

OOPT STAGE 2050/2060

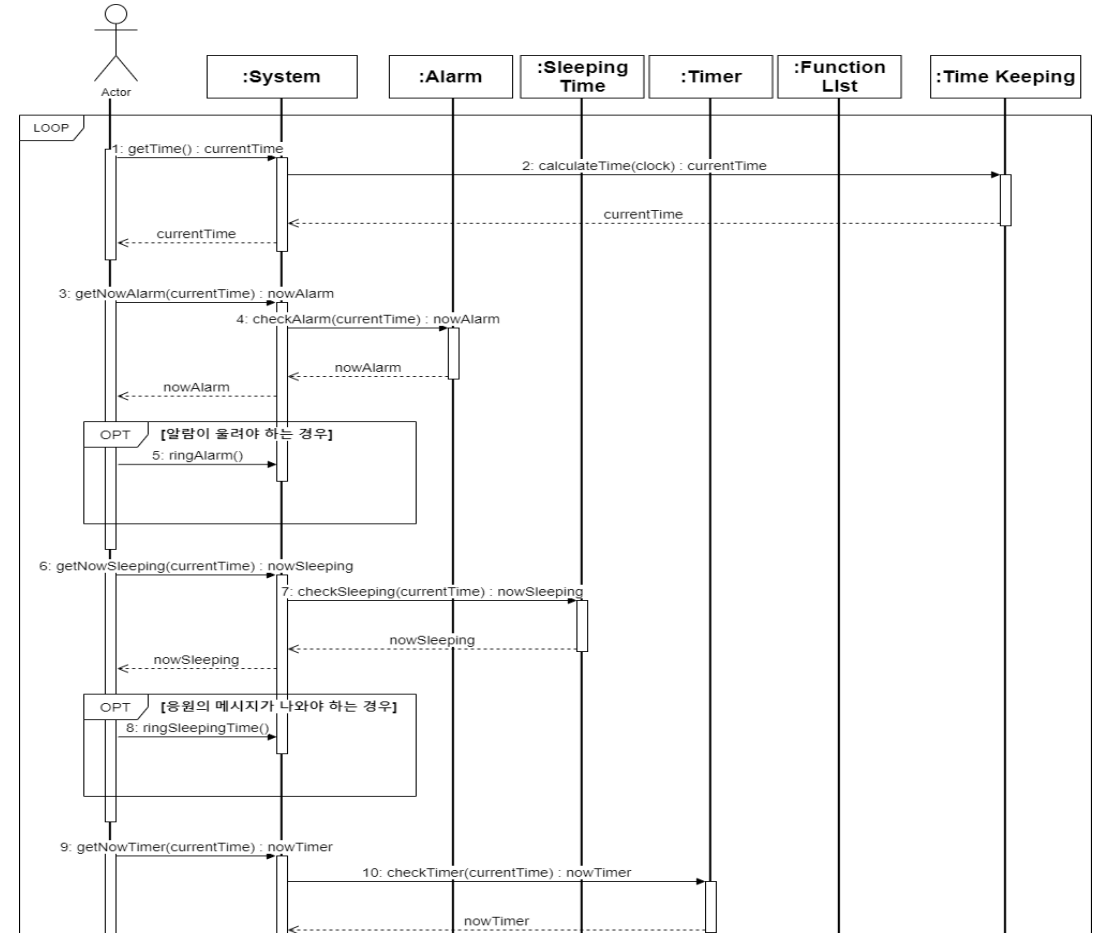
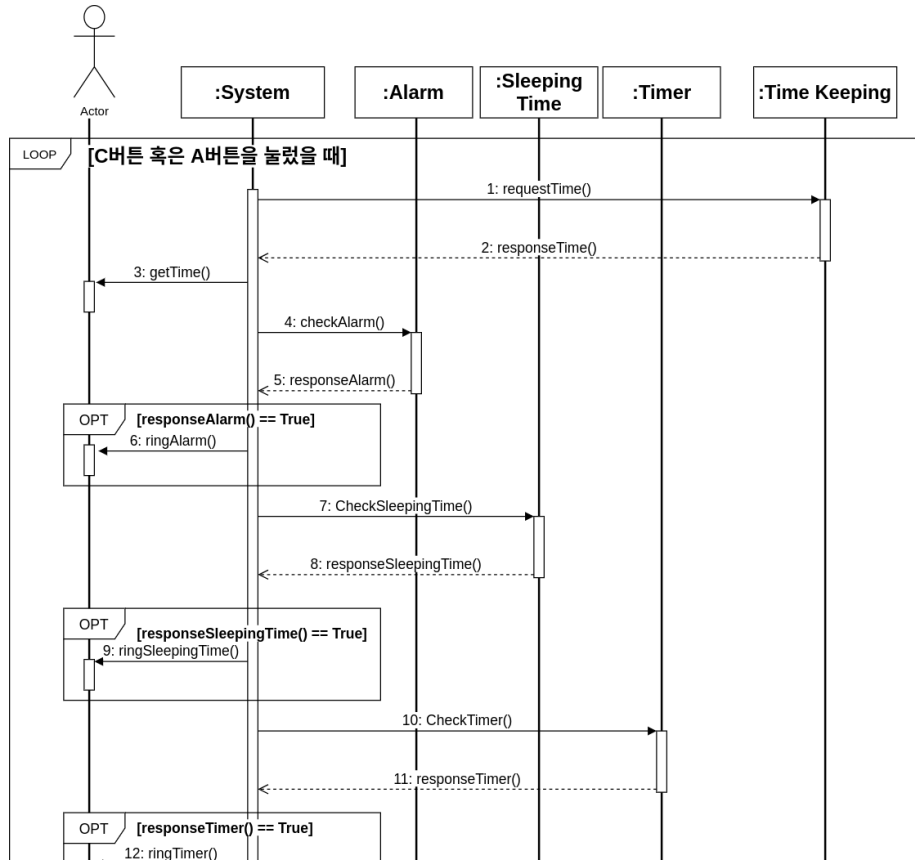
Team #4 김경진, 박용준, 이관석, 이유진

