객체지향개발방법론

OOAD 2040
(Object-Oriented Design)

201212519 컴퓨터공학과 김선우
201510624 컴퓨터공학과 김용현
201614157 컴퓨터공학과 김도연
201614158 컴퓨터공학과 장다혜

DWS Define Requirements

< Functional Requirements >

1. 현재 시간
   - 사용자가 나라, 날짜 및 시간 설정을 한다.
   - 현재 날짜를 출력한다.
   - 현재 시간을 출력한다.
   - 오후 시간 표시는 24시로 표현한다.

2. 스탑워치 기능
   - 시간의 경과를 알려준다.
   - 1/100초 단위로 측정한다.
   - 일시정지/재시작이 가능하다.

3. 알람 기능
   - 알람 시간은 4개까지 저장 가능하다.
   - 알람은 3초 후에 꺼지거나, 버튼을 누르면 꺼진다.
   - 알람 기능을 켜면 화면에 표시된다.

4. 타이머 기능
   - 타이머 시간을 설정한다.
   - 시간이 다되면 알람이 울린다.
   - 타이머 일시정지 가능하다.
   - 시간이 초과되면 초기화된다.

5. 세계시간 기능
   - 한국, 미국, 영국, 중국, 러시아, 싱가폴의 시간을 제공할 수 있다.
   - 선택한 나라의 날짜를 출력한다.
   - 선택한 나라의 시간을 출력한다.

6. 밝기 조절 기능
   - 시간에 따라 화면 밝기가 자동 조절된다.
   - 밝기 수준을 선택할 수 있다.

7. 기능 선택 기능
   - 6가지의 기능 중 4가지를 선택할 수 있다.
   - 기능 변경이 되어도 알람 시간은 항상 저장된다.

Activity 2041 :
Design Real Use Cases

Use case	**1. Set Time**
Actor	User
Type	Evident
Pre-Requisites	currentMode의 값이 TKM이어야 한다.
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) : User가 A 버튼을 누른다. (S) : enterSetCountry()을 호출한다.(한국, 미국, 영국, 중국, 러시아, 싱가폴) (S) : display(country[i])를 호출한다. 1.1 (A) : 사용자가 B 버튼을 눌러 country배열 내에서 순환 선택한다. (S) : display(country[i+1])를 호출한다. 1.2 (A) : 사용자가 D 버튼을 눌러 country배열 내에서 역순환 선택된다. (S) : display(country[i-1])를 호출한다. 2. (A) : User가 A 버튼을 누른다. (S) : enterSetYear()를 호출한다. 2.1 (A) : 사용자가 B 버튼을 눌러 year값을 증가시킨다. (S) : display(year+1)을 호출한다. 2.2 (A) : 사용자가 D 버튼을 눌러 year값을 감소시킨다. (S) : display(year-1)을 호출한다. 3. (A) : User가 A버튼을 누른다. (S) : enterSetMonth()를 호출한다. 3.1 (A) : 사용자가 B 버튼을 눌러 month값을 증가시킨다. (S) : display(month+1)을 호출한다.

Typical Courses of Events	3.2 (A) : 사용자가 D 버튼을 눌러 month값을 감소시킨다. (S) : display(month-1)을 호출한다. 4. (A) : User가 A버튼을 누른다. (S) : enterSetDay()를 호출한다. 4.1 (A) : 사용자가 B 버튼을 눌러 day값을 증가시킨다. (S) : display(day+1)을 호출한다. 4.2 (A) : 사용자가 D 버튼을 눌러 day값을 감소시킨다. (S) : display(day-1)을 호출한다. 5. (A) : User가 A버튼을 누른다. (S) : enterSetHour()를 호출한다. 5.1 (A) : 사용자가 B 버튼을 눌러 hour값을 증가시킨다. (S) : display(hour+1)을 호출한다. 5.2 (A) : 사용자가 D 버튼을 눌러 hour값을 감소시킨다. (S) : display(hour-1)을 호출한다. 6. (A) : User가 A버튼을 누른다. (S) : enterSetMinute()를 호출한다. 6.1 (A) : 사용자가 B 버튼을 눌러 minute값을 증가시킨다. (S) : display(minute+1)을 호출한다. 6.2 (A) : 사용자가 D 버튼을 눌러 minute값을 감소시킨다. (S) : display(minute-1)을 호출한다. 7. (A) : User가 A버튼을 누른다. (S) : enterSetSecond()를 호출한다. 7.1 (A) : 사용자가 B 버튼을 눌러 second값을 증가시킨다. (S) : display(second+1)을 호출한다.

Typical Courses of Events	7.2 (A) : 사용자가 D 버튼을 눌러 second값을 감소시킨다. (S) : display(second-1)을 호출한다. 8. (A) : User가 C버튼을 누른다. (S) : Timekeeping 모드로 돌아간다.
Alternative Courses of Events	A1. 각 설정화면에서 C 버튼을 누르면 Timekeeping 모드로 돌아간다.
Exceptional Courses of Event	E1. year값이 2100 일 때, B 버튼을 눌러 증가시키면 2000이 된다. 2000 일 때, D 버튼을 눌러 감소시키면 2100이 된다. E2. month값이 12 일 때, B 버튼을 눌러 증가시키면 1 이 된다. 1 일 때, D 버튼을 눌러 감소시키면 12 이 된다. E3. day값이 month 값에 따라 30, 31 일 때, B 버튼을 눌러 증가시키면 1 이 된다. month 값에 따라 1 일 때, D 버튼을 눌러 감소시키면 30, 31 이 된다. E4. hour값이 24H 일 때, B 버튼을 눌러 증가시키면 1H 이 된다. 1H 일 때, D 버튼을 눌러 감소시키면 24H 이 된다. E5. minute값이 59 일 때, B 버튼을 눌러 증가시키면 0 이 된다. 0 일 때, D 버튼을 눌러 감소시키면 59 이 된다. E6. second값이 59 일 때, B 버튼을 눌러 증가시키면 0 이 된다. 0 일 때, D 버튼을 눌러 감소시키면 59 이 된다.

Use case	**2. Show Current Time**
Actor	System
Type	Hidden
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) : display(currentTime)을 호출한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	N/A

Use case	**3. Show Current Date**
Actor	System
Type	Hidden
Pre-Requisites	N/A
Typical Courses of Events	(A) : Actor, (S) : System 1.(S) : display(currentDate)를 보여준다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	E1. year값이 최대값(2100)을 초과할경우, year값을 초기화(2020)한다.

Use Case	**4. Change mode**
Actor	User
Type	Evident
Pre requisites	각 선택된 모드의 첫 진입 상태여야 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 C 버튼을 눌러 changeMode()를 호출한다. 2. (S) display(selectedSw[i+1])을 호출한다.(i는 현재 선택된 모드의 인덱스이다) 2-1. Timekeeping (S) display(currentTime)를 호출한다. (S) display(alarmIndicator)를 호출한다. (S) display(country[i])를 호출한다. 2-2 Stopwatch (S) stopwatchTime변수값을 확인한다. (S) display(stopwatchTime)를 호출한다. (S) display(currentTime)를 호출한다.

Typical courses of events	2-3 Timer (S) timerTime 변수값을 확인한다. (S) display(timerTime)을 호출한다. (S) display(currentTime)을 호출한다. 2-4 Alarm (S) 알람 목록에 알람이 저장되어 있는지 확인한다. (S) display(alarmArray[0])을 호출한다. (S) display(currentTime)을 호출한다. 2-5 World time (S) display(worldTime[0])을 호출한다. (S) display(worldName[0])을 호출한다. (S) display(currentTime)을 호출한다. 2-6 Brightness Control (S) brightness, brightLevel 변수값을 확인한다. (S) display(brightLevel)을 호출한다. (S) display(brightness)을 호출한다. (S) display(currentTime)을 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1. selectedSw[]의 인덱스 범위를 초과하게 되면, display(selectedSw[0])를 호출한다. E2. 저장되어 있는 알람이 없을 때, 알람 모드로 진입 시, 1. --:-- 를 출력한다.

Use Case	**5. Stop ringing**
Actor	User
Type	Evident
Pre requisites	Ring onOff 변수값이 true여야한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) : ring이 울릴 때, User는 아무 버튼(A, B, C, D 버튼)이나 누른다. 2. (S) : check3Second()를 호출하고, User의 요청이 들어오면 onOff값을 false로 변경하여 ring을 멈춘다.
Alternative courses of events	A1. ring이 울린지 3초가 지났을 때, onOff값을 false로 변경하고 자동적으로 ring을 멈춘다.
Exceptional courses of events	N/A

Use Case	**6. Start stopwatch**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 STW이어야 한다. 2. Stopwatch의 onOff 변수가 false 상태이어여 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) : User가 B(start)버튼을 누른다. (S) startStopwatch()를호출한다. 2. (S) : display(stopwatchTime)를 보여준다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use Case	**7. Stop stopwatch**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 STW이어야 한다. 2. Stopwatch의 onOff 변수가 true 상태이어여 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 B(stop) 버튼을 누른다. (S) stopStopwatch()를 호출한다. 2. (S) display(stopwatchTime)을 보여준다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use Case	**8. Reset stopwatch**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 STW이어야 한다. 2. Stopwatch의 onOff 변수가 false 상태이어여 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 D(reset) 버튼을 누른다. (S) resetStopwatch()를 호출한다. 2. (S) display(stopwatchTime)를 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1.스톱워치가 진행 중이면 D(reset) 버튼을 눌러도 초기화되지 않는다.

