



#5 2nd Testing - System Test/Static Analysis

Software Verification

Software Verification Team 4

강 송 신 정 상 승 모 연 화

INDEX

1

2nd Specification
Review

- Team 7
- Team 8

2

2nd System Test

- Team 7 Category-Partition Test
- Team 8 Category-Partition Test
- Team 7 Pairwise Test
- Team 8 Pairwise Test
- Team 7 Brute Force Test
- Team 8 Brute Force Test

3

Static Analysis

- Team 7
- Team 8

4

Overall

1

2nd Specification Review

- Team 7 Specification Review
- Team 8 Specification Review



2nd Specification Review
- Team 7 -



Team 7 Specification Review

Stage 2040 Design

1.3 Start Timer(R.2.1)

Use Case	3.Start Timer
Actors	User
Purpose	User가 타이머를 동작한다.
Overview	User가 타이머의 시간을 흐르게 한다.
Type	Evident
Cross Reference	Function : R.2.2, R.10.0 Use Case : "Stop Timer", "Alarm"
Pre-Requisites	Timer 시간 멈춤 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 타이머 시작버튼을 누른다. 2. (S) 타이머에 저장된 시간을 1초씩 줄이며 흐르게 한다. 3. (S) 타이머 시간이 0이 되면 알람을 10초 울린다. 4. (S) 타이머 시간 화면에 초과된 시간을 1초씩 흐르게 한다.
Alternative Courses of Events	
Exceptional Courses of Events	E1. 타이머가 이미 동작 상태이면 동작을 멈춘다. E3. 알람이 울리는 동안 버튼을 누르면 알람이 꺼진다. E4. 타이머가 양으로 흐르는 상태에서 59분 59초 이후에 흐른다면 0분으로 타이머가 초기화되고 멈춘다.

- 알람이 울리지 않음
→ 소리 X , 알람 표시 X

Team 7 Specification Review

Stage 2050 Implementation

1.1 Set Time(R.1.0)

Name	SetTime
Responsibilities	시계의 현재 시간을 조정한다.
Type	UI
Cross References	R.4.3, R.6.0
Note	입력된 버튼에 따라 연, 월, 일, 시, 분을 1씩 조정할 수 있다.
Pre-Conditions	현재 시간을 보는 상태
Post-Conditions	조정된 시간을 사용자에게 보여준다.

- 현재 시간 조정 불가

→ 조정하고 난 뒤 조정된 시간이 적용이 되지 않음 (시, 분 모두)

Team 7 Specification Review

Stage 2050 Implementation

1.12 Set Number Range(R.5.2)

Name	Set Number Range
Responsibilities	생성될 난수의 범위 를 조정한다.
Type	UI
Cross References	
Note	버튼을 입력하여, 생성될 난수 범위를 1씩 증가시킨다. default 난수 범위는 0이다.
Pre-Conditions	RNG(난수생성)모드 상태.
Post-Conditions	생성될 난수 범위가 저장되고, 사용자에게 보여준다.

- 범위의 최대값에 대한 부분 누락
→ 최대값 59

1.14 Reset Number Range(R.5.2)

Name	Reset Number Range
Responsibilities	난수 생성 범위를 0으로 초기화 한다.
Type	UI
Cross References	
Note	버튼을 입력하여, 난수 생성 범위를 0으로 초기화한다.
Pre-Conditions	RNG(난수생성)모드 상태.
Post-Conditions	난수를 0으로 초기화하여 사용자에게 보여준다.

- 0으로 초기화 → 1로 초기화

Team 7 Specification Review

Stage 2050 Implementation

1.15 Set Global Time(R.6.0)

Name	Set Global Time
Responsibilities	Global time을 조정한다.
Type	UI
Cross References	
Note	버튼을 입력하여, Global Time을 15분씩 혹은 1시간씩 조정한다. default 시간은 현재 시간과 동일하다.
Pre-Conditions	Global Time 모드 상태.
Post-Conditions	조정된 Global time을 사용자에게 보여준다.

- 시간을 조정할 때 15분/ 1시간 씩 증가 X
→ 1시간, 2시간, 3시간/ 15분, 30분, 45분 형식으로 증가

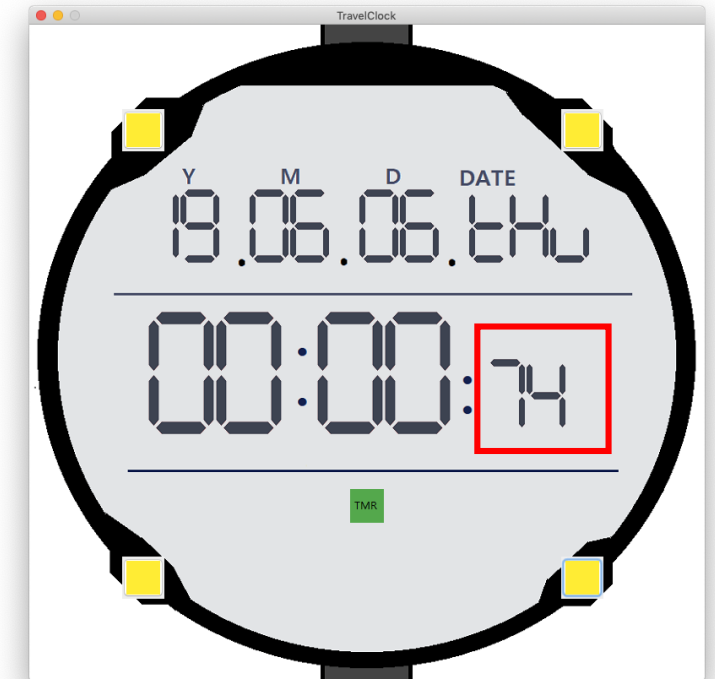
Team 7 Specification Review

Stage 2050 Implementation

1.3 Start Timer(R.2.1)

Name	StartTimer
Responsibilities	타이머 시간이 줄어들기 시작한다.
Type	UI
Cross References	R.2.0, R.2.2
Note	버튼이 입력되면 타이머에 저장되어있던 시간이 줄어들기 시작한다.
Pre-Conditions	타이머 모드 상태
Post-Conditions	타이머 시간이 1초마다 줄어 들고, 사용자에게 보여준다.

- 00: 00: 00 인 상태에서 시작 버튼을 누르면 멍통



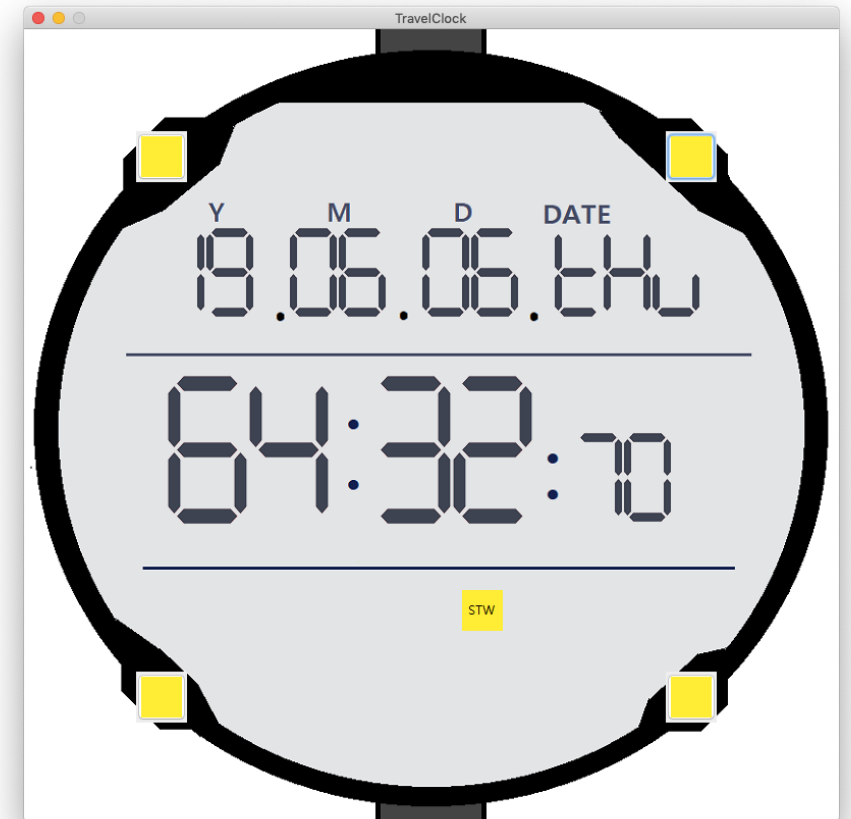
- 알 수 없는 오류..

Team 7 Specification Review

Stage 2050 Implementation

1.5 Start Stopwatch(R.3.0)

Name	StartStopwatch
Responsibilities	스톱워치 시간이 증가하기 시작한다.
Type	UI
Cross References	
Note	버튼이 입력되면 스톱워치 시간이 최대 시간 1시간 59분 59초까지 증가한다.
Pre-Conditions	스톱워치 모드 상태.
Post-Conditions	스톱워치 시간이 증가하고, 사용자에게 보여준다.



- 59분에서 1시간으로 넘어가지 않음

| Team 7 Specification Review

Summary

- **작동 방법이 복잡하여 매뉴얼을 보고도 익히는데 한참 걸림**
- **프로그램이 적어도 5분 안에 먹통이 됨**
- **이전에 지적했던 부분에서 개선되지 않은 부분이 다수 존재**



2nd Specification Review
- Team 8 -



| Team 8 Specification Review

Stage 1000 Planning

<Activity 1003. Define Requirements>

- Activity 1003. Define Requirements
 - 2. Performance Requirements
 - 간단하고 직관적인 기능 인터페이스를 제공해야 한다.
- => 메뉴얼을 봐야 이해가 된다. 각 버튼의 이름이 각 버튼의 기능을 설명해주지 않는다.
ex) START 버튼이 너무 많은 역할을 한다.

대응방안->

- 2. Performance Requirements
 - 타 디지털 시계와 비교하여 비슷한 기능 인터페이스를 제공해야 한다
 - 비전문가도 사용하기 쉬워야 한다
 - 시간이나 밀물 썰물의 오차를 최소화해야 한다

=> 실제 시계의 버튼을 그대로 차용하여 기능 인터페이스를 구성하였기 때문에 보고서 내용을 수정하였다

수정사항 확인 완료!

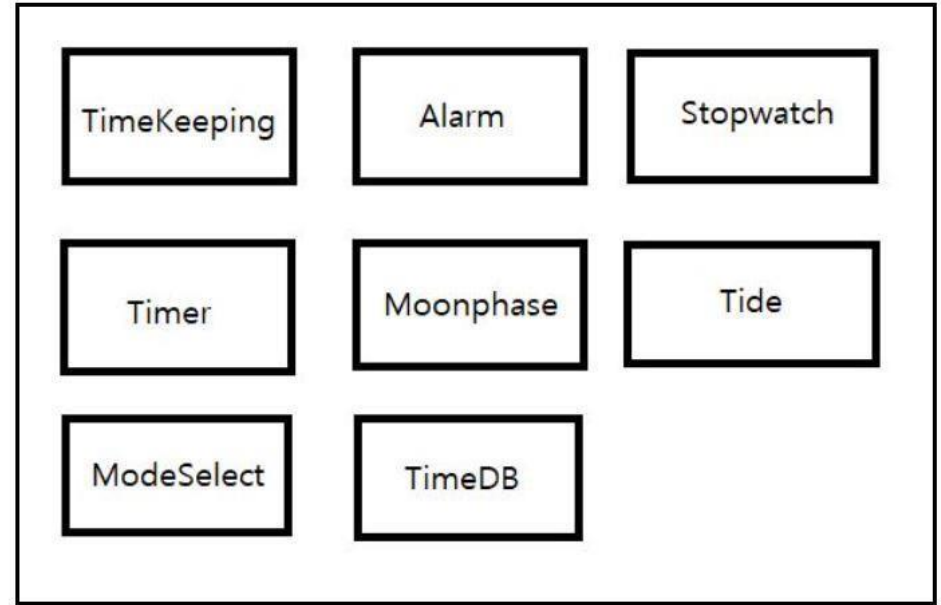
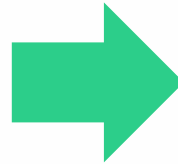
Team 8 Specification Review

Stage 1000 Planning

- Activity 1008. Define Business Concept Models



- => 해당 Business Concept Model이 시스템에 반영되어 있지 않다.
- => Timer, Alarm, Stopwatch, Moonphase 등이 되어야 한다.



=> Business Concept Model이 System에 반영이 되도록 보고서 내용을 수정하였다

수정사항 확인 완료!