CONTENTS

2040	Refinements in previous stages
2055	Write Unit Test Code
2061	Unit Testing
2063	System Testing
2066	Testing Traceability Analysis

Activity 2044

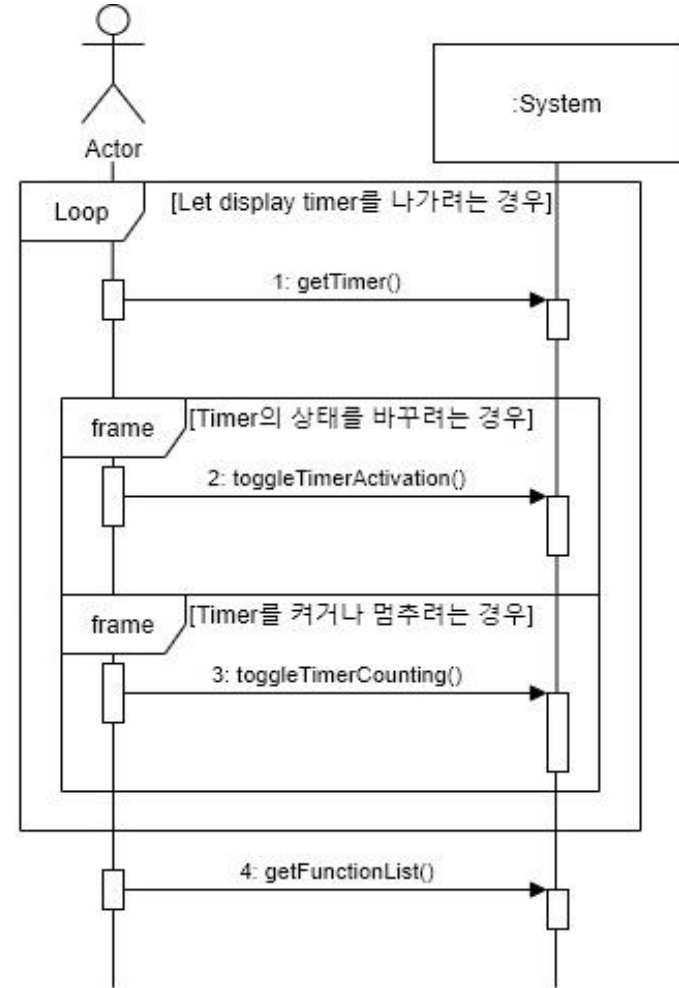
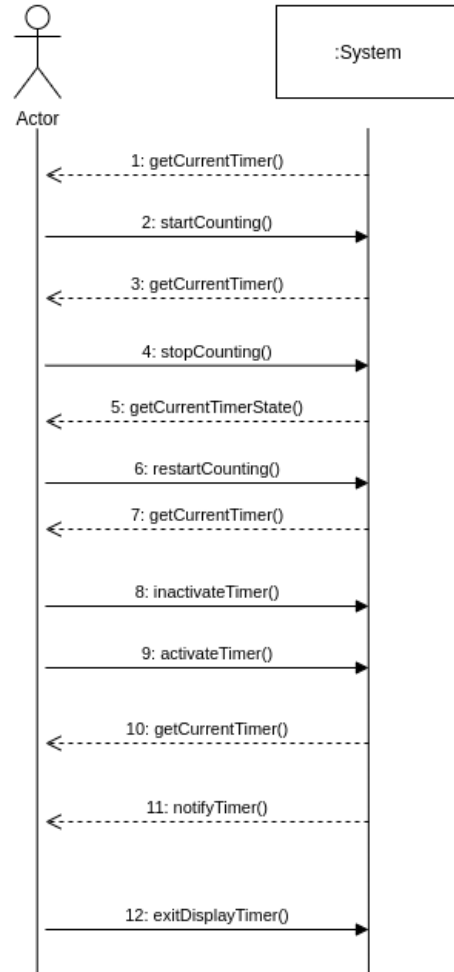
Activity 2044. Define Interaction Diagrams