Use Case	**9. Lap stopwatch**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 STW이어야 한다. 2. Stopwatch의 onOff 변수가 true 상태이어여 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 A(lap) 버튼을 누른다. (S) lapTime()를 호출한다. 3. (S) display(lapTime)을 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1. 기존의 STW는 뒤에서 계속 돌고있어야 한다.

Use Case	**10. Select alarm num**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 ALM이어야 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 B(select) 버튼을 누른다. (S) getAlarmTime(i)를 호출한다. 2. (S) display(alarmArray[i])을 호출한다.
Alternative courses of events	A1. 원하는 알람번호가 나올때까지 반복한다.
Exceptional courses of events	N/A

Use Case	**11. Remove alarm num**
Actor	User
Type	Evident
Pre requisites	1. currentMode의 값이 ALM이어야 한다.
Typical courses of events	(A) : Actor, (S) : System 1. (A) User가 D(remove) 버튼을 누른다. (S) removeAlarmNum(i)를 호출한다. 2. (S) display(defaultTime)를 호출한다.
Alternative courses of events	A1. alarmArray 내부의 값이 비어있을 경우, AlarmIndicator를 false로 업데이트한다.
Exceptional courses of events	N/A

Use Case	**12. Set Alarm Time**
Actor	User
Type	Evident
Pre-Requisites	1. currentMode의 값이 ALM이어야 한다.
Typical courses of events	(A): Actor, (S): System 1. (A) User가 A 버튼을 누른다. (S) enterSetSecond()을 호출한다. 1.1 (A) User가 B버튼을 누른다. (S) second 변수값이 증가한다. 1.2 (A) User가 D 버튼을 누른다. (S) seconde 변수값이 감소한다. 2. (A) User가 A 버튼을 누른다. (S) enterSetMinute()을 호출한다. 2.1 (A) User가 B버튼을 누른다. (S) minute 변수값이 증가한다.

Typical courses of events	2.2 (A) User가 D버튼을 누른다. (S) minute 변수값이 감소한다. 3. (A) User가 A버튼을 누른다. (S) enterSetHour()을 호출한다. 3.1 (A) User가 B버튼을 누른다. (S) Hour 변수값이 증가한다. 3.2 (A) User가 D버튼을 누른다. (S) Hour 변수값이 감소한다. 4. (A) User가 C버튼을 누른다. (S) setAlarmTime(alarmArray[i])을 호출한다. 5. (S) display(alarmArray[i])를 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1. 각 영역의 값이 최대값보다 커질 경우 0으로 0보다 작아질 경우는 최대값으로 한다 E2. 선택 가능한 영역에서 A버튼을 한번 더 누르면 다시 초기 영역으로 돌아온다. E3. alarmArray[i]의 값이 설정이 안됐을 경우 초기 설정값은 –시 --분 --초이다

Use Case	**13. Turn on Alarm**
Actor	System
Type	Hidden
Pre-Requisites	User가 remove alarm time을 했을 경우
Typical courses of events	(S): System 1. (S) checkAlarmArray()를 호출한다. 2. (S) display(alarmIndicator)를 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use Case	**14.Turn off Alarm**
Actor	System
Type	Hidden
Pre-Requisites	N/A
Typical courses of events	(S): System 1. (S) alarmArray를 확인하여 저장된 Time이 없는 경우, alarmIndicator의 값을 false로 변경한다. (checkAlarmArray()를 호출한다.) 2. (S) alarmIndicator의 값을 false로 변경한다. (turnOffAlarmIndicator()를 호출한다.)
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use Case	**15. Set timer time**
Actor	User
Type	Evident
Pre-Requisites	1. currentMode의 값이 TMR이어야 한다. 2. Timer 개체의 onOff 값이 false이어야 한다.
Typical courses of events	(A): Actor, (S): System 1. (A) User가 A 버튼을 누른다. (S) Set second()를 호출한다. 1-1. (A) User가 B 버튼을 누른다. (S) second의 값이 증가한다.

Typical courses of events	1-2. (A) User가 D 버튼을 누른다. (S) second의 값이 감소한다. 2. (A) User가 A 버튼을 누른다. (S) Set minute()를 호출한다. 2-1. (A) User가 B 버튼을 누른다. (S) minute의 값이 증가한다. 2-2. (A) User가 D 버튼을 누른다. (S) minute의 값이 감소한다. 3. (A) User가 A 버튼을 누른다. (S) Set minute()를 호출한다. 3-1. (A) User가 B 버튼을 누른다. (S) hour의 값이 증가한다.

Typical courses of events	3-2. (A) User가 D 버튼을 누른다. (S) hour의 값이 감소한다. 4. (A) 사용자가 C 버튼을 누른다. (S) 저장된 Time 객체를 DWS controller의 alarmArray에 저장한다. (saveAlarmTime()을 호출한다.) (S) display(timerTime)을 호출한다.
Alternative courses of events	A1. 각 설정화면에서 C 버튼을 누르면 timerMode 초기화면으로 돌아간다.
Exceptional courses of events	E1. second의 최대값(59)에서 B 버튼을 누르면 변수값을 0으로 변경한다. E2. minute의 최대값(59)에서 B 버튼을 누르면 변수값을 0으로 변경한다. E3. hour의 최대값(23)에서 B 버튼을 누르면 변수값을 0으로 변경한다.

Use Case	**16. Start timer**
Actor	User
Type	Evident
Pre-Requisites	1. currentMode의 값이 TMR이어야 한다. 2. Timer 개체의 onOff 값이 false이어야 한다.
Typical courses of events	(A): Actor, (S): System 1. (A) User가 B버튼을 누른다. (S) timer 스레드를 생성하여 startTimer(timerTime)을 호출한다. (S) Timer 객체의 onOff 값을 true로 변경한다. 2. (S) timer 스레드는 timerTime의 값이 0이 되는 시점에 없어진다.
Alternative courses of events	A1. count 스레드는 메인 스레드와 관계없이 동작한다.

Use Case	**17. Stop timer**
Actor	User
Type	Evident
Pre-Requisites	1. currentMode의 값이 TMR이어야 한다. 2. Timer 개체의 onOff 값이 true이어야 한다.
Typical courses of events	(A): Actor, (S): System 1. (A) User가 B 버튼을 누른다. (S) timer 스레드에서 timerTime을 저장하고, startTimer()를 멈추는 메소드인 stopTimer()를 호출한다. (S) Timer 객체의 onOff 값을 false로 변경한다.
Alternative courses of events	A1. count 스레드는 메인 스레드와 관계없이 동작한다.

Use Case	**18. Reset timer**
Actor	User
Type	Evident
Pre-Requisites	1. currentMode의 값이 TMR이어야 한다. 2. Timer 개체의 onOff 값이 false이어야 한다.
Typical courses of events	(A): Actor, (S): System 1. (A) User가 D 버튼을 누른다. (S) count 스레드는 timerTime을 초기화하고, 종료된다.(resetTimer()을 호출한다.) (S) display(timerTIme)을 호출한다.
Alternative courses of events	N/A

Use Case	**19. Ringing timer**
Actor	System
Type	Hidden
Pre-Requisites	N/A
Typical courses of events	1. (S) checkAlarm(), checkTimer()을 호출한다. 2. (S) Ring 객체는 onOff 값에 따라 ringing()을 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use case	**20. Change world**
Actor	User
Type	Evident
Pre - requisites	1. currentMode의 값이 WLT이어야 한다.
Typical courses of events	1.1 (A) : 사용자가 B 버튼을 누른다. (S) : display(worldName[i+1])를 호출한다. (S) : display(worldDate[i+1])을 호출한다. (S) : display(worldTime[i+1])을 호출한다. 1.2 (A) : User가 D 버튼을 누른다. (S) : display(worldName[i-1])를 호출한다. (S) : display(worldDate[i-1])을 호출한다. (S) : display(worldTime[i-1])을 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

Use case	**21. Turn on brightness control**
Actor	User
Type	Evident
Pre - requisites	1. currentMode의 값이 BRC이어야한다.
Typical courses of events	1. (A) User 가 D 버튼을 누른다. 2. (S) checkBrightness() 를 호출하여 brightness 를 확인한다. 3. (S) switchBrightness() 를 호출하여 brightness 에 true 를 저장한다. 4. (S) display(brightness) 를 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1. Brightness control (BC) 기능이 켜진상태에서는 동작하지 않는다.

Use case	**22. Turn off brightness control**
Actor	User
Type	Evident
Pre - requisites	1. currentMode의 값이 BRC이어야한다.
Typical courses of events	1. (A) User 가 D 버튼을 누른다 . 2. (S) checkBrightness()를 호출하여 brightness 를 확인한다. 3. (S) switchBrightness() 를 호출하여 brightness 에 false 를 저장한다. 4. (S) initializeBrightness(0) 를 호출하여 brightness level을 ‘0’ 으로 조정한다. 5. (S) display(brightness) 를 호출한다.
Alternative courses of events	N/A
Exceptional courses of events	E1.Brightness control (BC) 기능이 켜진 상태에서만 동작한다.