Team 8 Specification Review

Stage 2030 Analysis - Activity 2031. Define Essential Use Case

Use case 2. adjustTime

Typical Courses of Events	(A): Actor / (S) : System	(A): Actor / (S) : System
	<ol style="list-style-type: none">1. (A) 버튼을 눌러 시간 조정 모드로 바꾼다2. (S) TimeDB에서 가져온 현재 시각을 화면에 표시해주고 연도 부분이 조정 가능한 상태임을 표시한다.3. (A) 사용자가 연도를 수정하고 다음 버튼을 누른다4. (S) 현재 수정가능상태를 월로 바꿔준다5. (A) 사용자가 월을 수정하고 다음 버튼을 누른다6. (S) 현재 수정가능상태를 일로 바꿔준다.7. (A) 사용자가 일을 수정하고 다음 버튼을 누른다8. (S) 현재 수정가능상태를 시로 바꿔준다9. (A) 사용자가 시를 수정하고 다음 버튼을 누른다10. (S) 현재 수정가능상태를 분으로 바꿔준다11. (A) 사용자가 분을 수정하고 설정을 종료한다12. (S) 초를 00으로 초기화한다13. (S) 조정된 시간을 TimeDB에 저장한다. 이 때, 초는 00으로 초기화한다.	<ol style="list-style-type: none">1. (A) 버튼을 눌러 시간 조정 모드로 바꾼다2. (S) TimeDB에서 가져온 현재 시각을 화면에 표시해주고 연도 부분이 조정 가능한 상태임을 표시한다.3. (A) 사용자가 연도를 수정하거나 하지 않고 다음 버튼을 누른다4. (S) 현재 수정가능상태를 월로 바꿔준다5. (A) 사용자가 월을 수정하거나 하지 않고 다음 버튼을 누른다6. (S) 현재 수정가능상태를 일로 바꿔준다.7. (A) 사용자가 일을 수정하거나 하지 않고 다음 버튼을 누른다8. (S) 현재 수정가능상태를 시로 바꿔준다9. (A) 사용자가 시를 수정하거나 하지 않고 다음 버튼을 누른다10. (S) 현재 수정가능상태를 분으로 바꿔준다11. (A) 사용자가 분을 수정하거나 하지 않고 설정을 종료한다12. (S) 초를 00으로 초기화한다13. (S) 조정된 시간을 TimeDB에 저장한다. 이 때, 초는 00으로 초기화한다.

수정사항 확인 완료!

Team 8 Specification Review

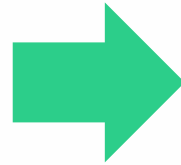
Stage 2030 Analysis - Activity 2031. Define Essential Use Case

Use case 4. setTimer의 Typical Courses of Events

2. (S) 타이머에서 이전에 설정한 타이머의 시간을 가져온다.

Typical Courses of Events	(A) Actor / (S) System
	1. (A) 버튼을 눌러 타이머 설정 가능 상태로 바꾼다.
	2. (S) 타이머에서 이전에 설정한 타이머의 시간을 가져온다
	3. (S) 현재 타이머가 동작중인 상태일 경우 타이머를 일시정지 시킨다
	4. (S) 현재 수정가능상태를 시로 표시해준다
	5. (A) 사용자가 버튼을 눌러 시를 설정한다
	6. (S) 현재 수정가능상태를 분으로 바꿔준다
	7. (A) 사용자가 버튼을 눌러 분을 설정한다
	8. (S) 현재 수정가능상태를 초로 바꿔준다
	9. (A) 사용자가 버튼을 눌러 초를 바꾸고 설정을 마친다
	10. (S) 조정된 시간을 저장한다

=> 이전에 설정한 타이머의 시간이 없을 때 어떠한 처리를 수행하는지 나와있지 않다.



Typical Courses of Events

(A) Actor / (S) System

1. (A) 버튼을 눌러 타이머 설정 가능 상태로 바꾼다.
2. (S) 현재 타이머가 동작중인 상태일 경우 타이머를 일시정지 시킨다
3. (S) 버튼을 누르기 전 화면에 표시되던 타이머의 시간을 화면에 보여준다.
4. (S) 현재 수정가능상태를 시로 표시해준다
5. (A) 사용자가 버튼을 눌러 시를 설정하거나 하지 않고 다음 버튼을 누른다.
6. (S) 현재 수정가능상태를 분으로 바꿔준다
7. (A) 사용자가 버튼을 눌러 분을 설정하거나 하지 않고 다음 버튼을 누른다.
8. (S) 현재 수정가능상태를 초로 바꿔준다
9. (A) 사용자가 버튼을 눌러 초를 바꾸거나 그러지 않고 버튼을 눌러 설정을 마친다
10. (S) 조정된 시간을 저장한다

=> '이전에 설정한 타이머' 라는 말이 의도한 바와 달라 수정했다. 본래 00:00:00을 포함해 저장되어 있던 타이머의 시간을 뜻하는 것이었으나 표현이 부정확했다. 더불어 동작 중일 때를 고려하면 '설정'이라는 표현이 사용자가 설정한 본래의 시간과 타이머가 동작함에 따라 줄어든 시간 중 가리키는 바가 애매할 수 있다고 생각해 정확하게 표현을 수정하고 그에 따라 이벤트 순서도 더 정확하게 수정했다.

수정사항 확인 완료!

Team 8 Specification Review

Stage 2030 Analysis - Activity 2031. Define Essential Use Case

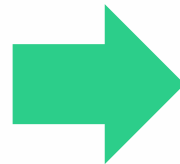
Use case 25. modeSelect의 Typical Courses of Events

Typical Courses of Events	(A): Actor / (S) : System
	1.(A) User가 버튼을 눌러 mode select로 진입한다
	2.(S) mode select 설정 화면을 표시한다
	3.(A) User가 선택 버튼 을 눌러 TimeKeeping 모드를 선택하거나 선택 해제한다.
	4.(A) User가 다음 버튼 을 눌러 Timer 모드로 화면을 넘긴다.
	5.(A) User가 선택 버튼 을 눌러 Timer 모드를 선택하거나 선택 해제한다.
	6.(A) User가 다음 버튼 을 눌러 Alarm 모드로 화면을 넘긴다.
	7.(A) User가 선택 버튼 을 눌러 Alarm 모드를 선택하거나 선택 해제한다.
	8.(A) User가 다음 버튼 을 눌러 Stopwatch모드로 화면을 넘긴다.
	9.(A) User가 선택 버튼 을 눌러 Stopwatch 모드를 선택하거나 선택 해제한다.
	10.(A) User가 다음 버튼 을 눌러 Tide 모드로 화면을 넘긴다.
	11.(A) User가 선택 버튼 을 눌러 Tide 모드를 선택하거나 선택 해제한다.
	12.(A) User가 다음 버튼 을 눌러 Moonphase 모드로 화면을 넘긴다.
	13.(A) User가 선택 버튼 을 눌러 Moonphase 모드를 선택하거나 선택 해제한다.
	14.(A) User가 확인 버튼 을 눌러 설정을 완료한다.
	15.(S) 모드 목록을 갱신한다.

3.(A), 5.(A), 7(A), 9(A), 11(A), 13(A) 선택 버튼을 눌러
=> 선택 버튼에서 START버튼으로 수정해야 한다.

4(A), 6(A), 8(A), 10(A), 12(A) 다음 버튼을 눌러
=> 다음 버튼에서 MODE버튼으로 수정해야 한다.

14.(A) 확인 버튼을 눌러
=> 확인 버튼에서 ADJUST버튼으로 수정해야 한다.



Typical Courses of Events

(A): Actor / (S) : System
1.(A) User가 버튼을 눌러 mode select로 진입한다
2.(S) mode select 설정 화면을 표시한다
3.(A) User가 버튼을 눌러 TimeKeeping 모드를 선택하거나 선택 해제한다.
4.(A) User가 버튼을 눌러 Timer 모드로 화면을 넘긴다.
5.(A) User가 버튼을 눌러 Timer 모드를 선택하거나 선택 해제한다.
6.(A) User가 버튼을 눌러 Alarm 모드로 화면을 넘긴다.
7.(A) User가 버튼을 눌러 Alarm 모드를 선택하거나 선택 해제한다.
8.(A) User가 버튼을 눌러 Stopwatch모드로 화면을 넘긴다.
9.(A) User가 버튼을 눌러 Stopwatch 모드를 선택하거나 선택 해제한다.
10.(A) User가 버튼을 눌러 Tide 모드로 화면을 넘긴다.
11.(A) User가 버튼을 눌러 Tide 모드를 선택하거나 선택 해제한다.
12.(A) User가 버튼을 눌러 Moonphase 모드로 화면을 넘긴다.
13.(A) User가 버튼을 눌러 Moonphase 모드를 선택하거나 선택 해제한다.
14.(A) User가 버튼을 눌러 설정을 완료한다.
15.(S) 모드 목록을 갱신한다.

=> 2030은 정확한 버튼 할당이 되어 있지 않은 상태의 보고서인데 나중에 수정하다가 버튼을 나눈 정보가 의도치 않게 들어가 수정하였다.

수정사항 확인 완료!

Team 8 Specification Review

Stage 2030 Analysis

Activity 2031. Define Essential Use Case Use case 12. deleteAlarm

Exceptional Courses of Events	E2. 알람목록에 알람이 0개 이하일 경우 에러메세지 를 출력한다
-------------------------------	---

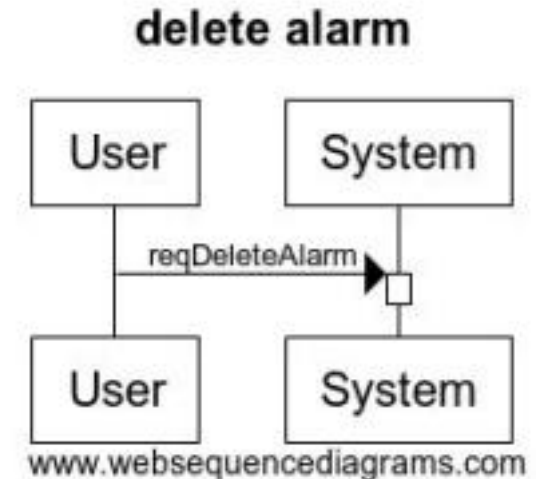


Exceptional Courses of Events	N/A
-------------------------------	-----

=> 이전버전에 대한 usecase를 수정할 때에 놓쳐 수정하지 못한 부분으로, sequence diagram은 최신에 맞게 올바르게 되어있는 것이고, usecase를 수정해주었다.

Activity 2035. Define System Sequence Diagrams

delete alarm



수정사항 확인 완료!

Team 8 Specification Review

Stage 2040 Design - Activity 2041. Define Real Use Cases

Use case 25. modeSelect

Typical Courses of Events	(A): Actor / (S) : System
	1.(A) User가 mode버튼을 4번 눌러 mode select로 진입한다 reset 버튼!
	2.(S) mode select 설정 화면을 표시한다
	3.(A) User가 start 버튼을 눌러 TimeKeeping 모드를 선택하거나 선택 해제한다.
	4.(A) User가 mode 버튼을 눌러 Timer 모드로 화면을 넘긴다.
	5.(A) User가 start 버튼을 눌러 Timer 모드를 선택하거나 선택 해제한다.

- 2030단계에는 버튼을 3초간 눌러->4번 눌러로 잘 반영되었는데, 버튼이 명시된 2040단계에서 버튼 이름이 잘못 되었다.

재수정 필요!

| Team 8 Specification Review

Stage 2040 Design - Activity 2042. Define Reports, UI, and Storyboards



누락되었던 modeSelect 화면 추가 확인 완료!

| Team 8 Specification Review

Summary

- 7조와 비교하였을 때, 상당히 **예외처리**를 잘함
- 구현에 맞게 **요구사항을 바꾼 부분이 다수 존재**

2

2nd System Test

- Team 7 Category-Partition Test
- Team 8 Category-Partition Test
- Team 7 Pairwise Test
- Team 8 Pairwise Test
- Team 7 Brute Force Test
- Team 8 Brute Force Test



Category-Partition Test
-Team 7



| Team 7 1st Category-Partition Test

Group	Category	Value	Constraints	
mode	mode	time	[property T]	
		set time	[property ST]	
		timer	[property TMR]	
		stopwatch	property STW]	
		alarm	[property ALM]	
		random number	[property RDM]	[error]
		global time	[property GMT]	[error]
		set global time	[property SGMT]	[error]
		select mode	[property SM]	[error]
		input	input	left top short time
left top long time				
left bottom short time				
left bottom long time				
right top short time				
right top long time				
right bottom short time				
right bottom long time				
data	data	valid year		
		invalid year	[error]	
		valid month		
		invalid month	[error]	
		valid day		
		invalid day	[error]	
		valid date		
		invalide date	[error]	
		valid hour		
		invalid hour	[error]	
		valid minute		
		invalid minute	[error]	
		valid second		
		invalid second	[error]	
valid millisecond				
invalid second	[error]			
performance	environment	common	[single]	
		sleep(절전모드)		

[error property] 2016 → 571
(71.68% 감소)

[single property] 571 → 292
(48.87% 감소)

[if property] 292 → 209
(28.43% 감소)

[1st Category – Partition test rate]

162/209 x 100 = 77.5%

Team 7 2nd Category-Partition Test

Group	Category	Value	Constraints		
mode	mode	time	[property T]		
		set time	[property ST]		
		timer	[property TMR]		
		stopwatch	property STW]		
		alarm	[property ALM]		
		random number	[property RNG]		
		global time	[property GMT]		
		select mode	[property SM]		
input	input	left top short time			
		left top long time			
		left bottom short time			
		left bottom long time			
		right top short time			
		right top long time			
		right bottom short time			
		right bottom long time			
data	data	valid year	[single]		
		invalid year	[error]		
		valid month	[single]		
		invalid month	[error]		
		valid day	[single]		
		invalid day	[error]		
		valid date	[single]		
		invalid date	[error]		
		valid hour	[single]		
		invalid hour	[error]		
		valid minute	[single]		
		invalid minute	[error]		
		valid second	[single]		
		invalid second	[error]		
		performance	environment	common	[single]
				sleep(절전모드)	

```

mode:
  time.                [property T]
  set time.            [property ST]
  timer.              [property TMR]
  stopwatch.          [property STW]
  alarm.              [property ALM]
  random number.     [property RNG]
  global time.       [property GMT]
  set mode.          [property SM]

input:
  left top short time. [if (T || ST || TMR || STW || ALM || RNG || GMT || SM)]
  left top long time. [if (TMR || ALM || GMT)]
  left bottom short time. [if (ST || TMR || ALM || RNG || GMT)]
  left bottom long time. [if (TMR || ALM || GMT)]
  right top short time. [if (ST || TMR || STW || ALM || RNG || SM)]
  right top long time. [if (ST || TMR || ALM)]
  right bottom short time. [if (T || ST || TMR || STW || ALM || RNG || GMT || SM)]
  right bottom long time. [if (T || ST || TMR || STW || ALM || RNG || GMT || SM)]

data:
  valid year.          [single]
  invalid year.        [error]
  valid month.         [single]
  invalid month.       [error]
  valid day.           [single]
  invalid day.         [error]
  valid date.          [single]
  invalid date.        [error]
  valid hour.          [single]
  invalid hour.        [error]
  valid minute.        [single]
  invalid minute.      [error]
  valid second.        [single]
  invalid second.      [error]

performance:
  environment:
    common.            [single]
    sleep.             [if (T || GMT)]
  
```

- 7조에서 1차 때 error constraints였던 random number, global time, set global time, set mode부분을 구현완료하여 error constraints가 아닌 것으로 수정하였다.