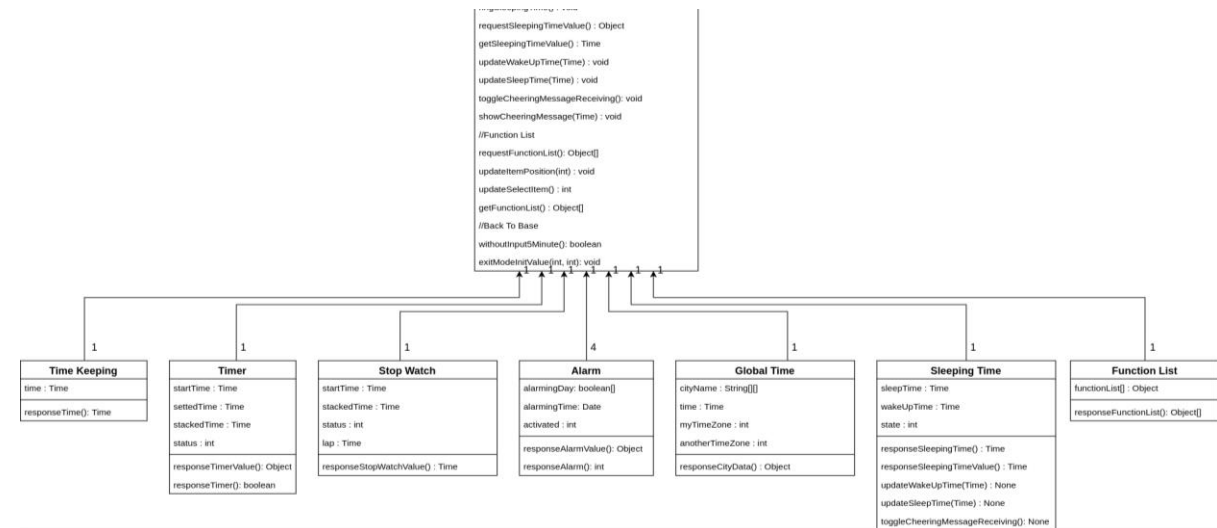
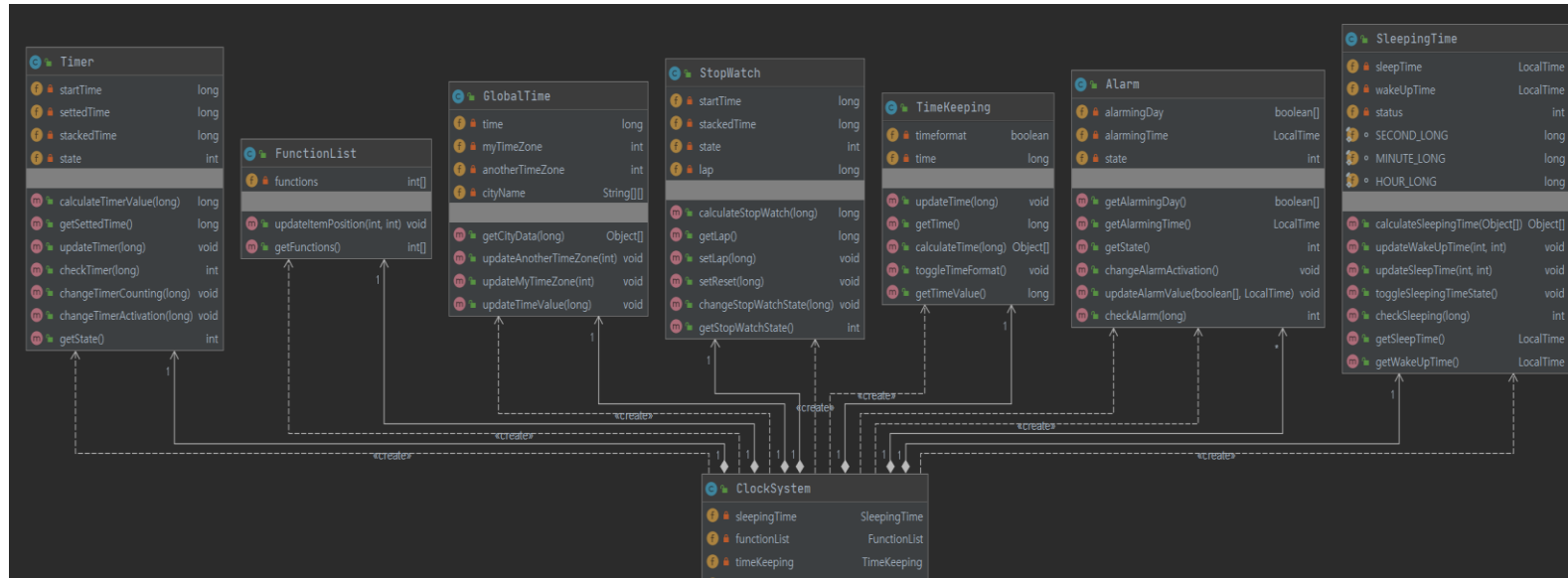
Activity 2033. Define System Sequence Diagrams

USE CASE: 3. Let display timer

1. (S)Set timer에서 설정된 시간을 화면에 보여준다.
2. (A)Counting의 시작을 요청한다 (B버튼을 누른다).
3. (S)Timer를 계산하고 진행 중인 시간을 화면에 표시한다.
4. (A)Counting의 일시정지/재시작을 요청한다 (B버튼을 누른다).
5. (S)사용자의 요청을 timer에 반영한다.
6. (A)timer의 활성화/비활성화를 요청한다 (D버튼을 누른다).
7. (S)사용자의 요청을 timer에 반영한다.
8. (S)timer가 끝나면 소리를 울린다. 이후 Set timer에서 설정된 시간을 화면에 보여준다.
9. (A)다음 function으로 넘어가기를 요청한다 (C버튼을 짧게 누른다).
10. (S)다음 선택된 function으로 이동한다.



Activity 2045. Define Design Class Diagrams



Activity 2055

Activity 2055. Write Unit Test Code

Timekeeping

```
1@Test
2public void 시간_설정_연도_증가_테스트(){
3    /**
4     * 만약 시간 설정 도중 2100년 위로 넘어간다면, 2000년 1월 1일로 돌아간다.
5     */
6
7     // 현재시간
8     long current = clockSystem.clock();
9     long currentTime = (long)timeKeeping.calculateTime(current)[0];
10    LocalDateTime currentLocalDateTime = Instant.ofEpochMilli(currentTime).atZone(ZoneId.of("+0")).toLocalDateTime();
11
12    int currentYear = currentLocalDateTime.getYear();
13    LocalDateTime futureLocalDateTime = currentLocalDateTime.plusYears(2100-currentYear);
14    long futureTime = futureLocalDateTime.atZone(ZoneId.of("+0")).toInstant().toEpochMilli();
15
16    timeKeeping.updateTime(futureTime-currentTime);
17
18    // 2000.01.01 00:00:00
19    LocalDateTime returnLocalDateTime = LocalDateTime.of(2000,1,1,0,0,0);
20    long returnTime = returnLocalDateTime.atZone(ZoneId.of("+0")).toInstant().toEpochMilli();
21
22    long testTime = (long)timeKeeping.calculateTime(current)[0];
23    LocalDateTime testLocalDateTime = Instant.ofEpochMilli(testTime).atZone(ZoneId.of("+0")).toLocalDateTime();
24
25    assertEquals(returnTime, testTime);
26 }
```