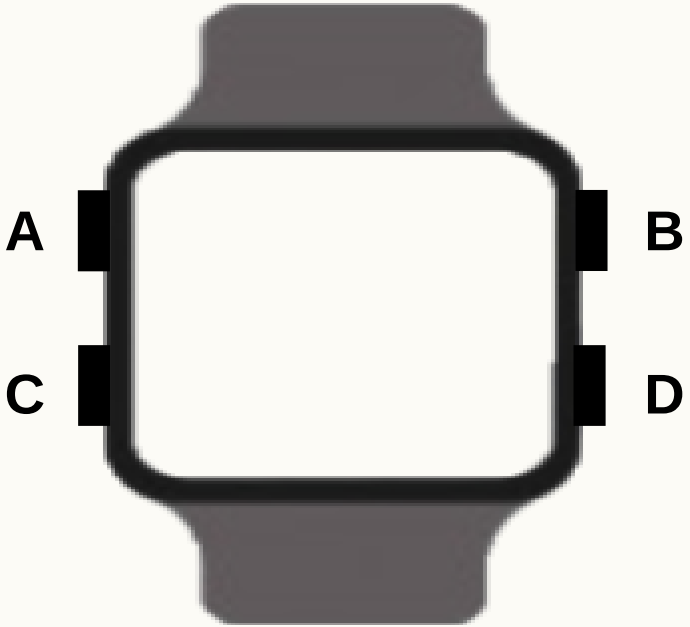
Use case	**23. Control brightness**
Actor	User
Type	Evident
Pre - requisites	1. currentMode의 값이 BRC이어야한다. 2. brightness 변수에 저장된 값이 true이어야한다.
Typical courses of events	1. (A) User 가 B 버튼을 누른다 . 2. (S) 밝기 level이 순환한다. 3. (A) User가 C 버튼을 누른다. 4. (S) controlBrightness() 를 호출하여 밝기 level 을 저장한다.
Alternative courses of events	밝기 level은 3단계가 버튼으로 순환한다.
Exceptional courses of events	N/A

Use case	**24. Change brightness**
Actor	User
Type	Hidden
Pre - requisites	1. brightness 변수에 저장된 값이 true여야한다.
Typical courses of events	1. (S) currentTime을 매 정각에 확인한다. 2. (S) changeBrightness()를 호출한다. 3. (S) DWS controller에 brightness를 전달한다.
Alternative courses of events	N/A
Exceptional courses of events	N/A

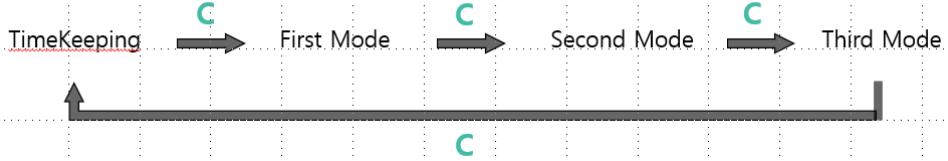
Use case	**25. Change SW**
Actor	User
Type	Evident
Pre - requisites	current Mode가 TimeKeeping이어야한다.
Typical courses of events	1. (A) User가 D 버튼을 누른다. 2. (S) enterChangeSW()을 호출한다. 3. (A) User가 B 버튼을 누른다. (S) changeSW(swMode[i])를 호출한다.(timekeeping 제외) (S) display(swMode[i])를 호출한다. 1. (A) User가 A 버튼을 누른다. (S) selectSW(swMode[i]) 를 통해 SW를 선택하고 저장한다. 1. (A) User가 D 버튼을 누른다. (S) TimeKeeping 모드로 돌아간다.
Alternative courses of events	N/A
Exceptional courses of events	E1. Timekeeping은 Default 기능으로 ChangeSW시 제외할 수 없다.

Activity 2042 :
Design UI

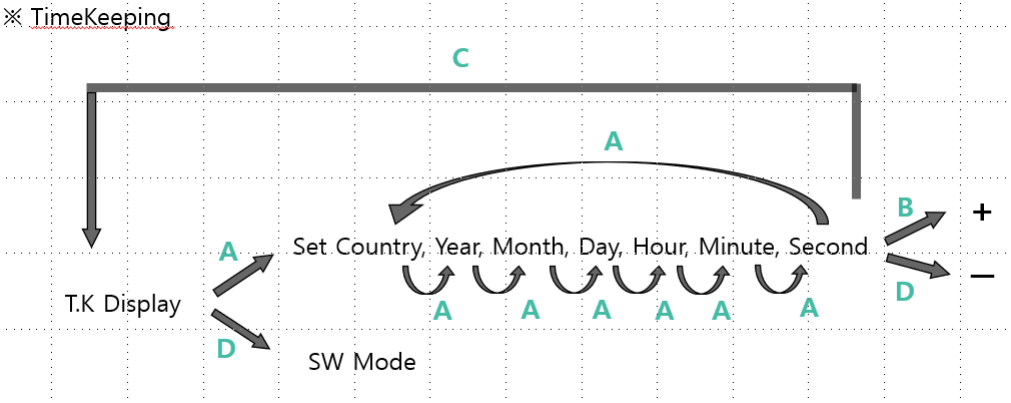
※ Button Map

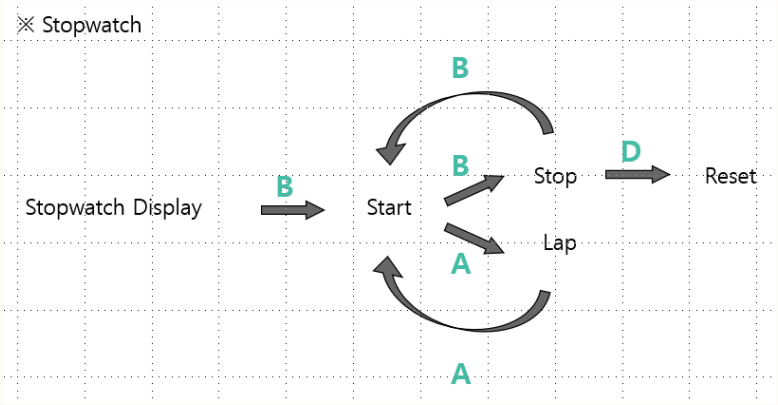


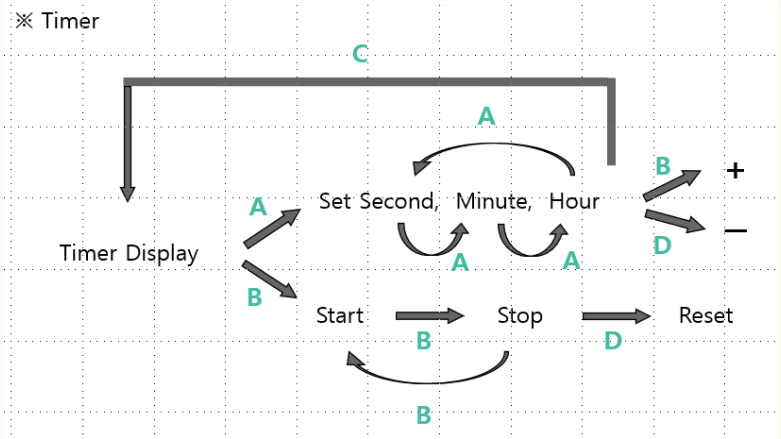
※ Mode

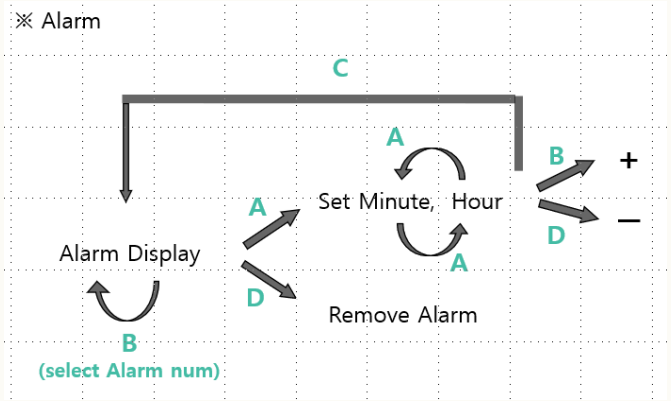


※ TimeKeeping

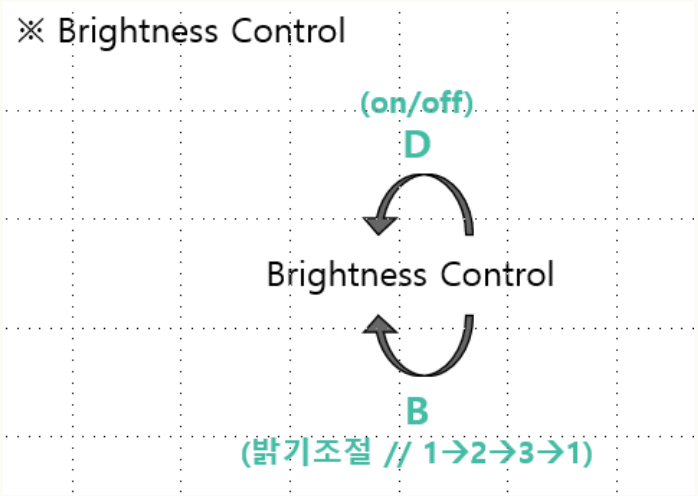



※ Stopwatch
B
B
Stopwatch Display
B
Start
B
Stop
D
Reset
A
Lap
A

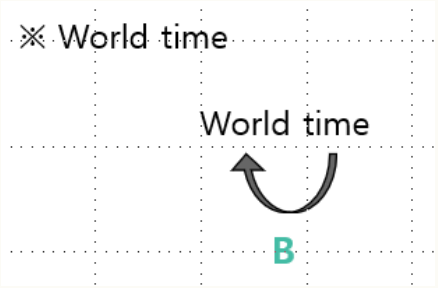

※ Timer
C
A
B
+
Set Second, Minute, Hour
A
A
D
–
Timer Display
A
B
Start
B
Stop
D
Reset
B