Team 7 2nd Category-Partition Test

Number	mode	input	data	environment	Key	Result
1			valid year		<single>	P
2			invalid year		<error>	P
3			valid month		<single>	P
4			invalid month		<error>	P
5			valid day		<single>	P
6			invalid day		<error>	P
7			valid date		<single>	P
8			invalid date		<error>	P
9			valid hour		<single>	P
10			invalid hour		<error>	P
11			valid minute		<single>	P
12			invalid minute		<error>	P
13			valid second		<single>	P
14			invalid second		<error>	P
15				common	<single>	P
16	time	left top short time		sleep	1.1.0.2	P
17	time	right bottom short time		sleep	1.7.0.2	P
18	time	right bottom long time		sleep	1.8.0.2	P
19	time	left top short time			2.1.0.0	P
20	set time	left bottom short time			2.3.0.0	P
21	set time	right top short time			2.5.0.0	P
22	set time	right top long time			2.6.0.0	F
23	set time	right bottom short time			2.7.0.0	P
24	set time	right bottom long time			2.8.0.0	P
25	timer	left top short time			3.1.0.0	P
26	timer	left top long time			3.2.0.0	P
27	timer	left bottom short time			3.3.0.0	P
28	timer	left bottom long time			3.4.0.0	P
29	timer	right top short time			3.5.0.0	P
30	timer	right top long time			3.6.0.0	P
31	timer	right bottom short time			3.7.0.0	P
32	timer	right bottom long time			3.8.0.0	P
33		stopwatch	left top short time			4.1.0.0 P
34		stopwatch	right top short time			4.5.0.0 P
35		stopwatch	right bottom short time			4.7.0.0 P
36		stopwatch	right bottom long time			4.8.0.0 P
37		alarm	left top short time			5.1.0.0 P
38		alarm	left top long time			5.2.0.0 P
39		alarm	left bottom short time			5.3.0.0 P
40		alarm	left bottom long time			5.4.0.0 P
41		alarm	right top short time			5.5.0.0 P
42		alarm	right top long time			5.6.0.0 P
43		alarm	right bottom short time			5.7.0.0 P
44		alarm	right bottom long time			5.8.0.0 P
45	random number	left top short time				6.1.0.0 P
46	random number	left bottom short time				6.3.0.0 P
47	random number	right top short time				6.5.0.0 P
48	random number	right bottom short time				6.7.0.0 P
49	random number	right bottom long time				6.8.0.0 P
50	global time	left top short time		sleep		7.1.0.2 F
51	global time	left top long time		sleep		7.2.0.2 F
52	global time	left bottom short time		sleep		7.3.0.2 F
53	global time	left bottom long time		sleep		7.4.0.2 F
54	global time	right bottom short time				7.7.0.2 P
55	global time	right bottom long time				7.8.0.2 P
56	set mode	left top short time				8.1.0.0 P
57	set mode	right top short time				8.5.0.0 F
58	set mode	right bottom short time				8.7.0.0 P
59	set mode	right bottom long time				8.8.0.0 P

[2nd Category – Partition test rate]

$$53/59 \times 100 = 89.8\%$$



Category-Partition Test
-Team 8



| Team 8 1st Category-Partition Test

Group	Category	Value	Constraints
mode	mode	timekeeping	[property TK]
		timesetting	[property TKS]
		timer	[property TM]
		timersetting	[property TMS]
		alarm	[property AM]
		alarmsetting	[property AMS]
		stopwatch	[property SW]
		tide	[property TD]
		moonphase	[property MP]
		mode select	[property MDS]
performance	environment	sleep	[if (TK TM SW TD MP)]
		common	
input	button type	none	[single]
		adjust	ref
		start	ref
		reset	ref
		mode	ref
		multi	[error]
	push	0	[single]
		1	[if (AD RS)]
		4	[if (MD ST)]
	data	invalid minute	[error]
		valid minute	[if (TKS AMS TMS TD MP)]
		invalid hour	[error]
		valid hour	[if (TKS AMS TMS TD MP)]
		invalid date	[error]
		valid date	[if (TK MP TD)]
		invalid month	[error]
		valid month	[if (TK MP TD)]
		invalid year	[error]
		valid year	[if (TK MP TD)]
		eclipse	[if MP]
		invalid	[error]
		east sea	[if TD]
		west sea	[if TD]
south sea	[if TD]		

[error property] 5400 → 2166
(59.89% 감소)

[single property] 2166 → 1447
(34.21% 감소)

[if property] 1447 → 95
(93.35% 감소)

[1st Category – Partition test rate]

$$81/95 \times 100 = 85.26\%$$

Team 8 2nd Category-Partition Test

Group	Category	Value	Constraints	number
mode	mode	timekeeping	[property TK]	1001
		timesetting	[property TKS]	1002
		timer	[property TM]	1003
		timersetting	[property TMS]	1004
		alarm	[property AM]	1005
		alarmsetting	[property AMS]	1006
		stopwatch	[property SW]	1007
		tide	[property TD]	1008
		moonphase	[property MP]	1009
		mode select	[property MDS]	1010
performance	environment	sleep	[if (TK TM SW)]	2001
		common		2002
input	button type	none	[single]	3001
		adjust	ref	3002
		start	ref	3003
		reset	ref	3004
		mode	ref	3005
		multi	[error]	3006
	push	0	[single]	3101
		1	[if (AD RS)]	3102
		4	[if (MD ST)]	3103
	data	invalid minute	[error]	3201
		valid minute	[if (TKS AMS TMS)]	3202
		invalid hour	[error]	3203
		valid hour	[if (TKS AMS TMS TD MP)]	3204
		invalid date	[error]	3205
		valid date	[if (TK MP TD)]	3206
		invalid month	[error]	3207
		valid month	[if (TK MP TD)]	3208
		invalid year	[error]	3209
		valid year	[if (TK MP TD)]	3210
		invalid sea	[error]	3211
		east sea	[if TD]	3212
west sea	[if TD]	3213		
south sea	[if TD]	3214		

- Sleep의 경우 시간이 흐르는 mode로 한정함
- Eclipse의 경우 requirement 에 맞지 않다고 판단
 - 제거
- Sleep의 경우 requirement 에 없었으나
 - 기본으로 충족해야 하는 조건이라 생각하여 변경 X
 - 7조의 경우 requirement 에 없었음에도 불구하고, 해당 조건을 충족함
 - ex) 스마트폰에 탑재되는 clock일 경우, sleep mode에 두었다가 다시 켜를 때 시간이 제대로 흐르지 않음
 - ⇒ 과연 이러한 경우에도 requirement 에 없었으니 fail이 나와도 상관없다고 할 수 있을지 의문점이 생김.
- Valid minute의 경우 tide, moonphase mode에서 제외함

Team 8 2nd Category-Partition Test

Group	Category	Value	Constraints	number
mode	mode	timekeeping	[property TK]	1001
		timesetting	[property TKS]	1002
		timer	[property TM]	1003
		timersetting	[property TMS]	1004
		alarm	[property AM]	1005
		alarmsetting	[property AMS]	1006
		stopwatch	[property SW]	1007
		tide	[property TD]	1008
		moonphase	[property MP]	1009
		mode select	[property MDS]	1010
performance	environment	sleep	[if (TK TM SW)]	2001
		common		2002
input	button type	none	[single]	3001
		adjust	ref	3002
		start	ref	3003
		reset	ref	3004
		mode	ref	3005
		multi	[error]	3006
	push	0	[single]	3101
		1	[if (AD RS)]	3102
		4	[if (MD ST)]	3103
	data	invalid minute	[error]	3201
		valid minute	[if (TKS AMS TMS)]	3202
		invalid hour	[error]	3203
		valid hour	[if (TKS AMS TMS TD MP)]	3204
		invalid date	[error]	3205
		valid date	[if (TK MP TD)]	3206
		invalid month	[error]	3207
		valid month	[if (TK MP TD)]	3208
		invalid year	[error]	3209
		valid year	[if (TK MP TD)]	3210
		invalid sea	[error]	3211
		east sea	[if TD]	3212
		west sea	[if TD]	3213
		south sea	[if TD]	3214

```

mode:
  mode:
    timekeeping. [property TK]
    timesetting. [property TKS]
    timer. [property TM]
    timersetting. [property TMS]
    alarm. [property AM]
    alarmsetting. [property AMS]
    stopwatch. [property SW]
    tide. [property TD]
    moonphase. [property MP]
    mode select. [property MDS]

performance:
  environment:
    sleep. [if (TK || TM || SW)]
    common.

input:
  button type:
    none. [single]
    adjust. [if (TK || TKS || TM || TMS || AM || AMS || SW || MDS)][property AD]
    start. [if (TKS || TM || TMS || AM || AMS || SW || TD || MDS)][property ST]
    reset. [if (TK || TM || AM || SW || TD || MP)][property RS]
    mode. [if (TK || TKS || TM || TMS || AM || AMS || SW || TD || MP || MDS)][property MD]
    multi. [error]

  push:
    0. [single]
    1. [if (AD || RS)]
    4 <=. [if (MD || ST)]

  data:
    invalid minute. [error]
    invalid hour. [error]
    valid minute. [if (TKS || AMS || TMS)]
    valid hour. [if (TKS || AMS || TMS || TD || MP)]
    invalid date. [error]
    invalid month. [error]
    invalid year. [error]
    valid date. [if (TK || MP || TD)]
    valid month. [if (TK || MP || TD)]
    valid year. [if (TK || MP || TD)]
    invalid sea. [error]
    east sea. [if TD]
    west sea. [if TD]
    south sea. [if TD]
  
```

[error property] 5040 → 2407
(52.25% 감소)

[single property] 2166 → 1289
(40.49% 감소)

[if property] 1289 → 97
(92.48% 감소)