Activity 2055. Write Unit Test Code

Timer

```
1 @Test
2 public void 타이머_정지_테스트() throws Exception{
3
4     long currentTime = (long)clockSystem.getTime()[0];
5     timer.changeTimerActivation();
6
7     timer.updateTimer(this.settedTime);
8
9     currentTime = (long)clockSystem.getTime()[0];
10    timer.changeTimerCounting(currentTime);
11    this.startTime = currentTime;
12
13    Thread.sleep(5000);
14
15    currentTime = (long)clockSystem.getTime()[0];
16    timer.changeTimerCounting(currentTime);
17    assertEquals(1, timer.getState()); // 정지
18
19    this.stackedTime += currentTime - this.startTime;
20
21    long nowTimerTime = this.settedTime - this.stackedTime;
22
23    assertEquals(timer.calculateTimerValue(currentTime), nowTimerTime);
24 }
```

```
1 @Test
2 public void 타이머_활성화_및_시작_테스트(){
3     /**
4      * 비활성화 되어있는 타이머를 활성화시킨 뒤 (0 -> 1),
5      * 활성화 된 타이머를 2분으로 셋팅한 뒤 동작시킨다 (1 -> 2)
6      */
7
8     long currentTime = (long)clockSystem.getTime()[0];
9     timer.changeTimerActivation();
10    assertEquals(timer.getState(), 1);
11
12    timer.updateTimer(this.settedTime);
13
14    currentTime = (long)clockSystem.getTime()[0];
15    timer.changeTimerCounting(currentTime);
16    this.startTime = currentTime;
17
18    assertEquals(2, timer.getState());
19 }
```

Activity 2055. Write Unit Test Code

Timer

```
1@Test
2public void 타이머_재시작_테스트() throws Exception{
3    long currentTime = (long)clockSystem.getTime()[0];
4    timer.changeTimerActivation();
5
6    timer.updateTimer(this.settedTime);
7
8    currentTime = (long)clockSystem.getTime()[0];
9    timer.changeTimerCounting(currentTime);
10   this.startTime = currentTime;
11
12   Thread.sleep(5000);
13
14   currentTime = (long)clockSystem.getTime()[0];
15   timer.changeTimerCounting(currentTime);
16   assertEquals(1, timer.getState()); // 정지
17
18   this.stackedTime += currentTime - this.startTime;
19
20   Thread.sleep(2000);
21
22   currentTime = (long)clockSystem.getTime()[0];
23   timer.changeTimerCounting(currentTime);
24   this.startTime = currentTime;
25
26   assertEquals(2, timer.getState());
27
28   long nowTimerTime2 = this.settedTime - (currentTime - this.startTime + this.stackedTime);
29
30   assertEquals(timer.calculateTimerValue(currentTime), nowTimerTime2);
31}
```

Activity 2055. Write Unit Test Code

Stopwatch

```
1 @Test
2 public void 스톱워치_상태_테스트(){
3
4     stopwatch.changeStopWatchState((long)clockSystem.getTime()[0]);
5     boolean nowState = (stopwatch.getStopWatchState() == 1);
6     this.startTime = (long)clockSystem.getTime()[0];
7
8     assertEquals(true, nowState);
9 }
```

```
1 @Test
2 public void 스톱워치_랩_테스트(){
3
4     boolean nowState = (stopwatch.getStopWatchState() == 0);
5
6     long currentTime = (long)clockSystem.getTime()[0];
7     stopwatch.setLap(currentTime);
8
9     long lap = stopwatch.getLap();
10
11     assertEquals(true, nowState);
12     assertEquals(lap, 0);
13 }
```

```
1 @Test
2 public void 스톱워치_리셋_테스트(){
3     long currentTime = (long)clockSystem.getTime()[0];
4
5     stopwatch.changeStopWatchState(currentTime);
6     boolean nowState = (stopwatch.getStopWatchState() == 0);
7     this.startTime = currentTime;
8
9     stopwatch.setReset(currentTime);
10
11     assertEquals(false, nowState);
12     assertEquals(0, stopwatch.getLap());
13     assertEquals(0, stopwatch.calculateStopWatch(currentTime));
14 }
```

```
1 public void 스톱워치_재시작_테스트(){
2
3     long currentTime = (long)clockSystem.getTime()[0];
4
5     stopwatch.changeStopWatchState(currentTime);
6     boolean nowState = (stopwatch.getStopWatchState() == 1);
7     this.startTime = currentTime;
8
9     currentTime = (long)clockSystem.getTime()[0];
10
11     long calculateTime = stopwatch.calculateStopWatch(currentTime);
12     long testTime = currentTime - this.startTime + this.stackedTime;
13
14     assertEquals(true, nowState);
15     assertEquals(testTime, calculateTime);
16 }
```

Activity 2055. Write Unit Test Code

Stopwatch

```
1 @Test
2 public void 스톱워치_정지_테스트(){
3     long currentTime = (long)clockSystem.getTime()[0];
4     stopWatch.changeStopWatchState(currentTime);
5     boolean nowState = (stopWatch.getStopWatchState() == 0);
6     this.stackedTime = currentTime - this.startTime + this.stackedTime;
7
8     assertEquals(false, nowState);
9 }
```

```
1 @Test
2 public void 스톱워치_리셋_테스트(){
3     long currentTime = (long)clockSystem.getTime()[0];
4
5     stopWatch.changeStopWatchState(currentTime);
6     boolean nowState = (stopWatch.getStopWatchState() == 0);
7     this.startTime = currentTime;
8
9     stopWatch.setReset(currentTime);
10
11    assertEquals(false, nowState);
12    assertEquals(0, stopWatch.getLap());
13    assertEquals(0, stopWatch.calculateStopWatch(currentTime));
14 }
```