※ Alarm
C
A
B
+
A
Set Minute, Hour
D
–
Alarm Display
A
D
Remove Alarm
B
(select Alarm num)

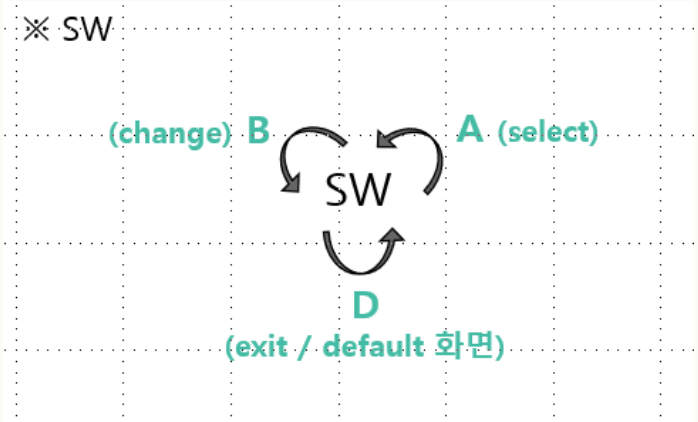
※ Brightness Control



※ World time



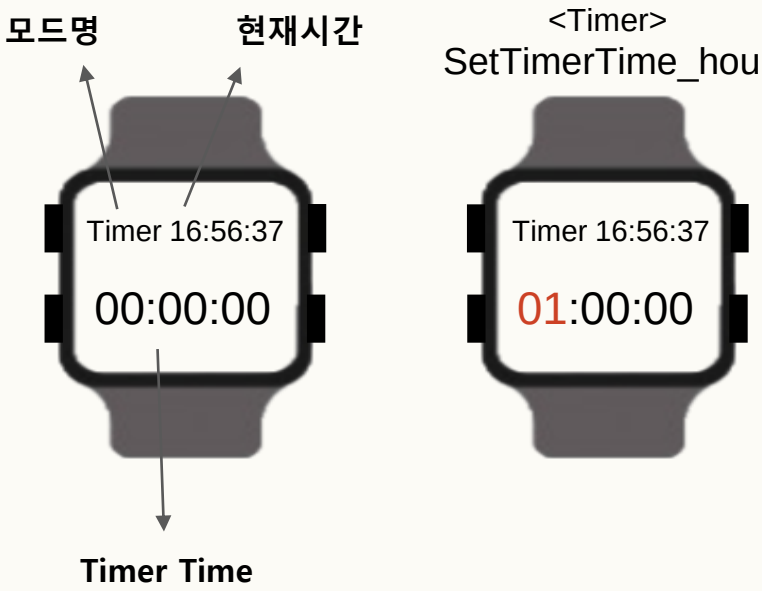
※ SW



Timekeeping

Stopwatch

Timer


모드명
현재시간
Timer 16:56:37
00:00:00
Timer Time

<Timer>
SetTimerTime_hour
Timer 16:56:37
01:00:00

Alarm

모드명
현재시간
Alarm 16:56:37
1. 12 : 30
알람 번호
Alarm Time

<Alarm>
selectAlarmNum
Alarm 16:56:37
2. - - : - -

Brightness Control

<Brightness Control>
MODE ON : LV.1

밝기 레벨

BC
ON/OFF

<Brightness Control>
MODE ON : LV.2

<Brightness Control>
MODE ON : LV.3

<Brightness Control>
MODE ON:
특정 시간 이후

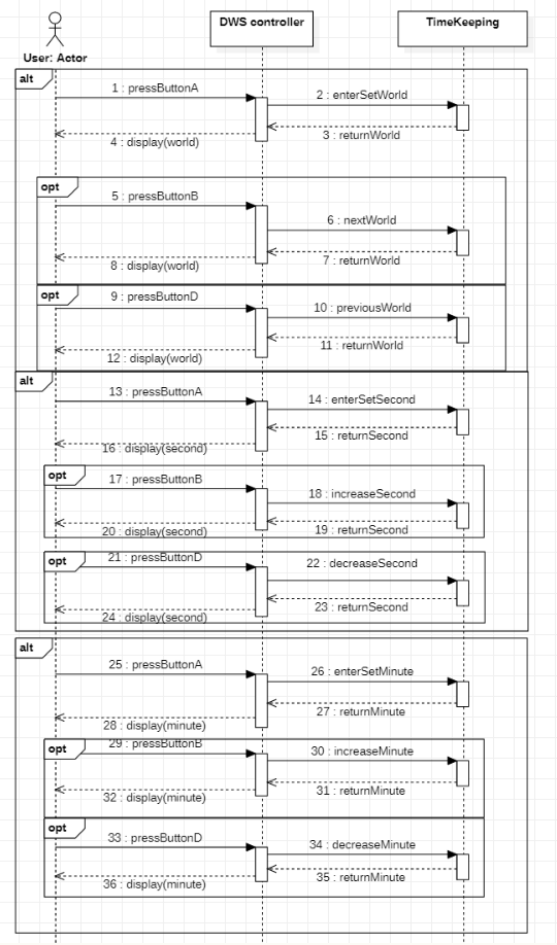
<Brightness Control>
MODE OFF:
LV.0

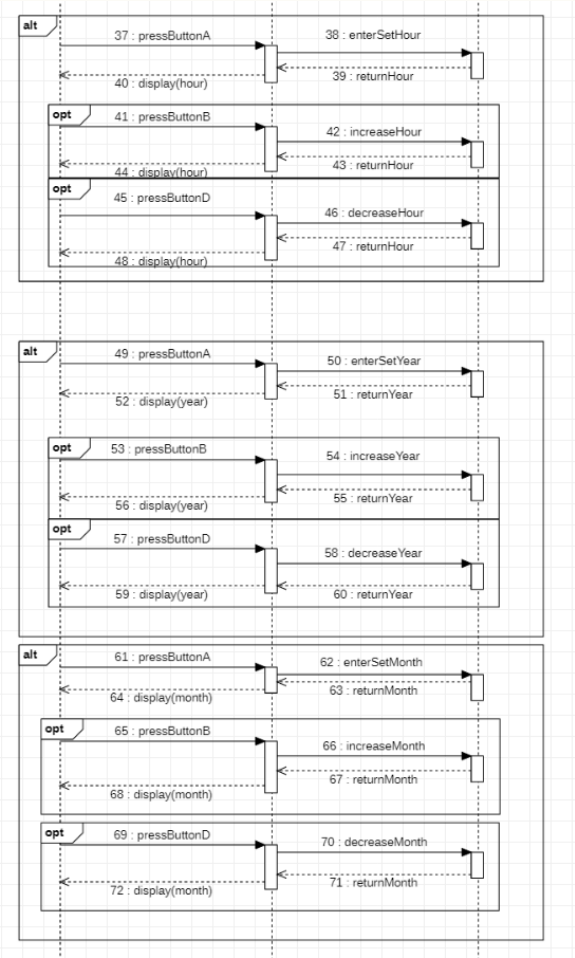
<WorldTime>
Russia

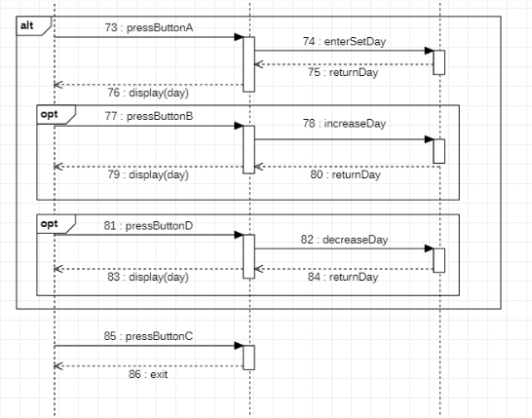
SW

Activity 2043 :
Define Interaction Diagrams

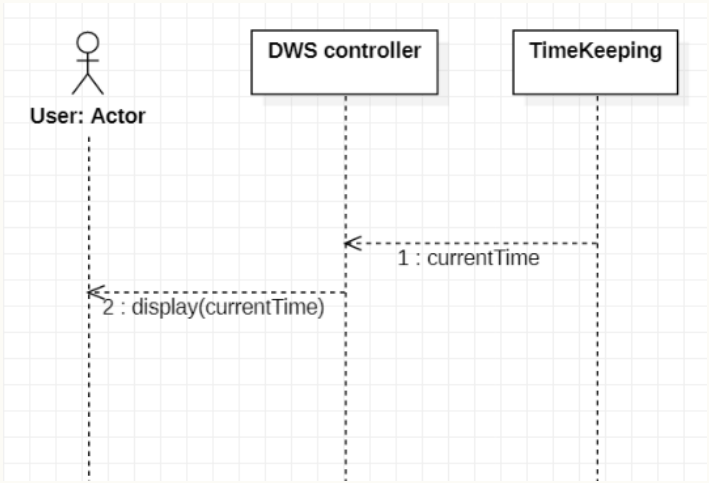
1.