Team 8 2nd Category-Partition Test

case	number	result							
1	3001	T	37	1003, 2001, 3005, 3103, 3200	F	73	1008, 2002, 3004, 3102, 3204	T	
2	3006	T	38	1003, 2002, 3002, 3102, 3200	T	74	1008, 2002, 3004, 3102, 3208	T	
3	3101	T	39	1003, 2002, 3003, 3103, 3200	T	75	1008, 2002, 3004, 3102, 3209	T	
4	3201	T	40	1003, 2002, 3004, 3102, 3200	T	76	1008, 2002, 3004, 3102, 3210	T	
5	3203	T	41	1003, 2002, 3005, 3103, 3200	T	77	1008, 2002, 3004, 3102, 3212	T	
6	3205	T	42	1004, 2002, 3002, 3102, 3203	T	78	1008, 2002, 3004, 3102, 3213	T	
7	3207	T	43	1004, 2002, 3002, 3102, 3204	T	79	1008, 2002, 3004, 3102, 3214	T	
8	3209	T	44	1004, 2002, 3003, 3103, 3203	T	80	1008, 2002, 3005, 3103, 3204	T	
9	3211	T	45	1004, 2002, 3003, 3103, 3204	T	81	1008, 2002, 3005, 3103, 3208	T	
0	1001, 2001, 3002, 3102, 3208	T	46	1004, 2002, 3005, 3103, 3203	T	82	1008, 2002, 3005, 3103, 3209	T	
11	1001, 2001, 3002, 3102, 3209	T	47	1004, 2002, 3005, 3103, 3204	T	83	1008, 2002, 3005, 3103, 3210	T	
12	1001, 2001, 3002, 3102, 3210	T	48	1005, 2002, 3002, 3102, 3200	T	84	1008, 2002, 3005, 3103, 3212	T	
13	1001, 2001, 3004, 3102, 3208	F	49	1005, 2002, 3003, 3103, 3200	T	85	1008, 2002, 3005, 3103, 3213	T	
14	1001, 2001, 3004, 3102, 3209	F	50	1005, 2002, 3004, 3102, 3200	T	86	1008, 2002, 3005, 3103, 3214	T	
15	1001, 2001, 3004, 3102, 3210	F	51	1005, 2002, 3005, 3103, 3200	T	87	1009, 2002, 3004, 3102, 3204	T	
16	1001, 2001, 3005, 3103, 3208	F	52	1006, 2002, 3002, 3102, 3203	T	88	1009, 2002, 3004, 3102, 3208	T	
17	1001, 2001, 3005, 3103, 3209	F	53	1006, 2002, 3002, 3102, 3204	T	89	1009, 2002, 3004, 3102, 3209	T	
18	1001, 2001, 3005, 3103, 3210	F	54	1006, 2002, 3003, 3103, 3203	T	90	1009, 2002, 3004, 3102, 3210	T	
19	1001, 2002, 3002, 3102, 3208	T	55	1006, 2002, 3003, 3103, 3204	T	91	1009, 2002, 3005, 3103, 3204	T	
20	1001, 2002, 3002, 3102, 3209	T	56	1006, 2002, 3005, 3103, 3203	T	92	1009, 2002, 3005, 3103, 3208	T	
21	1001, 2002, 3002, 3102, 3210	T	57	1006, 2002, 3005, 3103, 3204	T	93	1009, 2002, 3005, 3103, 3209	T	
22	1001, 2002, 3004, 3102, 3208	T	58	1007, 2001, 3002, 3102, 3200	F	94	1009, 2002, 3005, 3103, 3210	T	
23	1001, 2002, 3004, 3102, 3209	T	59	1007, 2001, 3003, 3103, 3200	F	95	1010, 2002, 3002, 3102, 3200	T	
24	1001, 2002, 3004, 3102, 3210	T	60	1007, 2001, 3004, 3102, 3200	T	96	1010, 2002, 3003, 3103, 3200	T	
25	1001, 2002, 3005, 3103, 3208	T	61	1007, 2001, 3005, 3103, 3200	F	97	1010, 2002, 3005, 3102, 3200	T	
26	1001, 2002, 3005, 3103, 3209	T	62	1007, 2002, 3002, 3102, 3200	T				
27	1001, 2002, 3005, 3103, 3210	T	63	1007, 2002, 3003, 3103, 3200	T				
28	1002, 2002, 3002, 3102, 3203	T	64	1007, 2002, 3004, 3102, 3200	T				
29	1002, 2002, 3002, 3102, 3204	T	65	1007, 2002, 3005, 3103, 3200	T				
30	1002, 2002, 3003, 3103, 3203	T	66	1008, 2002, 3003, 3103, 3204	T				
31	1002, 2002, 3003, 3103, 3204	T	67	1008, 2002, 3003, 3103, 3208	T				
32	1002, 2002, 3005, 3103, 3203	T	68	1008, 2002, 3003, 3103, 3209	T				
33	1002, 2002, 3005, 3103, 3204	T	69	1008, 2002, 3003, 3103, 3210	T				
34	1003, 2001, 3002, 3102, 3200	T	70	1008, 2002, 3003, 3103, 3212	T				
35	1003, 2001, 3003, 3103, 3200	F	71	1008, 2002, 3003, 3103, 3213	T				
36	1003, 2001, 3004, 3102, 3200	T	72	1008, 2002, 3003, 3103, 3214	T				

[2nd Category – Partition test rate]

$$86/97 \times 100 = 88.65\%$$



Pairwise Test
-Team 7



Team 7 1st Pairwise test

number	mode	input	data	result
1	alarm	left bottom long time	valid	P
2	alarm	left bottom short time	valid	P
3	alarm	right bottom long time	valid	P
4	alarm	right top short time	valid	P
5	alarm	left top short time	valid	P
6	alarm	left top long time	valid	P
7	alarm	right top long time	valid	F
8	alarm	right bottom short time	valid	P
9	global time	right bottom short time	none	P
10	global time	right bottom long time	none	P
11	set time	right top long time	valid	F
12	set time	right bottom long time	valid	P
13	set time	right bottom short time	valid	P
14	set time	left top short time	valid	P
15	set time	left bottom short time	valid	P
16	stopwatch	right top short time	none	P
17	stopwatch	right bottom long time	none	P
18	stopwatch	left top short time	none	F
19	stopwatch	right bottom short time	none	P
20	time	right bottom short time	none	P
21	time	left top short time	none	P
22	timer	right bottom long time	valid	P
23	timer	left bottom long time	valid	P
24	timer	right bottom short time	valid	P
25	timer	right top short time	valid	P
26	timer	left bottom short time	valid	P
27	timer	left top long time	valid	P
28	timer	left top short time	valid	P
29	timer	right top long time	valid	F

```

1 mode: time, set time, timer, stopwatch, alarm, global time
2
3 input: left top short time, left top long time, left bottom short time, left bottom long time, right top short
time, right top long time, right bottom short time, right bottom long time
4
5 data: valid, none
6
7 IF [mode] = "time" THEN [input] IN {"left top short time", "right bottom short time", "right bottom long time"};
8 IF [mode] = "set time" THEN [input] IN {"left top short time", "left bottom short time", "right top short time",
"right top long time", "right bottom short time", "right bottom long time"};
9 IF [mode] = "stopwatch" THEN [input] IN {"left top short time", "right top short time", "right bottom short
time", "right bottom long time"};
10 IF [mode] = "global time" THEN [input] = "right bottom short time" OR [input] = "right bottom long time";
11
12 IF [mode] IN {"time", "stopwatch", "global time"} THEN [data] = "none";
13 IF [mode] IN {"set time", "timer", "alarm"} THEN [data] = "valid";

```

[1st Pairwise pass rate]

→ $25/29 \times 100 = 86.2\%$

Team 7 2nd Pairwise test

number	mode	input	result
1	alarm	right top long time	P
2	alarm	left top long time	P
3	alarm	right top short time	P
4	alarm	left bottom short time	P
5	alarm	left top short time	P
6	alarm	left bottom long time	P
7	global time	left bottom long time	F
8	global time	left top short time	F
9	global time	left top long time	F
10	global time	left bottom short time	F
11	random number	left top short time	P
12	random number	left bottom short time	P
13	random number	right top short time	P
14	set mode	right top short time	F
15	set mode	right bottom short time	P
16	set mode	left top short time	P
17	set mode	right bottom long time	P
18	set time	right top short time	P
19	set time	right top long time	F
20	set time	left top short time	P
21	set time	left bottom short time	P
22	stopwatch	right top short time	P
23	stopwatch	left top short time	P
24	time	left top short time	P
25	timer	left top long time	P
26	timer	left bottom short time	P
27	timer	right top short time	P
28	timer	left bottom long time	P
29	timer	right top long time	P
30	timer	left top short time	P

```
1 mode: time, set time, timer, stopwatch, alarm, random number, global time, set mode
2
3 input: left top short time, left top long time, left bottom short time, left bottom long
   time, right top short time, right top long time, right bottom short time, right bottom
   long time
4
5 IF [input] = "right bottom short time" THEN [mode] = "set mode";
6 IF [input] = "right bottom long time" THEN [mode] = "set mode";
7
8 IF [mode] = "time" THEN [input] IN {"left top short time"};
9 IF [mode] = "set time" THEN [input] IN {"left top short time", "left bottom short time",
   "right top short time", "right top long time"};
10 IF [mode] = "stopwatch" THEN [input] IN {"left top short time", "right top short time"};
11 IF [mode] = "random number" THEN [input] IN {"left top short time", "left bottom short
   time", "right top short time"};
12 IF [mode] = "global time" THEN [input] IN {"left top short time", "left top long time",
   "left bottom short time", "left bottom long time"};
13 IF [mode] = "set mode" THEN [input] IN {"left top short time", "right top short time",
   "right bottom short time", "right bottom long time"};
```

[2nd Pairwise pass rate]

$$\rightarrow 24/30 \times 100 = 80\%$$



Pairwise Test
-Team 8



| Team 8 1st Pairwise test

environment	data	HW button	mode
sleep	time	start	tide
common	date	adjust	moonphase
-	sea	mode	timerunning
-	-	reset	setting

	environment	Data	HW button	mode	result
1	sleep	time	start	tide	P
2	sleep	date	adjust	moonphase	P
3	sleep	-	reset	setting	P
4	common	date	mode	setting	P
5	common	sea	reset	tide	P
6	common	-	start	moonphase	P
7	common	time	adjust	timerunning	P
8	-	-	adjust	setting	P
9	-	time	mode	tide	P
10	-	date	reset	moonphase	F
11	-	-	mode	moonphase	P
12	-	time	reset	timerunning	F
13	-	date	start	setting	P
14	-	sea	adjust	tide	P

<https://pairwise.teremokgames.com/4s8/>

[1st Pairwise pass rate]

→ $12/14 \times 100 = 85.71\%$

| Team 8 2nd Pairwise test

mode	input	data	result
mode select	adjust	time	T
tide	start	sea	T
timerunning	reset4	time	T
timer	start	date	T
moonphase	reset1	date	T
alarm	mode	date	T
mode select	mode	sea	T
alarm	start	time	T
tide	mode	time	T
set time	adjust	date	T
stopwatch	adjust	sea	T
set time	start	time	T
stopwatch	start	time	T
mode select	start	date	T
tide	reset4	date	T
tide	reset1	sea	T
timerunning	reset1	date	T
stopwatch	reset1	date	T
timerunning	adjust	sea	T
timer	reset1	time	T
stopwatch	mode	date	T
timerunning	mode	date	T
moonphase	reset4	time	T
alarm	reset1	date	T
timer	mode	time	T
set time	mode	date	T
alarm	reset4	date	T
timer	reset4	time	T
timer	adjust	time	T
alarm	adjust	date	T
timerunning	start	date	T
moonphase	mode	time	T
stopwatch	reset4	sea	T

```
mode: tide, moonphase, timerunning, timer, set time, stopwatch, alarm, mode select
input: start, adjust, mode, reset1, reset4
data: time, date, sea

IF [mode] = "tide" THEN [input] IN {"start", "reset4", "reset1", "mode"};
IF [mode] = "moonphase" THEN [input] IN {"reset1", "reset4", "mode"};
IF [mode] = "timerunning" THEN [input] IN {"start", "adjust", "mode", "reset1", "reset4"};
IF [mode] = "timer" THEN [input] IN {"start", "adjust", "mode", "reset1", "reset4"};
IF [mode] = "set time" THEN [input] IN {"adjust", "start", "mode"};
IF [mode] = "stopwatch" THEN [input] IN {"adjust", "start", "reset1", "reset4", "mode"};
IF [mode] = "alarm" THEN [input] IN {"adjust", "start", "reset1", "reset4", "mode"};
IF [mode] = "mode select" THEN [input] IN {"adjust", "mode", "start"};
IF [mode] = "tide" THEN [data] IN {"sea", "date", "time"};
IF [mode] IN {"moonphase", "timer", "alarm", "set time"} THEN [data] IN {"date", "time"};
```

지난번에 한 pairwise의 경우 부족하다고 판단하여 PICT사용.

Common 환경에서의 성능에는 문제가 없음.

[2nd Pairwise pass rate]

→ 33/33 x 100 = 100%



Brute Force Test -Team 7



| Team 7 Brute Force Test

Test Case & Result – (1)

num	Test case	Summary	Result
1	sma7-1 : R.2.1 시간을 00:00:00으로 세팅하고 타이머 시작	시간을 00:00:00으로 세팅하고 타이머 시작	F
2	sma7-2 : R.3.2 00:00:00인 상태에서 stopwatch 리셋 후 시작	00:00:00인 상태에서 stopwatch 리셋 후 시작	F
3	sma7-3 : R.2.1 Timer의 올바른 시간표시	Timer가 흐를 때, 올바른 시간 단위로 흐르는가를 확인한다. (ex. 1분일때 59초로 제대로 넘어가는가)	T
4	sma7-4 : R.3.0 Stopwatch의 올바른 시간표시	Stopwatch가 흐를 때, 올바른 시간 단위로 흐르는가를 확인한다. (ex. 분은 60분까지, 초는 60초까지/ 59분일 때 1시간으로 제대로 넘어가는가)	F
5	sma7-5 : R.2.1 Timer 실행중 시간 변경	Timer 실행 중 시간 변경이 가능하지 않은지 확인한다.	T
6	sma7-6 : R.3.0 Stopwatch 실행중 시간 변경	Stopwatch 실행 중 시간 변경이 가능하지 않은지 확인한다.	T
7	sma7-7 : R.10.0 중복된 Alarm	Alarm을 2개 이상 중복되게 설정한 경우 알람이 제대로 울리는지 확인한다.	F
8	sma7-8 : R.10.0 Time 모드일 때 알람	Time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
9	sma7-9 : R.10.0 Timer 모드일 때 알람	Timer 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
10	sma7-10 : R.10.0 Stopwatch 모드일 때 알람	Stopwatch 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
11	sma7-11 : R.10.0 Global time 모드일 때 알람	Global time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
12	sma7-12 : R.10.0 Time 모드일 때 타이머	Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
13	sma7-13 : R.10.0 Timer 모드일 때 타이머	Timer 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
14	sma7-14 : R.10.0 Stopwatch 모드일 때 타이머	Stopwatch 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
15	sma7-15 : R.10.0 Global Time모드일 때 타이머	Global Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
16	sma7-81 : R.4.0 Alarm 비활성화 시 알람이 울리지 않는지 확인	Alarm 비활성화 시 알람이 울리지 않는지 확인한다.	T
17	sma7-82 : R.6.0 Global Time이 제대로 설정되는지 확인	Global Time이 설정하고 싶은 시간으로 제대로 설정되는지 확인	F
18	sma7-83 : R.8.0 4가지 모드 설정 가능한지 확인	시간, 타이머, 스탑워치, 알람, 여행지 시간, 난수 생성 중 4가지를 선택하여 모드 변경으로 선택이 가능한지 확인한다.	T

| Team 7 Brute Force Test

Test Case & Result – (2)

Num`	Test case	Summary	Result
19	sma7-84 : R.5.2 난수 숫자 범위 설정이 가능한지 확인	난수를 생성할 숫자 범위 설정이 가능한지 확인한다.	T
20	sma7-85 : R.5.1 난수 생성이 가능한지 확인	User가 난수를 요청하면 R.5.2에서 설정한 숫자 범위 내의 수를 복원 추 출하여 보여주는지 확인	T
21	sma7-86 : R.1.0 요일이 제대로 변경되는지 확인	연, 월, 일 을 조정할 때 날짜에 맞게 요일이 제대로 변하는지 확인	T
22	sma7-87 : R.2.0 타이머의 최대 시간	타이머의 최대 시간이 일정한 지 확인	F
23	sma7-88 : R.1.0 시간 설정	내가 설정하고 싶은 시간대로 시간을 설정할 수 있는가를 확인	F

Team 7 Brute Force Test

Failed Case -(1)

1	sma7-1 : R.2.1 시간을 00:00:00으로 세팅하고 타이머 시작	시간을 00:00:00으로 세팅하고 타이머 시작	F
2	sma7-2 : R.3.2 00:00:00인 상태에서 stopwatch 리셋 후 시작	00:00:00인 상태에서 stopwatch 리셋 후 시작	F

➤ 시계가 고장남

4	sma7-4 : R.3.0 Stopwatch의 올바른 시간표시	Stopwatch가 흐를 때, 올바른 시간 단위로 흐르는가를 확인한다. (ex. 분은 60분까지, 초는 60초까지/ 59분일 때 1시간으로 제대로 넘어가는가)	F
---	------------------------------------	---	---

➤ 59분에서 1시간으로 넘어가지 않는다.