Activity 2055. Write Unit Test Code

Alarm

```
1 @Test
2 public void 알람_올리는지_테스트(){
3     //알람 값 설정 (기본값 08:00시, 일~토 모두 활성화된 상태로 설정)
4     alarm.updateAlarmValue(new boolean[]{true, true, true, true, true, true, true}, LocalTime.of(0, 0));
5     alarm.changeAlarmActivation();
6
7     //알람에 넣을 값 설정
8     LocalDateTime testTime = LocalDateTime.of(LocalDate.of(2020, 6, 8), LocalTime.of(8, 0));
9     long testLongValue = testTime.atZone(ZoneId.of("+0")).toInstant().toEpochMilli();
10
11    //알람이 울려야 하는지 테스트
12    int result = alarm.checkAlarm(testLongValue);
13
14    Assertions.assertEquals(2, result);
15 }
```

```
1 @Test
2 public void 알람_값_업데이트_테스트(){
3     //현재 알람 값에 8시간 더함
4     alarm.updateAlarmValue(new boolean[]{true, true, true, true, true, true, true},
5         LocalTime.of(8, 0));
6
7     //기본이 8시이므로, 알람은 16시여야 함.
8     int alarmHour = alarm.getAlarmingTime().getHour();
9
10    Assertions.assertEquals(16, alarmHour);
11 }
```

Activity 2055. Write Unit Test Code

Global time

```
1 @BeforeEach
2 public void 객체_초기화(){
3     clockSystem = new ClockSystem();
4     globalTime = new GlobalTime();
5 }
```

```
1 @Test
2 public void Global_Time_도시_데이터_받아오기_테스트(){
3     //globalTime에 넣을 값 설정
4     LocalDateTime testTime = LocalDateTime.of(LocalDate.of(2020, 6, 8), LocalTime.of(8, 0));
5     long testLongValue = testTime.atZone(ZoneId.of("+0")).toInstant().toEpochMilli();
6
7     //출력값 설정
8     Object[] result = globalTime.getCityData(testLongValue);
9     Object[] result0 = (Object[]) result[0];
10    String[] myCityName = (String[]) result0[2];
11    Object[] result1 = (Object[]) result[1];
12    String[] otherCityName = (String[]) result1[2];
13
14    //UTC가 기본은 9으로 설정되어 있으므로, 9에 해당되는 도시인 런던, 암스테르담, 파리를 가져와야 함
15    //다른 도시의 UTC는 0이므로, {"런던", "암스테르담", "파리"}와 같아야 함
16    Assertions.assertArrayEquals(new String[]{"도쿄", "서울", "평양"}, myCityName);
17    Assertions.assertArrayEquals(new String[]{"런던", "암스테르담", "파리"}, otherCityName);
18
19    //다른 도시의 UTC변경 -> 1 혹은 -1 만 넣어줄 수 있음
20    //6시간 늦춤, 다른 도시의 UTC는 +6이 되어야 함
21    for(int i = 0; i < 6; i++) globalTime.updateAnotherTimeZone(1);
22    result = globalTime.getCityData(testLongValue);
23    result1 = (Object[]) result[1];
24    otherCityName = (String[]) result1[2];
25
26    //다른 도시는 +6 UTC 이므로, 부탄, 비슈켄트, 알마티를 가져와야 함
27    Assertions.assertArrayEquals(new String[]{"부탄", "비슈켄트", "알마티"}, otherCityName);
28
29    //내도시를 30번 이동시킴
30    //지구 한바퀴를 돌아서 UTC +3이 되어야 함.
31    for(int i = 0; i < 30; i++) globalTime.updateMyTimeZone(1);
32    result = globalTime.getCityData(testLongValue);
33    result0 = (Object[]) result[0];
34    myCityName = (String[]) result0[2];
35
36    //내 도시는 UTC +12 이므로, 모스크바, 이스탄불, 나이로비 이어야 함
37    Assertions.assertArrayEquals(new String[]{"마셜 제도", "오클랜드", "투발루"}, myCityName);
38 }
```

Activity 2055. Write Unit Test Code

Sleeping time

```
1 @Test
2 public void 최소_수면시각_테스트(){
3
4     //기존에 7시간 30분이었던 수면시각을 30분으로 업데이트 한다
5
6     System.out.println(sleepingTime.getSleepTime());
7
8     for(int i = 0 ; i < 6 ; i++)
9         sleepingTime.updateSleepTime(1,0);
10    System.out.println(sleepingTime.getSleepTime());
11
12    LocalTime testSleepTime = sleepingTime.getSleepTime();
13    LocalTime returnSleepTime = LocalTime.of(1,30);
14
15    assertEquals(returnSleepTime, testSleepTime);
16 }
```

```
1 @Test
2 public void 적정_수면시간_반환_테스트_1(){
3
4     // 2020년 6월 8일 13시 31분 50초
5     Object[] dummyTime = new Object[]{1591590710000L, 1};
6     Object[] sleepTimes = sleepingTime.calculateSleepingTime(dummyTime);
7
8     LocalTime firstTime = (LocalTime) sleepTimes[0];
9     LocalTime secondTime = (LocalTime) sleepTimes[1];
10
11    assertEquals(firstTime.getHour(), 5);
12    assertEquals(firstTime.getMinute(), 10);
13    assertEquals(secondTime.getHour(), 6);
14    assertEquals(secondTime.getMinute(), 40);
15 }
```

```
1 @Test
2 public void 최대_수면시각_테스트(){
3
4     System.out.println(sleepingTime.getWakeUpTime());
5     //기존에 8시간 분이었던 수면시각을 12시간 30분으로 업데이트 한다
6     for(int i = 0 ; i < 5 ; i++)
7         sleepingTime.updateSleepTime(1,1);
8     System.out.println(sleepingTime.getSleepTime());
9     LocalTime testSleepTime = sleepingTime.getSleepTime();
10    LocalTime returnSleepTime = LocalTime.of(1,30);
11
12    assertEquals(returnSleepTime, testSleepTime);
13 }
```