User: Actor
DWS controller
TimeKeeping
alt
1 : pressButtonA
2 : enterSetWorld
3 : returnWorld
4 : display(world)
opt
5 : pressButtonB
6 : nextWorld
7 : returnWorld
8 : display(world)
opt
9 : pressButtonD
10 : previousWorld
11 : returnWorld
12 : display(world)
alt
13 : pressButtonA
14 : enterSetSecond
15 : returnSecond
16 : display(second)
opt
17 : pressButtonB
18 : increaseSecond
19 : returnSecond
20 : display(second)
opt
21 : pressButtonD
22 : decreaseSecond
23 : returnSecond
24 : display(second)
alt
25 : pressButtonA
26 : enterSetMinute
27 : returnMinute
28 : display(minute)
opt
29 : pressButtonB
30 : increaseMinute
31 : returnMinute
32 : display(minute)
opt
33 : pressButtonD
34 : decreaseMinute
35 : returnMinute
36 : display(minute)

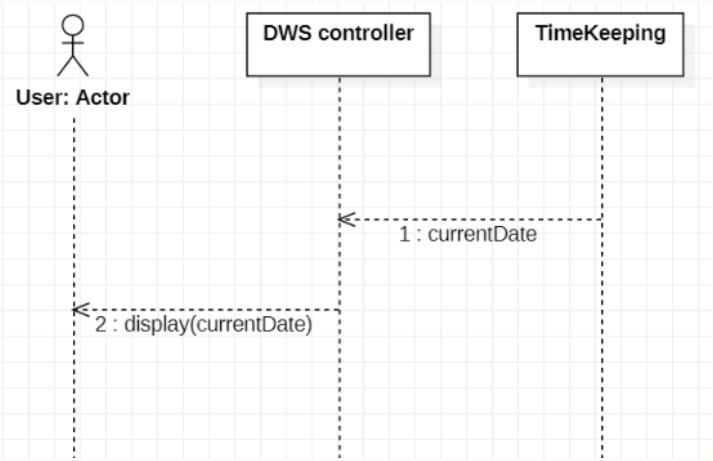

alt
37 : pressButtonA
38 : enterSetHour
40 : display(hour)
39 : returnHour
opt
41 : pressButtonB
42 : increaseHour
44 : display(hour)
43 : returnHour
opt
45 : pressButtonD
46 : decreaseHour
48 : display(hour)
47 : returnHour
alt
49 : pressButtonA
50 : enterSetYear
52 : display(year)
51 : returnYear
opt
53 : pressButtonB
54 : increaseYear
56 : display(year)
55 : returnYear
opt
57 : pressButtonD
58 : decreaseYear
59 : display(year)
60 : returnYear
alt
61 : pressButtonA
62 : enterSetMonth
64 : display(month)
63 : returnMonth
opt
65 : pressButtonB
66 : increaseMonth
68 : display(month)
67 : returnMonth
opt
69 : pressButtonD
70 : decreaseMonth
72 : display(month)
71 : returnMonth


alt
73 : pressButtonA
74 : enterSetDay
76 : display(day)
75 : returnDay
opt
77 : pressButtonB
78 : increaseDay
79 : display(day)
80 : returnDay
opt
81 : pressButtonD
82 : decreaseDay
83 : display(day)
84 : returnDay
85 : pressButtonC
86 : exit

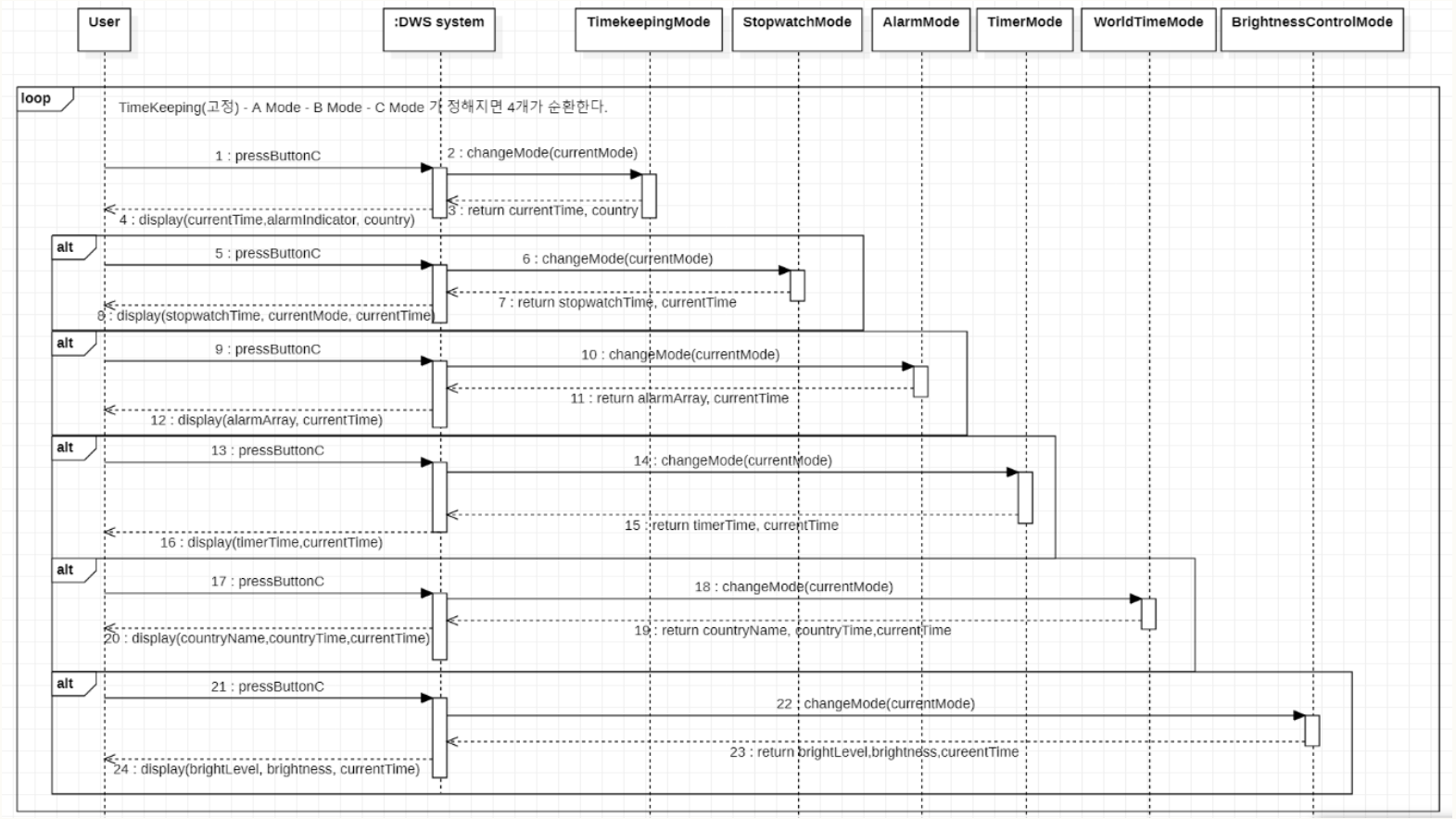
2.


