193
src/main/java	33.3 %	1,344	2,686	4,030
(default package)	0.0 %	0	62	62
clock	33.7 %	1,305	2,565	3,870
Alarm.java	80.7 %	330	79	409
Background.java	8.3 %	17	187	204
Buzzer.java	33.3 %	6	12	18
Cactus.java	0.0 %	0	111	111
Dinosaur.java	22.1 %	49	173	222
Game.java	5.5 %	13	225	238
Hurdle.java	0.0 %	0	184	184
ModeManager.java	0.0 %	0	822	822
Stopwatch.java	44.0 %	109	139	248
Time.java	90.8 %	356	36	392
Timer.java	52.5 %	170	154	324
WatchSystem.java	0.0 %	0	256	256
Worldtime.java	57.7 %	255	187	442
util	39.8 %	39	59	98
src/test/java	79.3 %	1,715	448	2,163
clock	79.3 %	1,715	448	2,163
AlarmTest.java	93.8 %	423	28	451
GameTest.java	6.8 %	3	41	44
ModeManagerTest.java	100.0 %	18	0	18
StopwatchTest.java	87.2 %	238	35	273
TimerTest.java	96.1 %	323	13	336
TimeTest.java	95.4 %	472	23	495
WatchSystemTest.java	100.0 %	16	0	16
WorldtimeTest.java	41.9 %	222	308	530

```
1 package clock;
2 import java.util.*;
3
4 public class Worldtime implements Mode{
5
6     private int curCity;
7     private int timeDiff[];
8     private String city[];
9     private boolean isSummerTime;
10    private Calendar worldClock;
11    private Calendar GMT9;
12
13    public Worldtime(Calendar curTime) {
14        timeDiff = new int[] {-20, -19, -18, -17, -16, -15, -14, -13, -12, -11, -10, -9, -8, -7, -6, -5, -4, -3, -2, -1, 0, 1, 2, 3};
15        city = new String[] {"PAGO PAGO", "HAWAII", "ALASKA", "LA", "DENVER", "CHICAGO", "NEW YORK", "CARACAS", "SAO PAULO", "FERNANDO", "AZORES", "LOND"};
16        curCity = 20;
17        isSummerTime = false;
18
19        GMT9 = curTime;
20        calWorldTime();
21    }
22
23    public String[] requestWorldtime(Calendar curTime) {
24        update(curTime);
25        calWorldTime();
26        // TODO implement here
27        if(isSummerTime == true)
28            return new String[] {"WORLD-" + city[curCity] + "(S)", String.format("%02d", worldClock.get(Calendar.HOUR_OF_DAY)) + ":" + String.format("%02d", worldClock.get(Calendar.MINUTE)) + ":" + String.format("%02d", worldClock.get(Calendar.SECOND))};
29        else
30            return new String[] {"WORLD-" + city[curCity], String.format("%02d", worldClock.get(Calendar.HOUR_OF_DAY)) + ":" + String.format("%02d", worldClock.get(Calendar.MINUTE)) + ":" + String.format("%02d", worldClock.get(Calendar.SECOND))};
31    }
32 }
```

실제 코드에서 지나가지 않는 빨간 부분

[Spec Review]

- 미흡한 부분을 구현한 내용에 맞게 업데이트
- 아직 보완되지 않은 부분 존재

[System Testing]

- Category Partition Testing, Pairwise Testing: 0% -> 100%
- Bruteforce Testing: 39.1% -> 60.7%
- 대부분의 버그 해결 완료
- 버튼 입력 처리, 다수의 beep 관련하여 처리

[Static Analysis]

- Thread를 사용하면서 serialize를 결정하지 않거나 interrupt를 처리하지 않음
- 전체적으로 static 변수를 잘못 사용함
- 생성자가 만들어지지 않은 클래스 다수 존재
- Exception Handling 미흡

[Coverage]

- Unit Test Code 보충 필요

감사합니다!