Activity 2055. Write Unit Test Code

Sleeping time

```
1 @Test
2 public void 최소_수면시각_테스트(){
3
4     //기존에 7시간 30분이었던 수면시각을 30분으로 업데이트 한다
5
6     System.out.println(sleepingTime.getSleepTime());
7
8     for(int i = 0 ; i < 6 ; i++)
9         sleepingTime.updateSleepTime(1,0);
10    System.out.println(sleepingTime.getSleepTime());
11
12    LocalDateTime testSleepTime = sleepingTime.getSleepTime();
13    LocalDateTime returnSleepTime = LocalDateTime.of(1,30);
14
15    assertEquals(returnSleepTime, testSleepTime);
16 }
```

```
1 @Test
2 public void 기상시각_업데이트_테스트(){
3     //기존에 8시 10분으로 초기화 된 기상시각을 9시로 업데이트한다.
4     sleepingTime.updateWakeUpTime(1,1);
5     for(int i = 0 ; i < 10 ; i ++ )
6         sleepingTime.updateWakeUpTime(0, 0);
7
8     LocalDateTime testWakeUpTime = sleepingTime.getWakeUpTime();
9     LocalDateTime returnWakeUpTime = LocalDateTime.of(9,0);
10
11    assertEquals(returnWakeUpTime, testWakeUpTime);
12
13 }
```

```
1 @Test
2 public void 응원_메세지_활성화_테스트(){
3     /**
4      * 현재 응원 메세지 송신이 활성화 되었는지 안 되었는지 확인한다.
5      * 활성화 되어있지 않다가 활성화 됨
6      */
7
8     sleepingTime.toggleSleepingTimeState();
9     int state = sleepingTime.checkSleeping((long)clockSystem.getTime()[0]);
10
11    assertEquals(state, 0);
12 }
```


Activity 2055. Write Unit Test Code

Function list

```
1 @BeforeEach
2 public void 객체_초기화(){
3     clockSystem = new ClockSystem();
4     functionList = new FunctionList();
5 }
```

```
1 @Test
2 public void FunctionList_값_바꾸기_테스트(){
3
4     //FunctionList의 최좌측 값을 최우측 값과 변경
5     functionList.updateItemPosition(0, -1);
6     Assertions.assertArrayEquals(new int[]{5, 1, 2, 3, 4, 0}, functionList.getFunctions());
7
8     //FunctionList의 두번째 값을 한 칸 아래로 변경
9     functionList.updateItemPosition(1, 1);
10    Assertions.assertArrayEquals(new int[]{5, 2, 1, 3, 4, 0}, functionList.getFunctions());
11 }
```

Activity 2061

Activity 2061. Unit Testing

Timekeeping

▼ ✓ controller.mode.TimeKeepingTest	0 ms
✓ 시간_설정_연도_증가_테스트()	0 ms
▼ ✓ controller.mode.TimerTest	12 s 17 ms

Timer

▼ ✓ controller.mode.TimerTest	12 s 17 ms
✓ 타이머_재시작_테스트()	7 s 9 ms
✓ 타이머_정지_테스트()	5 s 8 ms
✓ 타이머_활성화_및_시작_테스트()	0 ms

Activity 2061. Unit Testing

Stop watch

▼ ✓ controller.mode.StopWatchTest	2 ms
✓ 스탑워치_랩_테스트()	1 ms
✓ 스탑워치_리셋_테스트()	0 ms
✓ 스탑워치_상태_테스트()	0 ms
✓ 스탑워치_재시작_테스트()	1 ms
✓ 스탑워치_정지_테스트()	0 ms

Alarm

▼ ✓ controller.mode.AlarmTest	16 ms
✓ 알람_값_업데이트_테스트()	0 ms
✓ 알람_울리는지_테스트()	16 ms

Global time

▼ ✓ controller.mode.GlobalTimeTest	1 ms
✓ Global_Time_도시_데이터_받아오기_테스트()	1 ms

Activity 2061. Unit Testing

Sleeping Time

▼ ✓ controller.mode.SleepingTimeTest	5 ms
✓ 기상시각_업데이트_테스트()	1 ms
✓ 수면시각_업데이트_테스트()	1 ms
✓ 응원_메세지_활성화_테스트()	1 ms
✓ 적정_수면시간_반환_테스트_1()	0 ms
✓ 적정_수면시간_반환_테스트_2()	0 ms
✓ 최대_수면시각_테스트()	2 ms
✓ 최소_수면시각_테스트()	0 ms

Function List

▼ ✓ controller.mode.FunctionListTest	1 ms
✓ FunctionList_값_바꾸기_테스트()	1 ms

Activity 2063

Activity 2063. System Testing

Test No.	Ref No.	Test case	Description	Use case	P/F
1	1	yearRangeTest	연도의 underflow, overflow, 그리고 예외처리 테스트	2. Set time	P
2	2	timeRangeTest	월, 일, 시, 분, 초 각각에 대하여 최솟값/최댓값 이후의 처리 테스트		P
3	2	toNextFunctionTest	Inactive 상태가 아닐 때 set timer로 이동하는지에 대한 테스트		P
4	2	timerRangeTest	시간의 underflow, overflow, 그리고 예외처리 테스트	3. Let display timer	P
5	2	timerTimeValueTest	시, 분, 초 각각에 대하여 최솟값/최댓값 이후의 처리 테스트	4. Set timer	P
6	2	timerInactiveTest	Timer가 inactive 상태에서 B 버튼이 동작하지 않는지에 대한 테스트	5. Change timer state 6. Change timer counting	P
7	3	stopWatchFunctionTest	Stop watch의 active, pause, reset, lap이 잘 동작하는지 테스트	9. Change stop watch state 10. Reset stop watch 11. Lap stop watch	P
8	3	stopWatchOverflowTest	Stop watch 시간의 overflow 테스트	12. Calculate stop watch	F
9	4	alarmDayOfWeek	Alarm 설정 시 요일 선택에 따른 동작 테스트	14. Set alarm	F
10	4	sameAlarmTimeTest	같은 시간에 alarm을 설정했을 때의 동작 테스트	16. Buzz alarm 17. Show alarm	F
11	4	timerAndAlarmTest	Timer와 alarm이 울리는 시간이 겹칠 때의 동작 테스트		F