User: Actor
DWS controller
TimeKeeping
1 : currentTime
2 : display(currentTime)

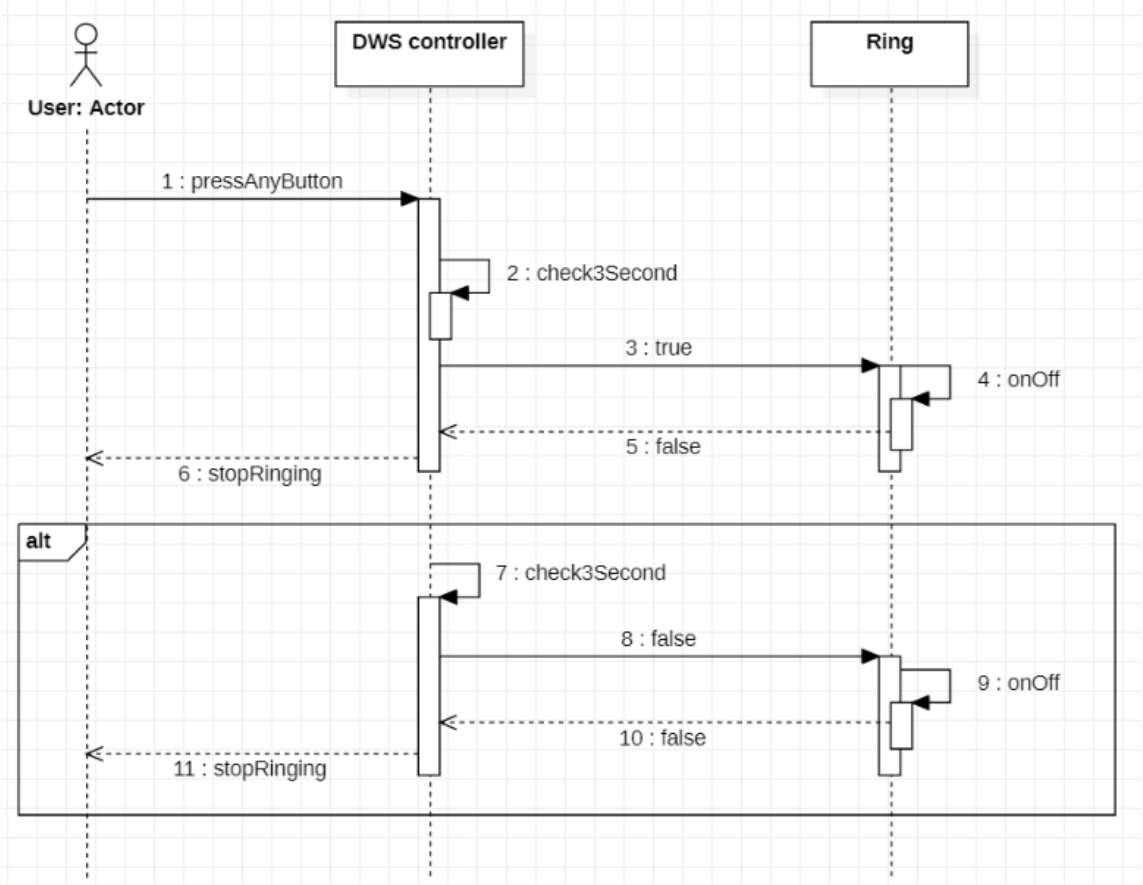
3.


User: Actor
DWS controller
TimeKeeping
1 : currentDate
2 : display(currentDate)

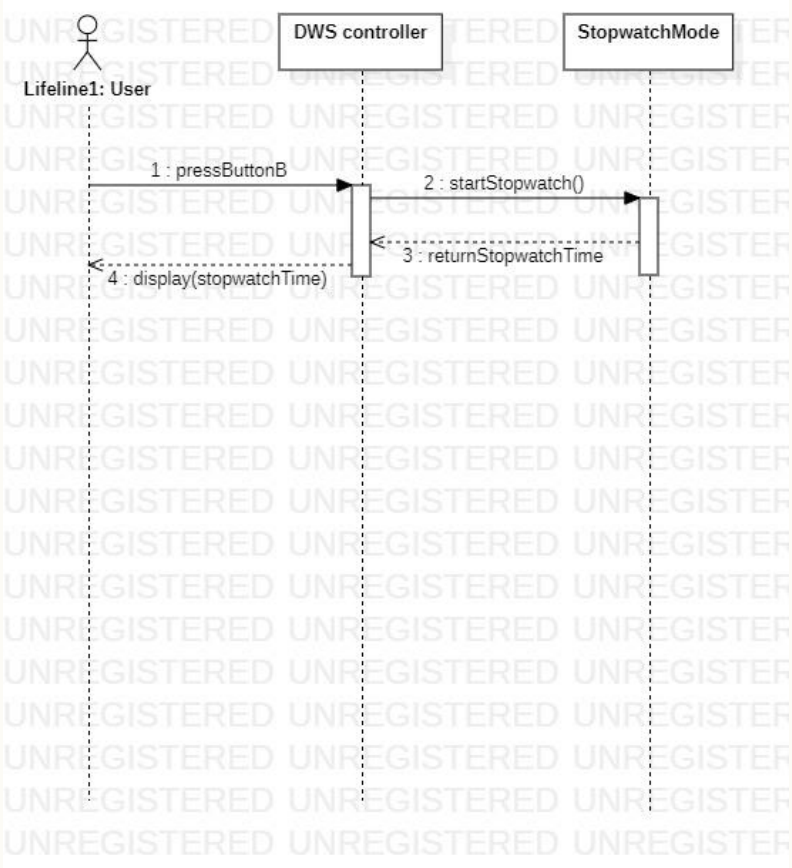
4.

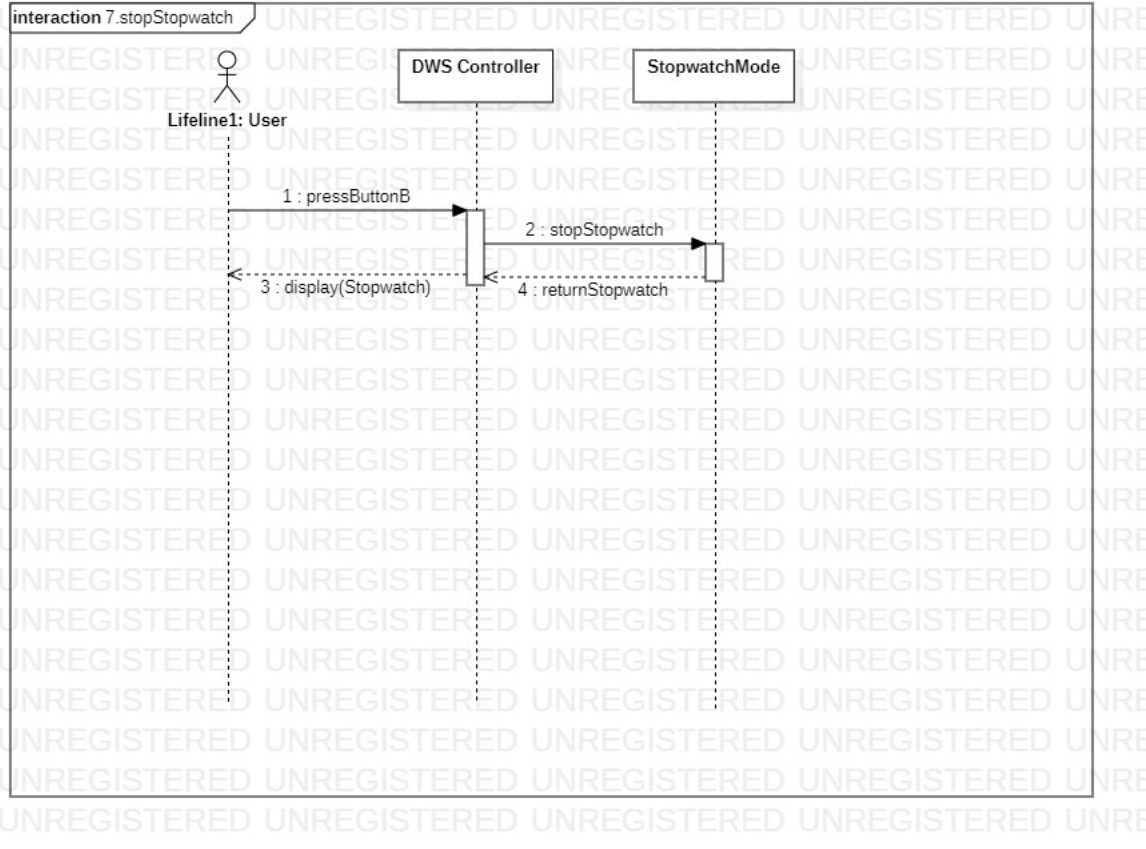

User
:DWS system
TimekeepingMode
StopwatchMode
AlarmMode
TimerMode
WorldTimeMode
BrightnessControlMode
loop
TimeKeeping(고정) - A Mode - B Mode - C Mode 가 정해지면 4개가 순환한다.
1 : pressButtonC
2 : changeMode(currentMode)
3 : return currentTime, country
4 : display(currentTime,alarmIndicator, country)
alt
5 : pressButtonC
6 : changeMode(currentMode)
7 : return stopwatchTime, currentTime
8 : display(stopwatchTime, currentMode, currentTime)
alt
9 : pressButtonC
10 : changeMode(currentMode)
11 : return alarmArray, currentTime
12 : display(alarmArray, currentTime)
alt
13 : pressButtonC
14 : changeMode(currentMode)
15 : return timerTime, currentTime
16 : display(timerTime,currentTime)
alt
17 : pressButtonC
18 : changeMode(currentMode)
19 : return countryName, countryTime,currentTime
20 : display(countryName,countryTime,currentTime)
alt
21 : pressButtonC
22 : changeMode(currentMode)
23 : return brightLevel,brightness,cureentTime
24 : display(brightLevel, brightness, currentTime)