7	sma7-7 : R.10.0 중복된 Alarm	Alarm을 2개 이상 중복되게 설정한 경우 알람이 제대로 울리는지 확인한다.	F
---	---------------------------	---	---

➤ Alarm을 2개 이상 중복되게 설정하면 알람이 울리지 않는다.

8	sma7-8 : R.10.0 Time 모드일 때 알람	Time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
9	sma7-9 : R.10.0 Timer 모드일 때 알람	Timer 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
10	sma7-10 : R.10.0 Stopwatch 모드일 때 알람	Stopwatch 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F
11	sma7-11 : R.10.0 Global time 모드일 때 알람	Global time 모드 화면일 때, 알람이 제때 울리는지 확인한다.	F

➤ 알람이 제시 시간에 울리지 않고 6초 정도 뒤에 울린다.

| Team 7 Brute Force Test

Failed Case – (2)

12	sma7-12 : R.10.0 Time 모드일 때 타이머	Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
13	sma7-13 : R.10.0 Timer 모드일 때 타이머	Timer 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
14	sma7-14 : R.10.0 Stopwatch 모드일 때 타이머	Stopwatch 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F
15	sma7-15 : R.10.0 Global Time모드일 때 타이머	Global Time 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	F

➤ 타이머가 끝나도 알람이 울리지 않는다.

17	sma7-82 : R.6.0 Global Time이 제대로 설정되는지 확인	Global Time이 설정하고 싶은 시간으로 제대로 설정되는지 확인	F
----	---	--	---

➤ 시간이 증가할 때 일정한 간격으로 증가하지 않아 원하는 시간으로 설정하기 불가능 하다.

23	sma7-88 : R.1.0 시간 설정	내가 설정하고 싶은 시간대로 시간을 설정할 수 있는가를 확인	F
----	-----------------------	-----------------------------------	---

➤ 시간을 설정하고 저장이 되지 않는다.

22	sma7-87 : R.2.0 타이머의 최대 시간	타이머의 최대 시간이 일정한 지 확인	F
----	----------------------------	----------------------	---

➤ 요구사항에는 1시간 59분이라고 되어있지만 , 2시간까지도 가능하다.



Brute Force Test
-Team 8



| Team 8 Brute Force Test

Test Case & Result – (1)

num	Test case	Summary	Result
1	sma8-1 : R.0.0 시간측정	시간이 정상적으로 1초마다 업데이트 되는지 확인한다.	T
2	sma8-2 : R.0.1 윤년	날짜 설정 시 윤년을 고려하는지 확인한다.	T
3	sma8-3 : R.2.2 같은 시각의 알람	같은 시각의 알람이 2개 이상 설정된 경우에도 울린다.	T
4	sma8-4 : R.2.4 TimeKeeping 모드일 때 알람	TimeKeeping 모드 화면일 때, 알람이 제때 울리는지 확인한다.	T
5	sma8-5 : R.2.4 Timer 모드일 때 알람	sma8-5 : R.2.4 Timer 모드일 때 알람	T
6	sma8-6 : R.2.4 Stopwatch 모드일 때 알람	Stopwatch 모드 화면일 때, 알람이 제때 울리는지 확인한다.	T
7	sma8-7 : R.2.4 Tide 모드일 때 알람	Tide 모드 화면일 때, 알람이 제때 울리는지 확인한다.	T
8	sma8-8 : R.2.4 Moonphase 모드일 때 알람	Moonphase 모드 화면일 때, 알람이 제때 울리는지 확인한다.	T
9	sma8-9 : R.2.4 SelectMode 모드일 때 알람	SelectMode 모드 화면일 때, 알람이 제때 울리는지 확인한다.	T
10	sma8-10 : R.1.5 TimeKeeping 모드일 때 타이머	TimeKeeping 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
11	sma8-11 : R.1.5 Alarm 모드일 때 타이머	Alarm 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
12	sma8-12 : R.1.5 Stopwatch 모드일 때 타이머	Stopwatch 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
13	sma8-13 : R.1.5 Tide 모드일 때 타이머	Tide 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
14	sma8-14 : R.1.5 Moonphase 모드일 때 타이머	Moonphase 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
15	sma8-15 : R.1.5 SelectMode 모드일 때 타이머	SelectMode 모드 화면일 때, 타이머가 제때 울리는지 확인한다.	T
16	sma8-16 : R.1.0 Timer의 올바른 시간 표시	타이머가 진행될 때, < 시 분 초 > 가 시간 개념에 맞게 정확하게 표시되어 진행되는지 확인한다.	T
17	sma8-17 : R.3.0 Stopwatch의 올바른 시간표시	스톱워치가 진행될 때, < 시 분 초 0.1초 > 가 시간 개념에 맞게 정확하게 표시되어 진행되는지 확인한다.	T
18	sma8-18 : R.6.0 TimeKeeping 모드 화면일 때, modeSelect 진입	TimeKeeping 모드 화면일 때, modeSelect로 제대로 진입하는지 확인	T

| Team 8 Brute Force Test

Test Case & Result – (2)

Num	Test case	Summary	Result
19	sma8-19 : R.6.0 Timer 모드 화면일 때, modeSelect 진입	Timer 모드 화면일 때, modeSelect로 제대로 진입하는지 확인	T
20	sma8-20 : R.6.0 Alarm 모드 화면일 때, modeSelect 진입	Alarm 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	F
21	sma8-21 : R.6.0 Stopwatch 모드 화면일 때, modeSelect 진입	Stopwatch 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	T
22	sma8-22 : R.6.0 Tide 모드 화면일 때, modeSelect 진입	Tide 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	T
23	sma8-23 : R.6.0 Moonphase 모드 화면일 때, modeSelect 진입	Moonphase 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	T

| Team 8 Brute Force Test

Failed Case

20	sma8-20 : R.6.0 Alarm 모드 화면일 때, modeSelect 진입	Alarm 모드 화면일 때, modeSelect로 제대로 진입하는지 확인.	F
----	---	---	---

- Alarm 모드 화면에서 modeSelect 모드로 진입할 수 없다.

3

Static Analysis

- Team 7 Static Analysis
- Team 8 Static Analysis

Static Analysis setting

SonarQube - Overview

조건

경계값 준수 여부는 프로젝트 단위로만 측정하며 하위 프로젝트, 디렉토리, 파일들은 무시합니다. [더보기](#)

Metric	Over Leak Period	Operator	Warning	Error		
Bugs	<input type="checkbox"/>	다음보다 큼	15	30	Update	삭제
코드 악취	<input type="checkbox"/>	다음보다 큼	1000	1500	Update	삭제
Duplicated lines (%)	<input type="checkbox"/>	다음보다 큼	3	5	Update	삭제
보안성 등급	Never	은(는) 다음보다 좋지 않음	B X	C X	Update	삭제

- SonarQube로 static analysis 수행 전 Quality gate 설정.
- Coverage의 경우 SonarQube에서 체크 되지 않는 문제가 발생하여 기준을 따로 설정하지 않음.

Static Analysis setting

SonarQube - rules

The screenshot shows the SonarQube rules configuration interface. On the left, the 'Language' dropdown is set to 'Java' (640 rules). In the center, the '타입coding_rules.facets.languagescoding' section has 'Bug' (565) and 'Vulnerability' (75) selected. On the right, a detailed view of the 'Tag' selection process is shown, with a list of tags and their counts:

Tag	Count
correctness	323
performance	82
multi-threading	78
cwe	75
cert	43
pitfall	10
error-handling	4
java8	4
serialization	4
unpredictable	4

➤ Java Language 선택 후 bug, vulnerability 항목 선택 후 몇 개의 tag 제외하고 모든 rule을 추가함.

- Spring → spring framework/boot project가 아니므로 제외
- Crypto → 별도의 암호화를 거치지 않았으므로 제외
- Android → android 개발이 아니므로 제외
- Owasp → 웹 어플리케이션이 아니므로 제외
- Misra-c 계열, based-on-misra → C언어 개발 표준이므로 제외



640개의 rule 추가.

Static Analysis setting

SonarQube - rules

The screenshot shows the SonarQube rules configuration interface. It is divided into three main sections:

- Language Selection:** A dropdown menu titled "Language" with a checked checkbox. The "Java" option is selected and highlighted with a red box, showing a count of 525.
- Category Selection:** A list of categories under the heading "타입 coding_rules.facets.languagescoding". The "Code Smell" category is selected and highlighted with a red box, showing a count of 525. Other categories include "Bug" (102) and "Vulnerability" (6).
- Tag Selection:** A detailed view of the "Code Smell" category, titled "Tag" with a checked checkbox. It lists various tags and their counts, with the entire list highlighted by a red box. The tags and counts are:

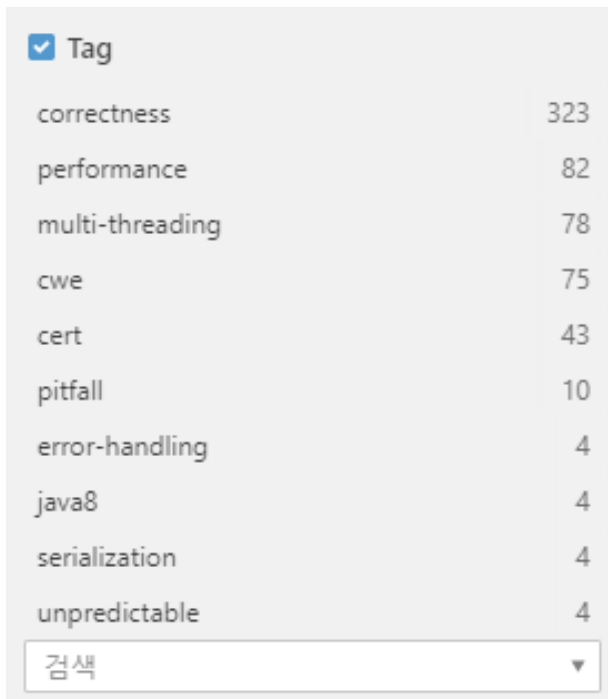
Tag	Count
style	167
bad-practice	116
cert	51
convention	50
pitfall	42
error-handling	39
clumsy	38
performance	32
suspicious	28
brain-overload	26
confusing	26
multithreading	7
unused-code	4

Red arrows indicate the flow of selection: from the "Java" language to the "Code Smell" category, and then to the detailed "Tag" list.

- Java Language 선택 후 code smell 항목 선택 후 몇 개의 tag 제외하고 모든 rule을 추가함.
 - 525개의 rule set 추가.

Static Analysis setting

SonarQube - rules



A screenshot of the SonarQube interface showing a list of built-in rule tags. The 'Tag' checkbox is checked. The list includes tags like 'correctness' (323), 'performance' (82), 'multi-threading' (78), 'cwe' (75), 'cert' (43), 'pitfall' (10), 'error-handling' (4), 'java8' (4), 'serialization' (4), and 'unpredictable' (4). A search box at the bottom contains the Korean text '검색'.

Tag	Count
correctness	323
performance	82
multi-threading	78
cwe	75
cert	43
pitfall	10
error-handling	4
java8	4
serialization	4
unpredictable	4

➤ CERT

- CERT 표준을 만족했는지 확인하는 rule.
- 대부분의 규칙이 language specific 하지 않음.
- good programming practice를 위해 필요

➤ CWE

- source code 보안 취약점 7가지(common weakness enumeration)

➤ Serialization

- object 직렬화 검사를 위한 rule.