Activity 2063. System Testing

Test No.	Ref No.	Test case	Description	Use case	P/F
12		sameTimeZoneTest	두 시간대가 같을 때에 대한 동작 테스트	20. Let global time	P
13		checkTimeUpdatedTest	Set time에 대한 변경 결과가 제대로 반영되는지 테스트		P
14		printSleepingTimeTest	가장 가까운 두 최적 수면 시간이 계산되어 화면에 출력되는지 테스트	22. Set wake up time	P
15		wakeUpTimeRangeTest	시, 분 각각에 대한 최솟값, 최댓값 이후의 처리 테스트		P
16		timerAndCheeringMsgTest	cheering message와 timer가 울리는 시각이 겹칠 때의 동작 테스트	24. Buzz cheering message 25. Show cheering message	F
17		toFirstFunctionTest	C 버튼을 눌렀을 때 설정한 첫 function으로 이동하는지에 대한 테스트	29. Control function position	P
18		timeOutTest	Set XXX에서 5분 정도 흘렀을 때 자동으로 Display XXX로 이동하는지에 대한 테스트	30. Back to base	F
19		5minutePressTest	Set XXX에서 어떤 버튼을 꼭 누르고 있었을 때의 동작 테스트		F
20		alarmInterruptTest	중간에 알람이 울릴 때의 동작 테스트		F
21		cheeringMsgInterruptTest	중간에 cheering message가 올 때의 동작 테스트		F