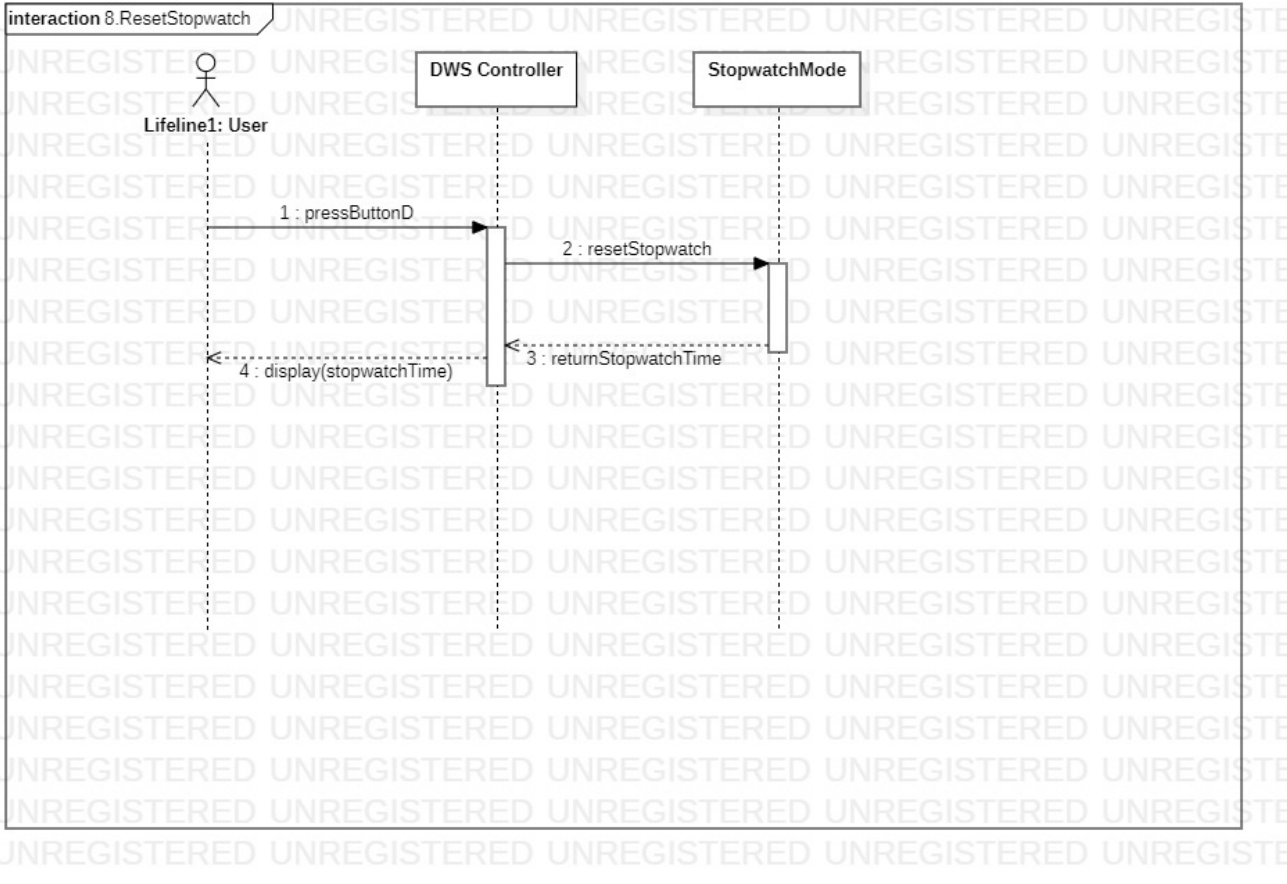
5.

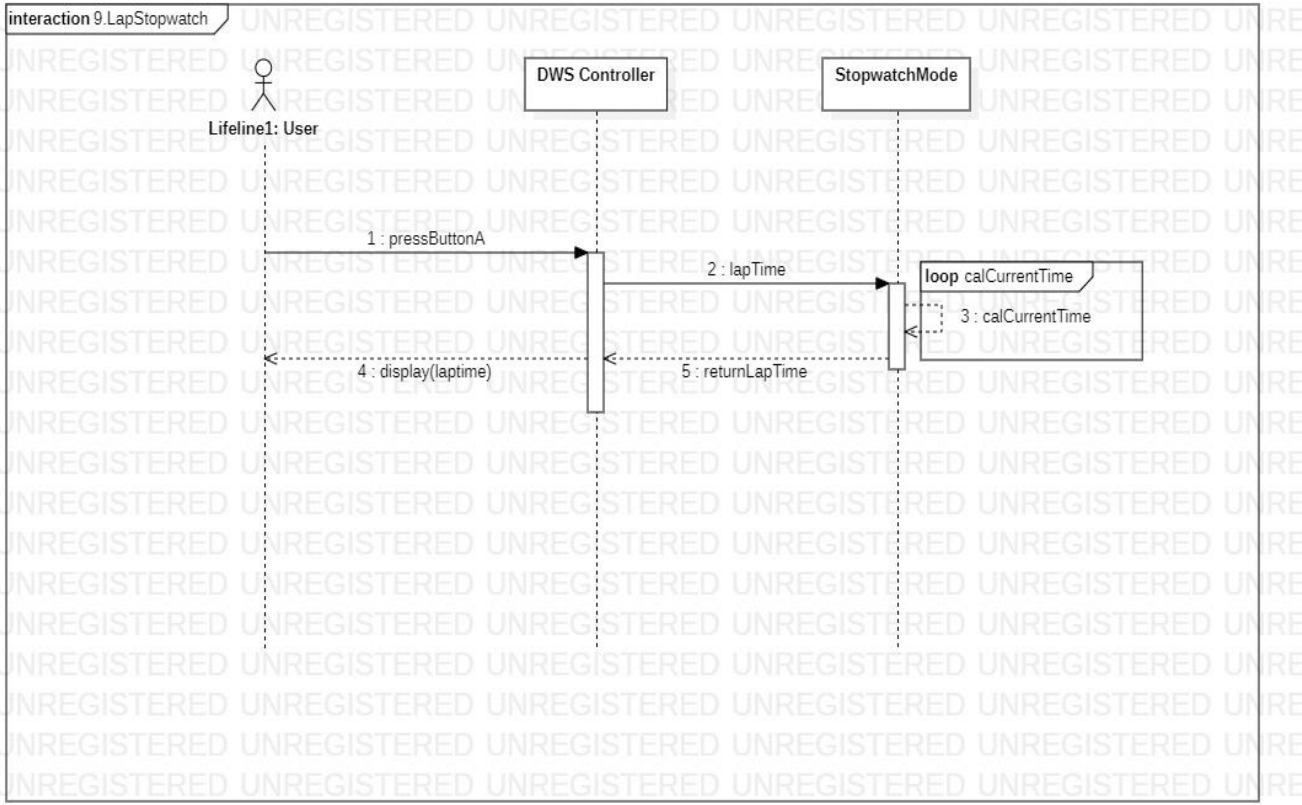

User: Actor
DWS controller
Ring
1 : pressAnyButton
2 : check3Second
3 : true
4 : onOff
5 : false
6 : stopRinging
alt
7 : check3Second
8 : false
9 : onOff
10 : false
11 : stopRinging

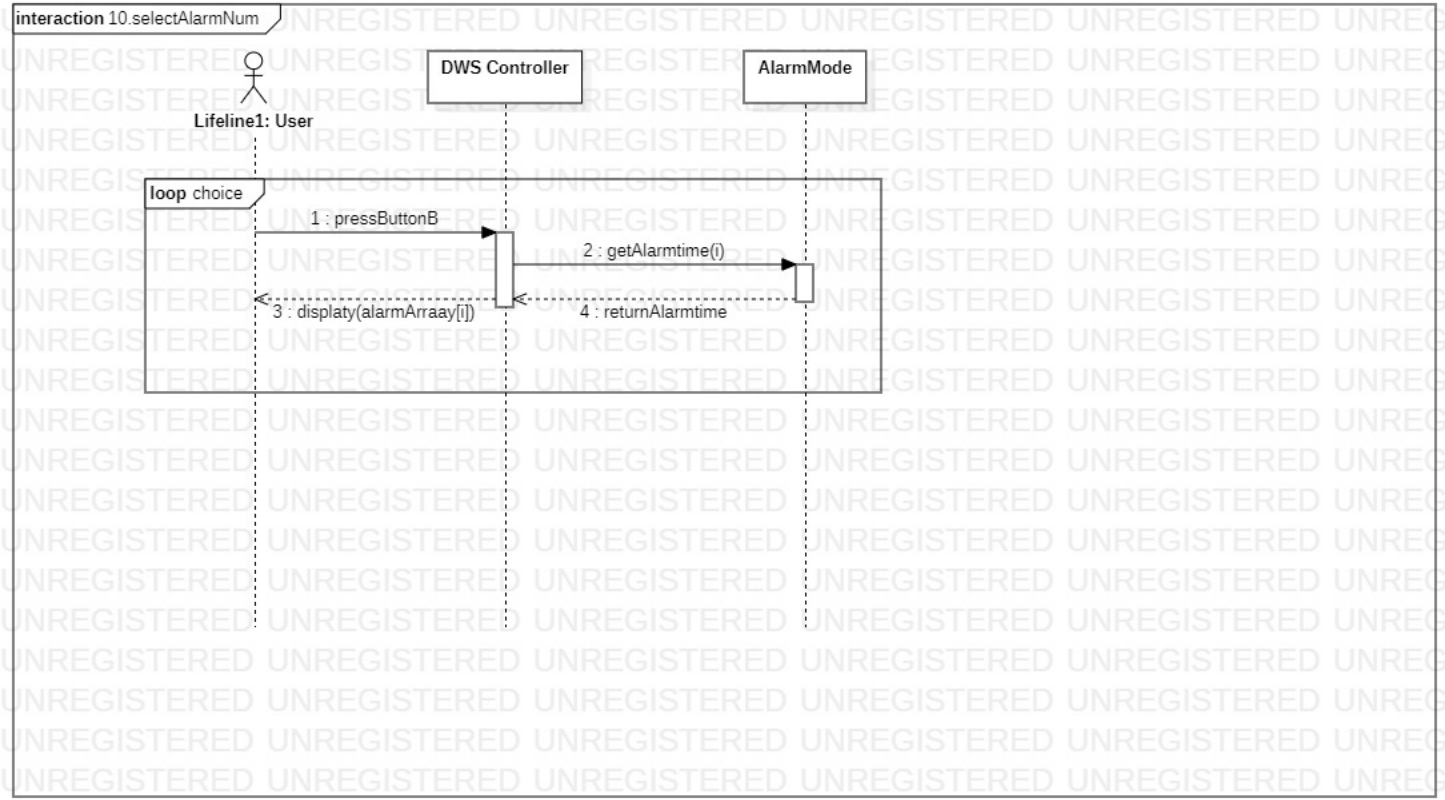
6.