➤ Error-handling

- error 처리(try – catch, exception)를 살펴보기 위한 rule

<https://docs.sonarqube.org/latest/user-guide/built-in-rule-tags/>

<https://rules.sonarsource.com/java>

Static Analysis setting

SonarQube - rules

<input checked="" type="checkbox"/> Tag	
style	167
bad-practice	116
cert	51
convention	50
pitfall	42
error-handling	39
clumsy	38
performance	32
suspicious	28
brain-overload	26
confusing	26
multithreading	7
unused-code	4

<https://docs.sonarqube.org/latest/user-guide/built-in-rule-tags/>

<https://rules.sonarsource.com/java>

- **Bad-practice**
 - 설계된 대로 실행은 되는데 설계 자체가 bad idea인 code.
- **Pitfall**
 - 아직 잘못되진 않았는데 미래에 잘못될 수도 있는 code.
- **Convention**
 - naming, formatting 같은 coding 습관.
- **Confusing**
 - 관리자가 이해하기 어렵게 만드는 code.
- **Unused-code**
 - 작성은 됐으나 실제로 사용되지 않는 code.
- **Suspicious**
 - bug라고 정해진 것은 아니지만 의심스러운 code.
- **Clumsy**
 - 어떤 작업을 수행할 때 불필요한 추가작업을 수행하는 code.

Static Analysis setting

SonarQube - rules

The screenshot shows the SonarQube rules configuration interface. On the left, a table lists rules with columns for '규칙' (Rule), '활성화' (Activated), and '비활성화' (Deactivated). The table shows 22 rules that are not active. A modal dialog is open, showing options to '대규모 변경' (Bulk Change) and '모든 필터 초기화' (Reset All Filters). The dialog has options to '다음에서 활성화...' (Activate from here...), '다음에서 비활성화...' (Deactivate from here...), and '다음에서 비활성화 custom' (Deactivate from here custom). The '다음에서 비활성화 custom' option is highlighted with a red box. Below the table, a red box highlights the text '유효하지 않은 규칙들 22' (Invalid rules 22). Another red box highlights the '기본값' (Default) button in the bottom right corner of the interface.

규칙	활성화	비활성화
Total	1.2k	708
Bugs	565	21
Vulnerabilities	75	126
Code Smells	524	561

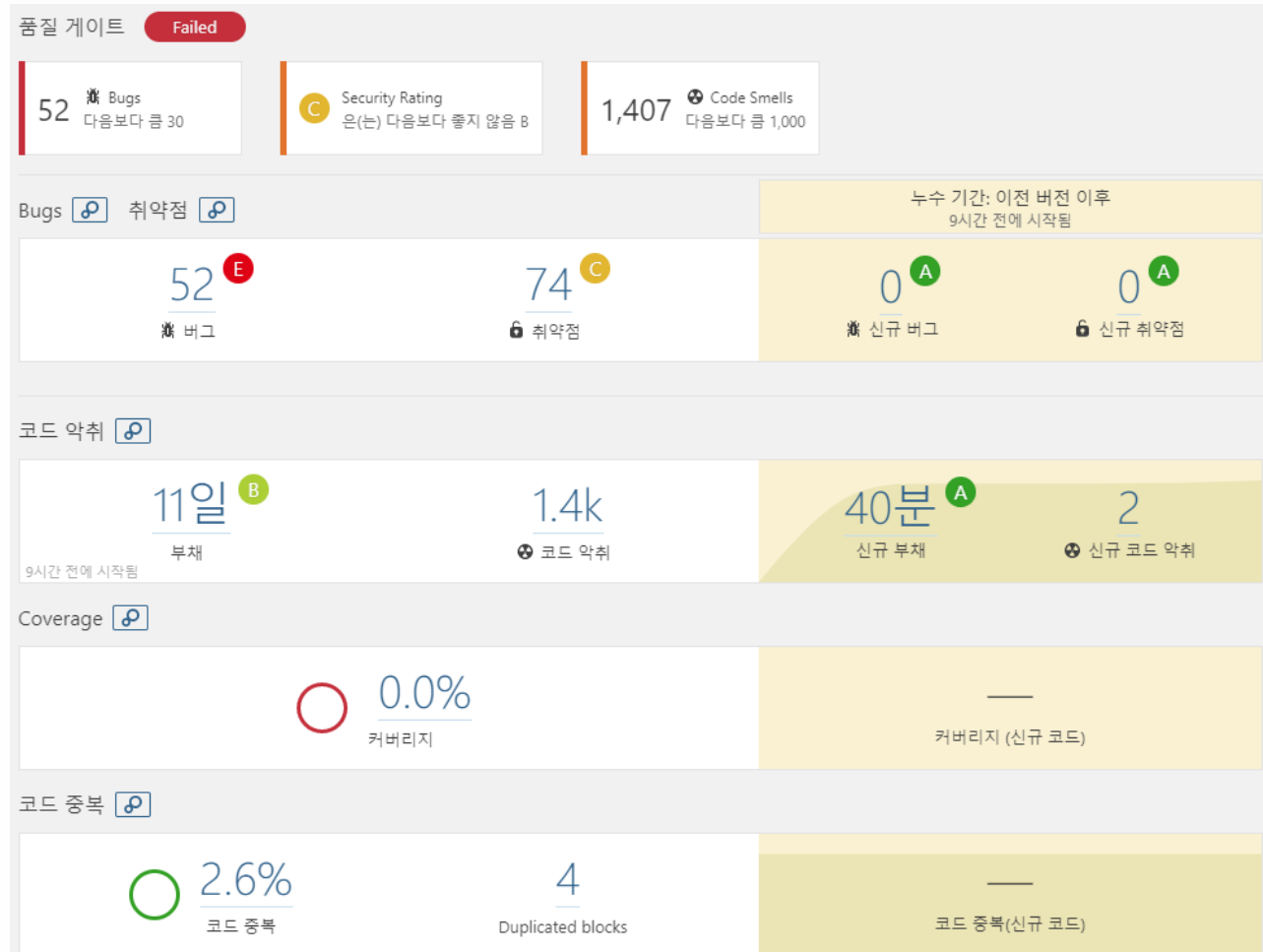
유효하지 않은 규칙들 22

기본값 1142 8초 전 21분 전

- 중복되는 (duplicated) rule 22개가 있어서 비활성화 처리하여 없앴.

Team 7 Static Analysis

SonarQube - Overview



Team 7 Static Analysis

SonarQube - Overview

- Bug

심각도			
Blocker	1	Minor	19
Critical	0	Info	0
Major	32		

- Vulnerability

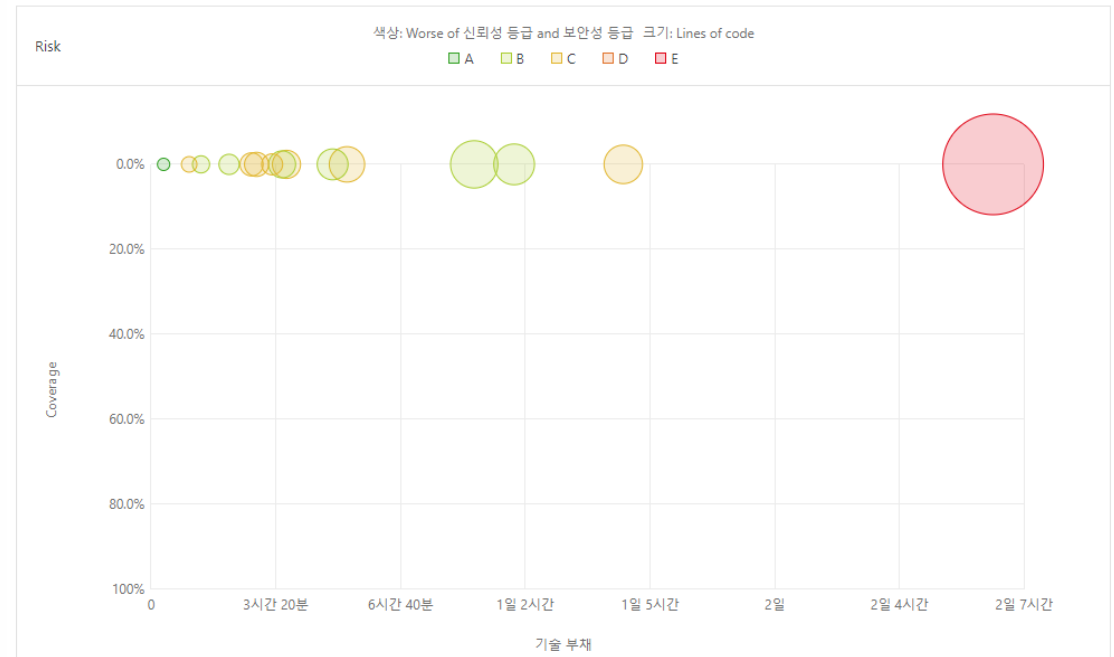
심각도			
Blocker	0	Minor	73
Critical	0	Info	0
Major	1		

- Code smell

심각도			
Blocker	5	Minor	1.2k
Critical	34	Info	16
Major	103		

Type	
Bug	52
Vulnerability	74
Code Smell	1.4k

Project overview



Team 7 Static Analysis

SonarQube - Bug

```
TimerTask task = new TimerTask() {
    @Override
    public void run(){
        if(isAlarmActivated) {
            if (isAlarmKilled) {
                try {
                    Thread.sleep(60000);
                    isAlarmKilled = false;
                } catch (InterruptedException e) {
```

Either re-interrupt this method or rethrow the "InterruptedException". ... 12일 전 ▾ L28 🔗

🚩 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 15min effort 코멘트

🔗 cwe, error-handling, multi-threading ▾

```
                try {
                    Thread.sleep(9);
                } catch (InterruptedException e) {
```

Either re-interrupt this method or rethrow the "InterruptedException". ... 3일 전 ▾ L706 🔗

🚩 Bug ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 15min effort 코멘트

```
                e.getMessage();
```

➤ InterruptedException 문제 발생

- 현재 사용하는 thread에게 확실하게 interrupt를 발생시켜줘야 함

| Team 7 Static Analysis

SonarQube - Bug

```
public void selectNextSelectableMode(int arrayIndex){ //입력값은 modesArrayIndex // 일반 모드에서 다음모드 선택.

    this.modesArrayIndex=arrayIndex;
    System.out.println(" mode index: "+this.modesArrayIndex);
    if(this.modesArrayIndex==5){
        this.modesArrayIndex=0;

    } else {

        if(currentModes[modesArrayIndex+1]){
            modesArrayIndex++;
        } else {
            this.selectNextSelectableMode(modesArrayIndex+1);
        }
    }
}
```

Method SetModes.selectNextSelectableMode(int) employs tail recursion ... 3일 전 L70

Bug Major Open 할당되지 않음 코멘트 performance

- tail recursive의 경우 반복문으로 refactoring 하는 것이 더 나을 수 있음.

| Team 7 Static Analysis

SonarQube - Vulnerability

```
public void generateRandomNumber() {  
    randomNumber = ((int) (Math.random()*this.inputNumber))+1; // 0~59까지 표현 가능  
}
```

The use of java.lang.Math.random() is predictable ... 3일 전 L37

Vulnerability Major Open 할당되지 않음 코멘트 cwe

➤ Predictable random 문제 발생

- SecureRandom을 사용해야 함.

Team 7 Static Analysis

SonarQube - Vulnerability

```
Alarm alarm0 = new Alarm();
```

Explicitly declare the visibility for "alarm0". ... 12일 전 ▼ L13 🔗

🔒 Vulnerability ▼ ⬇️ Minor ▼ ○ Open ▼ 할당되지 않음 ▼ 5min effort 코멘트 No tags ▼

```
Alarm alarm1 = new Alarm();
```

Explicitly declare the visibility for "alarm1". ... 12일 전 ▼ L14 🔗

🔒 Vulnerability ▼ ⬇️ Minor ▼ ○ Open ▼ 할당되지 않음 ▼ 5min effort 코멘트 No tags ▼

```
Alarm alarm2 = new Alarm();
```

Explicitly declare the visibility for "alarm2". ... 12일 전 ▼ L15 🔗

🔒 Vulnerability ▼ ⬇️ Minor ▼ ○ Open ▼ 할당되지 않음 ▼ 5min effort 코멘트 No tags ▼

```
Alarm alarm3 = new Alarm();
```

Explicitly declare the visibility for "alarm3". ... 12일 전 ▼ L16 🔗

🔒 Vulnerability ▼ ⬇️ Minor ▼ ○ Open ▼ 할당되지 않음 ▼ 5min effort 코멘트 No tags ▼

➤ 대부분은 접근 지정자 문제

| Team 7 Static Analysis

SonarQube – Code smell

```
try {
    Thread.sleep(60000);
    isAlarmKilled = false;
} catch (InterruptedException e) {
```

Either remove or fill this block of code. ... 12일 전 ▾ L28 🔗

🔍 Code Smell ▾ 🚨 Major ▾ 🔓 Open ▾ 할당되지 않음 ▾ 5min effort 코멘트 🗑 suspicious ▾

```
    }
} else {
    isAlarmSound(currentTime.get(Calendar.HOUR_OF_DAY), currentTime.get(Calendar.MINUTE));
}
```

- Code block은 채우거나 없앨 것.
 - Bug 항목의 interrupt 문제와 연결.