Activity 2066

Activity 2066. Testing Traceability Analysis

Function Requirement	Match	Use Case	Using	No.	Operation in SSD	Using	No.	Operation int Interaction Diagram	Using	No.	Method	Class	Using	No	Unit Test	
R 1.1 Display time	1.1	1. Let display time	1, 3, 25	1	getTime()	1, 35	1	getTime()	1	1	getTime()	System			1 시간 설정 연도 증가 테스트	
R 1.2 Set time	1.2	2. Set time	1, 2	2	setTime()	2, 37	2	setTime()	2	2	setTime()					2 타이머 활성화 및 시작 테스트
R 2.1 Display timer	2.1	3. Let display timer	4, 6, 7, 25	3	changeTimeFormat()	3, 36	3	changeTimeFormat()	3	3	changeTimeFormat()					3 타이머 정지 테스트
R 2.2 Set timer	2.2	4. Set timer	4, 5	4	getTime()	5, 39, 43	4	getTime()	4	4	getTime()					4 타이머 재시작 테스트
R 2.3 Change timer state	2.3	5. Change timer state	6	5	setTimer()	10, 45	5	setTimer()	5	5	setTimer()					5 스태틱 맵 테스트
R 2.4 Change timer counting	2.4	6. Change timer counting	7	6	toggleTimerActivation()	7, 41	6	toggleTimerActivation()	6	6	toggleTimerActivation()					6 스태틱 맵 테스트
R 2.5 Buzz timer	2.5	7. Buzz timer		7	toggleTimerCounting()	8, 42	7	toggleTimerCounting()	7	7	toggleTimerCounting()					7 스태틱 맵 테스트
R 3.1 Display stop watch	3.1	8. Let display stop watch	8, 9, 10, 11, 12, 25	8	getStopWatchTime()	11, 46, 47	8	getStopWatchTime()	8	8	getStopWatchTime()					8 스태틱 맵 테스트
R 3.2 Change stop watch state	3.2	9. Change stop watch state	12	9	lapStopWatch()	12, 48	9	lapStopWatch()	9	9	lapStopWatch()					9 알람 울리는지 테스트
R 3.3 Reset stop watch	3.3	10. Reset stop watch	10	10	resetStopWatch()	14, 51	10	resetStopWatch()	10	10	resetStopWatch()					10 알람 값 업데이트 테스트
R 3.4 Lap stop watch	3.4	11. Lap stop watch	9	11	checkStopWatchState()	12, 48	11	checkStopWatchState()	11	11	checkStopWatchState()					11 Global_time 도시 데이터 받아오기 테스트
R 3.5 Calculate stop watch	3.5	12. Calculate stop watch		12	toggleStopWatchState()	13, 49	12	toggleStopWatchState()	12	12	toggleStopWatchState()					12 수면시간 업데이트 테스트
R 4.1 Display alarm	4.1	13. Let display alarm	13, 15, 25	13	getAlarm()	18, 53, 56, 57	13	getAlarm()	13	13	getAlarm()					13 기상시간 업데이트 테스트
R 4.2 Set alarm	4.2	14. Set alarm	13, 14	14	setAlarm()	20, 55	14	setAlarm()	14	14	setAlarm()					14 최초 수면시간 테스트
R 4.3 Change active alarm	4.3	15. Change active alarm	15	15	toggleAlarmActivation()	19, 54	15	toggleAlarmActivation()	15	15	toggleAlarmActivation()					15 최소 수면시간 테스트
R 4.4 Buzz alarm	4.4	16. Buzz alarm		16	enterGlobalTime()	28, 38, 58	16	enterGlobalTime()	16	16	enterGlobalTime()					16 적장 수면시간 반환 테스트 1
R 4.5 Show alarming	4.5	17. Show alarming		17	ringAlarmTime()	21, 59	17	ringAlarmTime()	17	17	ringAlarmTime()					17 적장 수면시간 반환 테스트 2
R 4.6.1 Turn off alarm manually	4.6.1	18. Turn off alarm manually		18	setMyTimeZone()	22, 60	18	setMyTimeZone()	18	18	setMyTimeZone()					18 홈월 메세지 활성화 테스트
R 4.6.2 Turn off alarm automatically	4.6.2	19. Turn off alarm automatically		19	setAnotherTimeZone()	23, 61	19	setAnotherTimeZone()	19	19	setAnotherTimeZone()					19 FunctionList 값 바꾸기 테스트
R 5.1 Global time	5.1	20. Let global time	16, 17, 18, 19, 25	20	getSleepingTime()	29, 68	20	getSleepingTime()	20	20	getSleepingTime()					
R 6.1 Display sleeping time	6.1	21. Let display sleeping time	20, 24, 25	21	getSleepingTimeValue()	64, 65	21	getSleepingTimeValue()	21	21	getSleepingTimeValue()					
R 6.2 Set wake up time	6.2	22. Set wake up time	21, 22, 23	22	setWakeUpTime()	31, 66	22	setWakeUpTime()	22	22	setWakeUpTime()					
R 6.3 Change cheering message receiving	6.3	23. Change cheering message receiving	24	23	setSleepTime()	32, 67	23	setSleepTime()	23	23	setSleepTime()					
R 6.4 Buzz cheering message	6.4	24. Buzz cheering message		24	toggleCheeringMessageReceiving()	24, 62	24	toggleCheeringMessageReceiving()	24	24	toggleCheeringMessageReceiving()					
R 6.5 Show cheering message	6.5	25. Show cheering message		25	getNowSleepingList()	27, 69	25	getNowSleepingList()	25	25	getNowSleepingList()					
R 6.6.1 Turn off cheering message manually	6.6.1	26. Turn off cheering message manually		26	moveItem()	33, 70	26	moveItem()	26	26	moveItem()					
R 6.6.2 Turn off cheering message automatically	6.6.2	27. Turn off cheering message automatically		27	getFunctionList()		27	getFunctionList()	27	27	getFunctionList()					
R 7.1 Display function list	7.1	28. Let display function list	25, 26	28	enterGlobalTime()		28	enterGlobalTime()	28	28	enterGlobalTime()					
R 7.2 Control function position	7.2	29. Control function position	26	29	getSleepingTime()		29	getSleepingTime()	29	29	getSleepingTime()					
R 8.1 Back to base	8.1	30. Back to base		30	getSleepingTimeValue()		30	getSleepingTimeValue()	30	30	getSleepingTimeValue()					
				31	setWakeUpTime()		31	setWakeUpTime()	31	31	setWakeUpTime()					
				32	setSleepTime()		32	setSleepTime()	32	32	setSleepTime()					
				33	moveItem()		33	moveItem()	33	33	moveItem()					
				34	clock()		34	clock()	34	34	clock()					
				35	calculateTime()		35	calculateTime()	35	35	calculateTime()	Time keeping		1		
				36	toggleTimeFormat()		36	toggleTimeFormat()	36	36	toggleTimeFormat()					
				37	updateTime()		37	updateTime()	37	37	updateTime()					
				38	getTimeValue()		38	getTimeValue()	38	38	getTimeValue()					
				39	calculateTimerValue()		39	calculateTimerValue()	39	39	calculateTimerValue()			2		
				40	checkTimer()		40	checkTimer()	40	40	checkTimer()			3		
				41	changeTimerActivation()		41	changeTimerActivation()	41	41	changeTimerActivation()			4		
				42	changeTimerCounting()		42	changeTimerCounting()	42	42	changeTimerCounting()	Timer				
				43	getState()		43	getState()	43	43	getState()					
				44	getSettedTime()		44	getSettedTime()	44	44	getSettedTime()					
				45	updateTimer()		45	updateTimer()	45	45	updateTimer()					
				46	calculateStopWatchTime()		46	calculateStopWatchTime()	46	46	calculateStopWatchTime()			5		
				47	getLap()		47	getLap()	47	47	getLap()	Stop watch		6		
				48	getStopWatchState()		48	getStopWatchState()	48	48	getStopWatchState()			7		
				49	changeStopWatchState()		49	changeStopWatchState()	49	49	changeStopWatchState()			8		
				50	setLap()		50	setLap()	50	50	setLap()					
				51	setReset()		51	setReset()	51	51	setReset()					
				52	checkAlarm()		52	checkAlarm()	52	52	checkAlarm()			9		
				53	getAlarmingDay()		53	getAlarmingDay()	53	53	getAlarmingDay()			10		
				54	changeAlarmActivation()		54	changeAlarmActivation()	54	54	changeAlarmActivation()	Alarm				
				55	updateAlarmValue()		55	updateAlarmValue()	55	55	updateAlarmValue()					
				56	getAlarmingTime()		56	getAlarmingTime()	56	56	getAlarmingTime()					
				57	getState()		57	getState()	57	57	getState()					
				58	updateTimeValue()		58	updateTimeValue()	58	58	updateTimeValue()			11		
				59	getCityData()		59	getCityData()	59	59	getCityData()	Global time				
				60	updateMyTimeZone()		60	updateMyTimeZone()	60	60	updateMyTimeZone()					
				61	updateAnotherTimeZone()		61	updateAnotherTimeZone()	61	61	updateAnotherTimeZone()					
				62	toggleSleepingTimeState()		62	toggleSleepingTimeState()	62	62	toggleSleepingTimeState()			12		
				63	checkSleeping()		63	checkSleeping()	63	63	checkSleeping()			13		
				64	getSleepTime()		64	getSleepTime()	64	64	getSleepTime()			14		
				65	getWakeUpTime()		65	getWakeUpTime()	65	65	getWakeUpTime()			15		
				66	updateWakeUpTime()		66	updateWakeUpTime()	66	66	updateWakeUpTime()			16		
				67	updateSleepTime()		67	updateSleepTime()	67	67	updateSleepTime()			17		
				68	calculateSleepingTime()		68	calculateSleepingTime()	68	68	calculateSleepingTime()			18		
				69	getFunctions()		69	getFunctions()	69	69	getFunctions()			19		
				70	updateItemPosition()		70	updateItemPosition()	70	70	updateItemPosition()	Function list				
				71	showAlarming()		71	showAlarming()	71	71	showAlarming()					
				72	backToPrev()		72	backToPrev()	72	72	backToPrev()					
				73	showCheeringMessage()		73	showCheeringMessage()	73	73	showCheeringMessage()					
				74	withoutInput5Minute()		74	withoutInput5Minute()	74	74	withoutInput5Minute()	UI				