Team 7 Static Analysis

SonarQube – Code smell

```
public boolean isAlarmActivated(){ // GUI에서 종료할 아이콘 전송.
    if(alarm0.getIsAlarmActivated() || alarm1.getIsAlarmActivated() || alarm2.getIsAlarmActivated() || alarm3.getIsAlarmActivated()){
        return true;
    } else {
        return false;
    }
}

final URL[] icon_image =
{
    getClass().getClassLoader().getResource("icon_1.png"), getClass().getClassLoader().getResource("icon_2.png"),
    getClass().getClassLoader().getResource("icon_3.png"), getClass().getClassLoader().getResource("icon_4.png"),
    getClass().getClassLoader().getResource("icon_5.png"), getClass().getClassLoader().getResource("icon_6.png"),
    getClass().getClassLoader().getResource("icon_7.png"), getClass().getClassLoader().getResource("icon_8.png"),
}
```

Split this 137 characters long line (which is greater than 120 authorized). ... 12일 전 L214

Code Smell Major Open 할당되지 않음 1min effort 코멘트 convention

Split this 129 characters long line (which is greater than 120 authorized). ... 3일 전 L23

Code Smell Major Open 할당되지 않음 1min effort 코멘트 convention

Split this 129 characters long line (which is greater than 120 authorized). ... 3일 전 L24

Code Smell Major Open 할당되지 않음 1min effort 코멘트 convention

Split this 129 characters long line (which is greater than 120 authorized). ... 3일 전 L25

Code Smell Major Open 할당되지 않음 1min effort 코멘트 convention

Split this 130 characters long line (which is greater than 120 authorized). ... 3일 전 L26

Code Smell Major Open 할당되지 않음 1min effort 코멘트 convention

➤ 한 line이 너무 길어짐.

| Team 7 Static Analysis

SonarQube – Code smell

```
public static int getDayOFWeekDigit(int number, int base){  
    switch (base){ // 세그먼트 위치  
        case 0:  
            switch (number){ // 요일
```

Refactor the code to eliminate this nested "switch". ... 3일 전 ▾ L72 🔗

🔍 Code Smell ▾ 🚨 Critical ▾ 🔓 Open ▾ 할당되지 않음 ▾ 10min effort 코멘트 🗑️ pitfall ▾

➤ 중첩 switch 문은 나눌 것.

| Team 7 Static Analysis

SonarQube – Code smell

```
...
for (int i = 0; i < 15; i++) {
    segment[i].setName("Segment #" + i);
    segment[i].setLayout(null);
    for (int j = 0; j < 7; j++) segment[i].add(segment[i].getPanelSeg(j));
}
}
```

Missing curly brace. ... 3일 전 ▾ L94 🔗

⊕ Code Smell ▾ ⬆️ Critical ▾ ○ Open ▾ 할당되지 않음 ▾ 2min effort 코멘트 based-on-misra, cert, pitfall ▾

At most one statement is allowed per line, but 2 statements were found on this line. ... 3일 전 ▾ L94 🔗

⊕ Code Smell ▾ ⬆️ Major ▾ ○ Open ▾ 할당되지 않음 ▾ 1min effort 코멘트 style ▾

- 가급적이면 { }를 사용하고, 하나의 line에 하나의 statement를 기입할 것.

| Team 7 Static Analysis

Code Coverage(IntelliJ-Junit)

Element	Class, %	Method, %	Line, %
Alarm	100% (2/2)	40% (6/15)	40% (23/57)
AlarmManager	100% (1/1)	35% (5/14)	16% (16/98)
GlobalTime	100% (2/2)	33% (4/12)	43% (14/32)
NumberGenerator	100% (1/1)	100% (6/6)	92% (13/14)
SetModes	100% (1/1)	58% (7/12)	62% (28/45)
SetTime	100% (2/2)	39% (9/23)	46% (47/101)
Stopwatch	0% (0/2)	0% (0/8)	0% (0/21)
TimeKeeping	100% (2/2)	50% (10/20)	68% (42/61)
TimerT	0% (0/2)	0% (0/12)	0% (0/42)
WatchSystem	0% (0/1)	0% (0/71)	0% (0/126)

- 7조의 Unit Test code의 code coverage
- 0%인 Stopwatch, TimerT, WatchSystem은 아예 Unit Test code의 메소드가 작성이 되어있지 않음을 알 수 있다.

| Team 7 Static Analysis

Code Coverage(IntelliJ-Junit)

▼ ❌ <default package>	268 ms
▶️ ✓ SetModesTest	27 ms
▶️ ✓ StopwatchTest	1 ms
▶️ ❌ GlobalTimeTest	37 ms
▶️ ❌ NumberGeneratorTest	12 ms
▶️ ✓ TimeKeepingTest	19 ms
▶️ ✓ TimerTTest	5 ms
▶️ ❌ SetTimeTest	69 ms
▶️ ✓ WatchSystemTest	15 ms
▶️ ✓ AlarmTest	4 ms
▼ ❌ AlarmManagerTest	79 ms
✓ deactivateAlarm	
✓ isBuzzerOn	
✓ isAlarmActivated	
✓ killAlarm	1 ms
✓ getAlarmIndex	41 ms
✓ addAlarmIndex	11 ms
✓ getAlarmTimeMinute	
❌ setAlarmHour	12 ms
✓ getAlarmTimeHour	
✓ activateAlarm	
❌ setAlarmMinute	14 ms

❌ Tests failed: 11, passed: 138 of 149 tests – 268 ms

```
org.junit.ComparisonFailure: expected:<[]19> but was:<[20]19>  
Expected :19  
Actual   :2019
```

```
java.lang.AssertionError: expected:<1> but was:<0>  
Expected :1  
Actual   :0
```

```
java.lang.AssertionError: expected:<12> but was:<10>  
Expected :12  
Actual   :10
```


















```
org.junit.ComparisonFailure: expected:<[]19> but was:<[20]19>  
Expected :19  
Actual   :2019
```

```
java.lang.AssertionError: expected:<1> but was:<0>  
Expected :1  
Actual   :0
```

- GlobalTimeTest, NumberGeneratorTest, SetTimeTest, AlarmManagerTest에서 검출한 fail된 테스트케이스

Team 7 Static Analysis

Code Coverage(Eclipse-EclEmma)

Element	Coverage ^	Covered Instructions	Missed Instructions	Total Instructions
▼ SM07/src/main/java	 54.6 %	2,227	1,854	4,081
▼ (default package)	 54.6 %	2,227	1,854	4,081
▶ UI.java	 40.6 %	278	407	685
▶ WatchSystem.java	 16.3 %	50	257	307
▶ SetTime.java	 24.0 %	65	206	271
▶ TimerT.java	 15.6 %	35	190	225
▶ AlarmManager.java	 20.1 %	47	187	234
▶ SetModes.java	 31.7 %	52	112	164
▶ TimeKeeping.java	 43.1 %	85	112	197
▶ Alarm.java	 28.2 %	42	107	149
▶ Segment.java	 75.8 %	319	102	421
▶ GlobalTime.java	 48.2 %	66	71	137
▶ Stopwatch.java	 37.7 %	26	43	69
▶ NumberGenerator.java	 20.9 %	9	34	43
▶ GUI.java	 98.1 %	830	16	846
▶ Main.java	 36.4 %	4	7	11
▶ Constants.java	 99.1 %	319	3	322

Team 7 Static Analysis

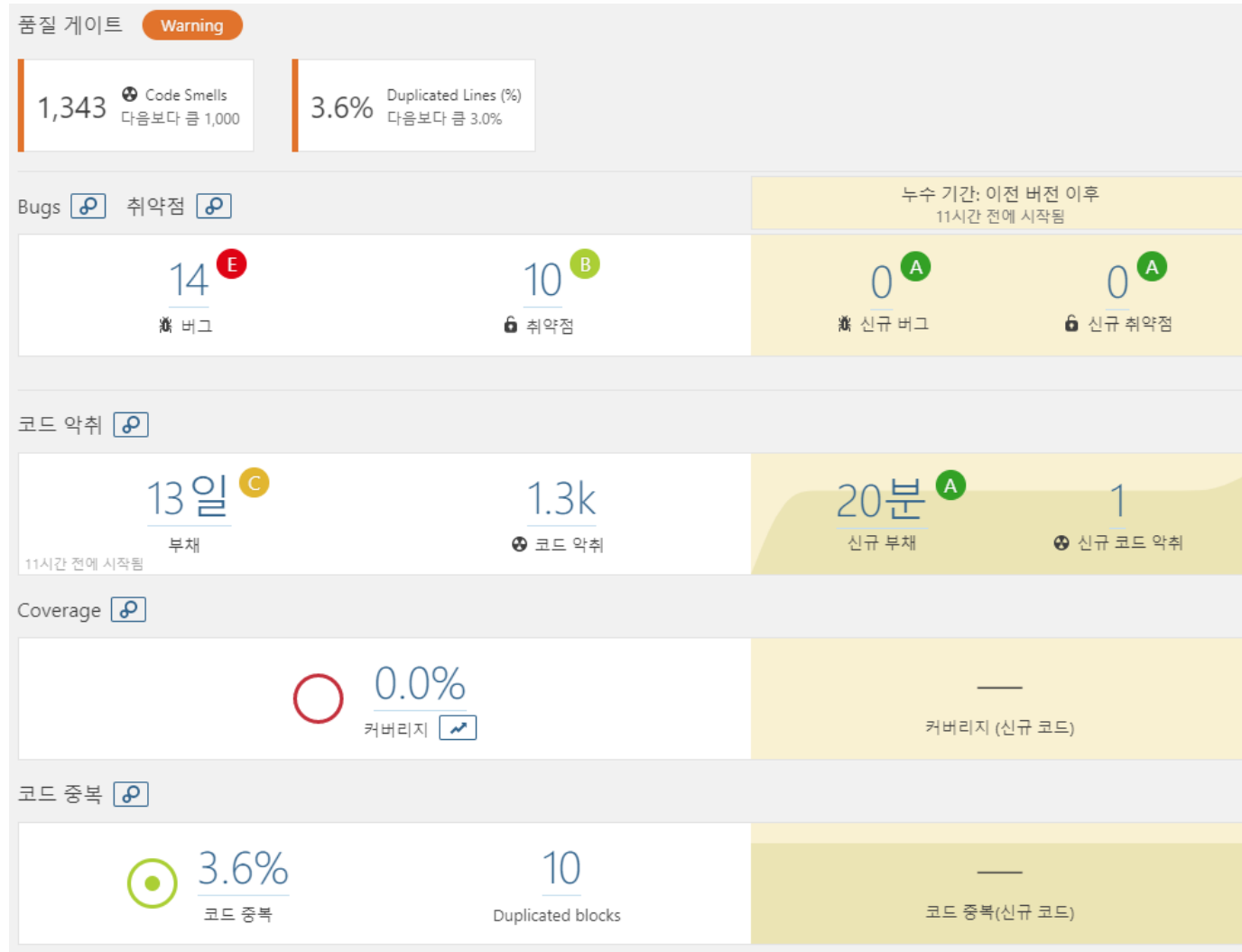
Code Coverage(Eclipse-EclEmma)

Element	Coverage ^
SM07/src/main/java	54.6 %
(default package)	54.6 %
UI.java	40.6 %
WatchSystem.java	16.3 %
SetTime.java	24.0 %
TimerT.java	15.6 %
AlarmManager.java	20.1 %
SetModes.java	31.7 %
TimeKeeping.java	43.1 %
Alarm.java	28.2 %
Segment.java	75.8 %
GlobalTime.java	48.2 %
Stopwatch.java	37.7 %
NumberGenerator.java	20.9 %
GUI.java	98.1 %
Main.java	36.4 %
Constants.java	99.1 %

```
Alarm.java
1 import java.util.Timer;
2
3
4
5
6
7 public class Alarm extends TimeKeeping{
8     private boolean isAlarmActivated;
9     private int alarmTimeHour;
10    private int alarmTimeMinute;
11    private boolean isAlarmKilled;
12    //private int currentHourModifier;
13    // private int currentMinuteModifier;
14    private boolean buzzerOn;
15    //StringTokenizer st;
16    Calendar currentTime;
17
18    TimerTask task = new TimerTask() {
19        @Override
20        public void run(){
21            if(isAlarmActivated) {
22                if (isAlarmKilled) {
23                    try {
24                        Thread.sleep(60000);
25                        isAlarmKilled = false;
26                    } catch (InterruptedException e) {
27
28                    }
29                } else {
30                    isAlarmSound(currentTime.get(Calendar.
31
32                }
33            }
34        };
35
36    Timer timer = new Timer();
37
```

Team 8 Static Analysis

SonarQube - Overview



Team 8 Static Analysis

SonarQube - Overview

- Bug

▼ 심각도				
🚫 Blocker	1	🟢 Minor	2	
🔴 Critical	0	ℹ Info	0	
🔴 Major	11			

- Vulnerability

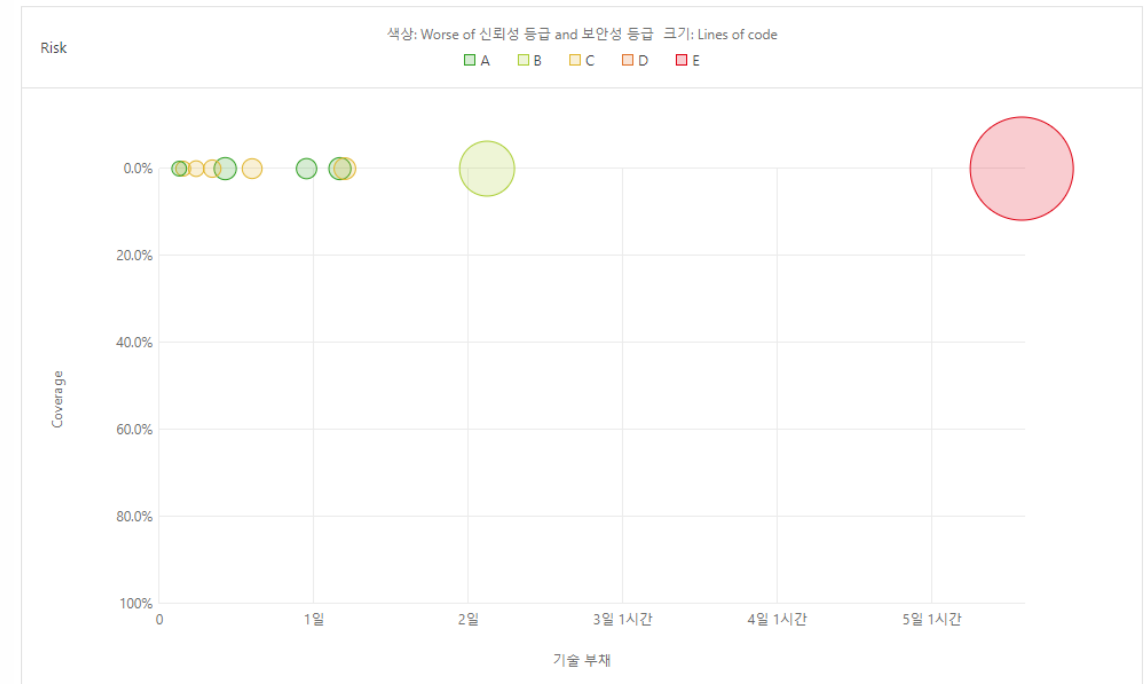
▼ 심각도				
🚫 Blocker	0	🟢 Minor	10	
🔴 Critical	0	ℹ Info	0	
🔴 Major	0			

- Code smell

▼ 심각도				
🚫 Blocker	1	🟢 Minor	1.1k	
🔴 Critical	52	ℹ Info	0	
🔴 Major	174			

▼ Type		
🚫 Bug	14	
🔒 Vulnerability	10	
⊕ Code Smell	1.3k	

Project overview



Team 8 Static Analysis

SonarQube - Bug

```
    }
    Thread.sleep(500);
} catch (InterruptedException e){break;}
```

Either re-interrupt this method or rethrow the "InterruptedException". ... 11일 전 ▾ L127 🔗
Bug ▾ Major ▾ Open ▾ 할당되지 않음 ▾ 15min effort 코멘트 cwe, error-handling, multi-threading ▾

```
        Thread.sleep(1000);
    } catch (InterruptedException e) {}
```

Either re-interrupt this method or rethrow the "InterruptedException". ... 12일 전 ▾ L27 🔗
Bug ▾ Major ▾ Open ▾ 할당되지 않음 ▾ 15min effort 코멘트 cwe, error-handling, multi-threading ▾

```
    }
    try {
        t.sleep(10);
    } catch (InterruptedException e) {
```

Either re-interrupt this method or rethrow the "InterruptedException". ... 14일 전 ▾ L1440 🔗
Bug ▾ Major ▾ Open ▾ 할당되지 않음 ▾ 15min effort 코멘트 cwe, error-handling, multi-threading ▾

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

➤ 동일하게 InterruptedException 문제 발생

- 현재 사용하는 thread에게 확실하게 interrupt를 발생시켜줘야 함

| Team 8 Static Analysis

SonarQube - Vulnerability

```
public void setSettingModelList(ArrayList<String> settingModelList){
    this.settingModelList=settingModelList;
}

//      createMode.setCreateList(createList);
//      deleteMode.setDeleteList(deleteList);
}
```

Store a copy of "settingModelList". ... 19일 전 ▾ L45 🔗

🔒 Vulnerability ▾ 🟢 Minor ▾ 🔵 Open ▾ 할당되지 않음 ▾ 5min effort 코멘트 cert, cwe, unpredictable ▾

➤ Copy store 문제 발생

- *.clone() method를 사용해야 함.

Team 8 Static Analysis

SonarQube – Code smell

```
public class DeleteMode {  
    Add a constructor to the class, or provide default values. ... 18일 전 L6  
    Code Smell Major Open 할당되지 않음 5min effort 코멘트 pitfall  
    // private String[] deleteList;  
    1 private ArrayList<String> deleteList;  
  
public class CreateMode {  
    Add a constructor to the class, or provide default values. ... 18일 전 L4  
    Code Smell Major Open 할당되지 않음 5min effort 코멘트 pitfall  
    // private String[] createList;  
    1 private ArrayList<String> createList;  
  
    // 현재 생성해야 하는 모드들이 createList에 저장  
    public void setCreateList(ArrayList<String> createList) {  
        this.createList = createList;  
        createModeByList();  
    }  
}
```

➤ 생성자를 사용하거나, default value를 할당해주어야 함.

Team 8 Static Analysis

SonarQube – Code smell

```
15 private static int times = 0;
```

```
public void setStopwatch(String time) {  
    String[] times = time.split("\\s");  
}
```

Rename "times" which hides the field declared at line 15. ... 17일 전 L74

Code Smell Major Open 할당되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

```
8 private int i,j,k;
```

```
for (int i = 0; i < matchTable.length; i++) {  
    dt[i] = 0;  
    for (int j = 0; j < 12; j++) {
```

Rename "i" which hides the field declared at line 8. ... 13일 전 L343

Code Smell Major Open 할당되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

Rename "j" which hides the field declared at line 8. ... 13일 전 L345

Code Smell Major Open 할당되지 않음 5min effort 코멘트 based-on-misra, cert, pitfall, suspicious

➤ 같은 이름을 가진 다른 변수가 이미 존재함.

| Team 8 Static Analysis

SonarQube – Code smell

```
lblFirst.setText("Timekeeping");  
t.start();
```

Move this "start" call to another method. ... 14일 전 ▾ L135 🔗

🔍 Code Smell ▾ 🚫 Blocker ▾ 🔵 Open ▾ 할당되지 않음 ▾ 30min effort 코멘트 🏷️ cert, multi-threading, pitfall ▾

- 상속받은 후 자식에서 super()를 호출했을 때 의도치 않게 thread가 실행될 수 있음.

| Team 8 Static Analysis

SonarQube – Code smell

```
else if(str[1].equals("5")) str[1]="MAY";
```

Missing curly brace. ... 13일 전 ▾ L471 🔗

⊕ Code Smell ▾ ⬆ Critical ▾ ○ Open ▾ 할당되지 않음 ▾ 2min effort 코멘트 based-on-misra, cert, pitfall ▾

At most one statement is allowed per line, but 2 statements were found on this line. ... 13일 전 ▾ L471 🔗

⊕ Code Smell ▾ ⬆ Major ▾ ○ Open ▾ 할당되지 않음 ▾ 1min effort 코멘트 style ▾

- 가급적이면 {}를 사용하고, 하나의 line에 하나의 statement만 넣을 것

| Team 8 Static Analysis

SonarQube – Code smell

```
try {  
    Thread.sleep(1000);  
}catch (InterruptedException e) {}
```

Either remove or fill this block of code. ... 12일 전 ▾ L27 🔗

🔍 Code Smell ▾ 🚫 Major ▾ 🔵 Open ▾ 할당되지 않음 ▾ 5min effort 코멘트 🗑 suspicious ▾

```
}
```

- Code block은 채우거나 없앨 것.
 - Bug 항목의 interrupt 문제와 연결.

| Team 8 Static Analysis























Code Coverage(IntelliJ-Junit)

Element	Class, %	Method, %	Line, %
Alarm	100% (1/1)	100% (11/11)	94% (49/52)
CreateMode	100% (1/1)	100% (2/2)	57% (11/19)
DeleteMode	100% (1/1)	100% (2/2)	52% (13/25)
ModeSelector	100% (1/1)	75% (6/8)	85% (35/41)
Moonphase	100% (1/1)	40% (2/5)	25% (14/55)
Stopwatch	100% (1/1)	100% (15/15)	87% (51/58)
SystemUI	100% (6/6)	16% (8/49)	13% (87/667)
Tide	100% (1/1)	50% (3/6)	19% (31/157)
TimeDB	66% (2/3)	92% (13/14)	72% (54/75)
TimeKeeping	100% (1/1)	100% (5/5)	100% (17/17)
Timer	33% (1/3)	77% (14/18)	74% (50/67)

- 8조의 Unit Test code의 code coverage
- %로 보았을 때, 7조보다 비교적 Unit Test code를 잘 작성하였음을 알 수 있다.

Team 8 Static Analysis

Code Coverage(IntelliJ-Junit)

▼  <default package>	4 s 469 ms
▶  AlarmTest	12 ms
▶  ModeSelectorTest	12 ms
▶  DeleteModeTest	6 ms
▶  TimeKeepingTest	21 ms
▶  MoonphaseTest	18 ms
▶  TimerTest	27 ms
▼  SystemUITest	4 s 354 ms
 run	0 ms
 main	4 s 354 ms
▶  TideTest	1 ms
▶  CreateModeTest	0 ms
▶  TimeDBTest	1 ms
▼  StopwatchTest	17 ms
 getInstance	0 ms
 getZeroSate	0 ms
 deleteInstance	1 ms
 getRunState	0 ms
 setStopwatch	1 ms
 pauseStopwatch	9 ms
 updateTime	0 ms
 getTime	0 ms
 recordStopwatch	0 ms
 startStopwatch	3 ms
 resetStopwatch	3 ms

Tests failed: 1, passed: 55 of 56 tests – 4 s 469 ms

➤ 8조의 fail된 테스트케이스는 단 하나로, 다음과 같다.

```
/Library/Java/JavaVirtualMachines/jdk1.8.0_201.jdk/Contents/Home/bin/java ...  
—— IntelliJ IDEA coverage runner ——  
sampling ...  
include patterns:  
exclude patterns:  
java.lang.Exception: Method calculateMoonphase() should be public
```

Team 8 Static Analysis

Code Coverage(Eclipse-EclEmma)

Element	Coverage ^	Covered Instructions	Missed Instructions	Total Instructions
▼ digital-watch		13,880	4,510	18,390
▼ src/main/java		13,180	4,480	17,660
▼ (default package)		13,180	4,480	17,660
▶ Moonphase.java		0	147	147
▶ SystemUI.java		72	3,413	3,485
▶ Alarm.java		107	94	201
▶ TimeDB.java		225	128	353
▶ DeleteMode.java		38	18	56
▶ CreateMode.java		34	12	46
▶ Timer.java		162	56	218
▶ ModeSelector.java		126	18	144
▶ Stopwatch.java		176	21	197
▶ Tide.java		12,150	573	12,723
▶ TimeKeeping.java		90	0	90

➤ Coverage %가 전체적으로 높다는 것을 알 수 있다.

Team 8 Static Analysis

Code Coverage(Eclipse-EclEmma)

Element	Coverage ^
digital-watch	75.5 %
src/main/java	74.6 %
(default package)	74.6 %
Moonphase.java	0.0 %
SystemUI.java	2.1 %
Alarm.java	53.2 %
TimeDB.java	63.7 %
DeleteMode.java	67.9 %
CreateMode.java	73.9 %
Timer.java	74.3 %
ModeSelector.java	87.5 %
Stopwatch.java	89.3 %
Tide.java	95.5 %
TimeKeeping.java	100.0 %

CreateMode.java의 일부

```
20 private void createModeByList(){
21     for(int i=0;i<createList.size();i++){
22         //일단 String값으로 비교를 하도록 설정했는데
23         switch (createList.get(i)){
24             case "TimeKeeping":
25                 TimeKeeping.getInstance();
26                 break;
27             case "Timer":
28                 Timer.getInstance();
29                 break;
30             case "Alarm":
31                 Alarm.getInstance();
32                 break;
33             case "Stopwatch":
34                 Stopwatch.getInstance();
35                 break;
36             case "Tide":
37                 Tide.getInstance();
38                 break;
39             case "Moonphase":
40                 Moonphase.getInstance();
41                 break;
42             default:
43                 break;
44         }
45     }
46 }
47 }
```


4

Overall

- Team 7 Overall
- Team 8 Overall

| Team 7 Overall

1st System Test Result

➤ **Category-partition Test**

$162/209 \times 100 = 77.5\% \text{ PASS}$

➤ **Pairwise Test**

$25/29 \times 100 = 86.2\% \text{ PASS}$

➤ **Brute Force Test**

$9/20 \times 100 = 45\% \text{ PASS}$

2nd System Test Result

➤ **Category-partition Test**

$53/59 \times 100 = 89.8\% \text{ PASS}$

➤ **Pairwise Test**

$24/30 \times 100 = 80\% \text{ PASS}$

➤ **Brute Force Test**

$15/23 \times 100 = 65.21\% \text{ PASS}$

| Team 8 Overall

1st System Test Result

➤ **Category-partition Test**

$81/95 \times 100 = 85.26\% \text{ PASS}$

➤ **Pairwise Test**

$12/14 \times 100 = 85.71\% \text{ PASS}$

➤ **Brute Force Test**

$22/23 \times 100 = 96\% \text{ PASS}$

2nd System Test Result

➤ **Category-partition Test**

$86/97 \times 100 = 88.65\% \text{ PASS}$

➤ **Pairwise Test**

$33/33 \times 100 = 100.0\% \text{ PASS}$

➤ **Brute Force Test**

$22/23 \times 100 = 96\% \text{ PASS}$



#5 2nd Testing - System Test/Static Analysis

THANK YOU