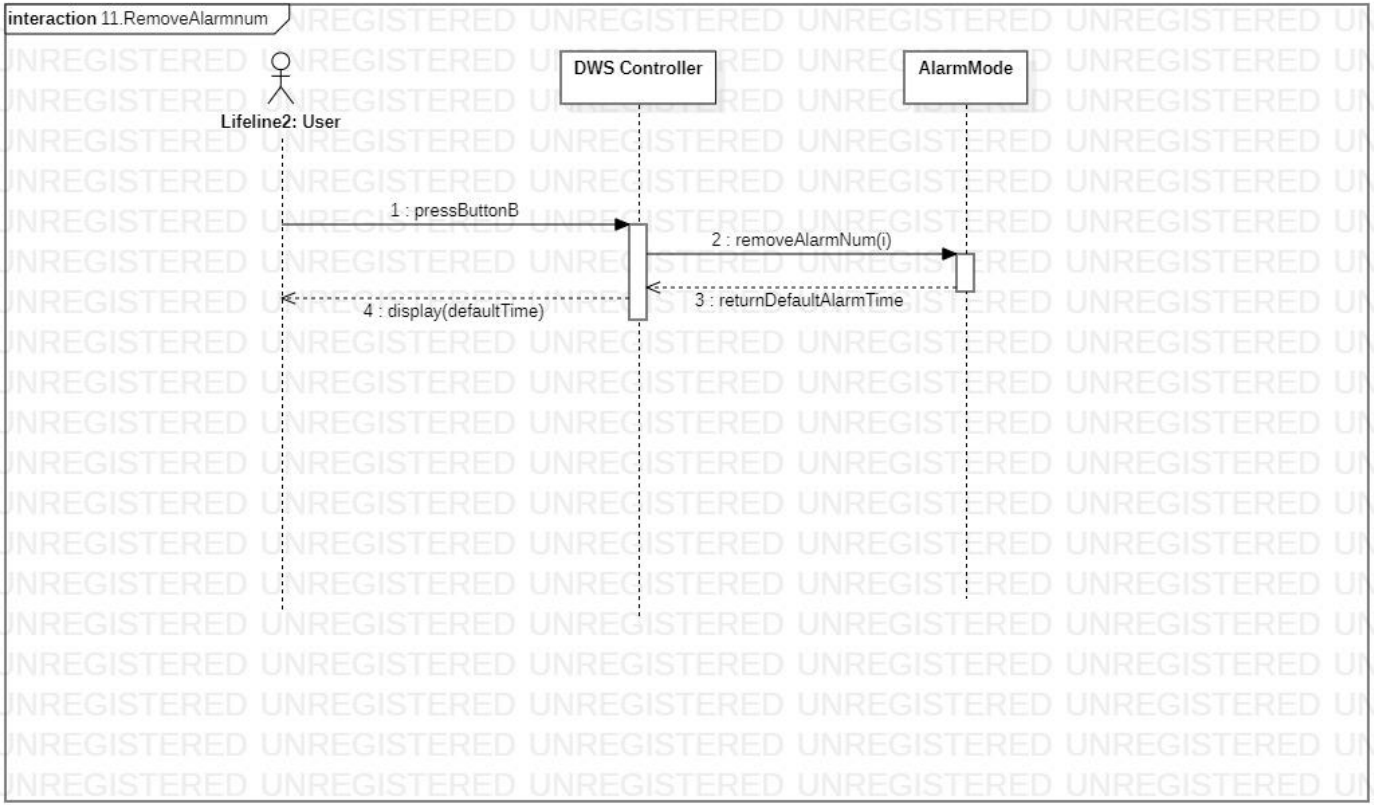

DWS controller
StopwatchMode
Lifeline1: User
1 : pressButtonB
2 : startStopwatch()
3 : returnStopwatchTime
4 : display(stopwatchTime)

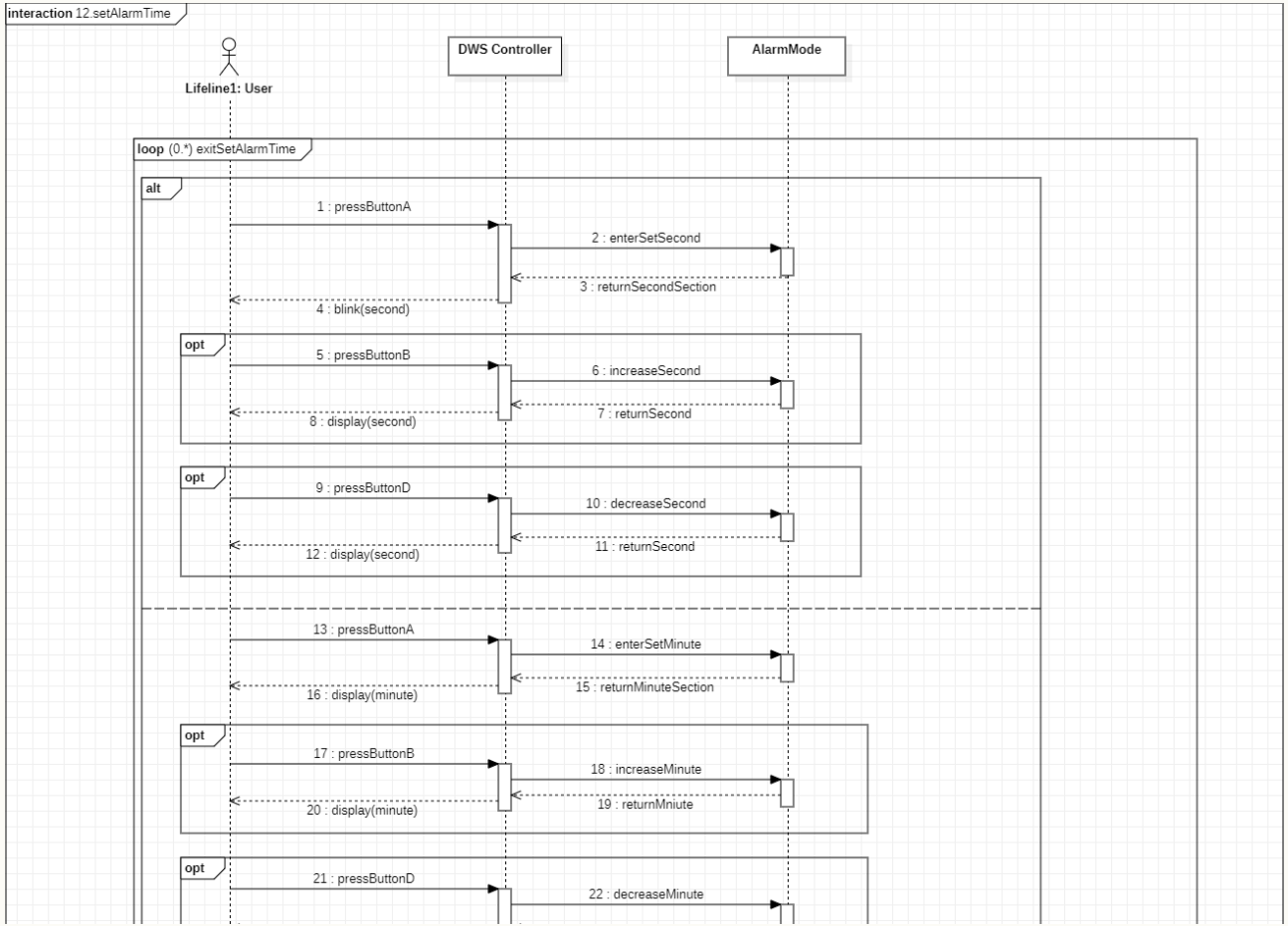

interaction 7.stopStopwatch
DWS Controller
StopwatchMode
Lifeline1: User
1 : pressButtonB
2 : stopStopwatch
3 : display(Stopwatch)
4 : returnStopwatch

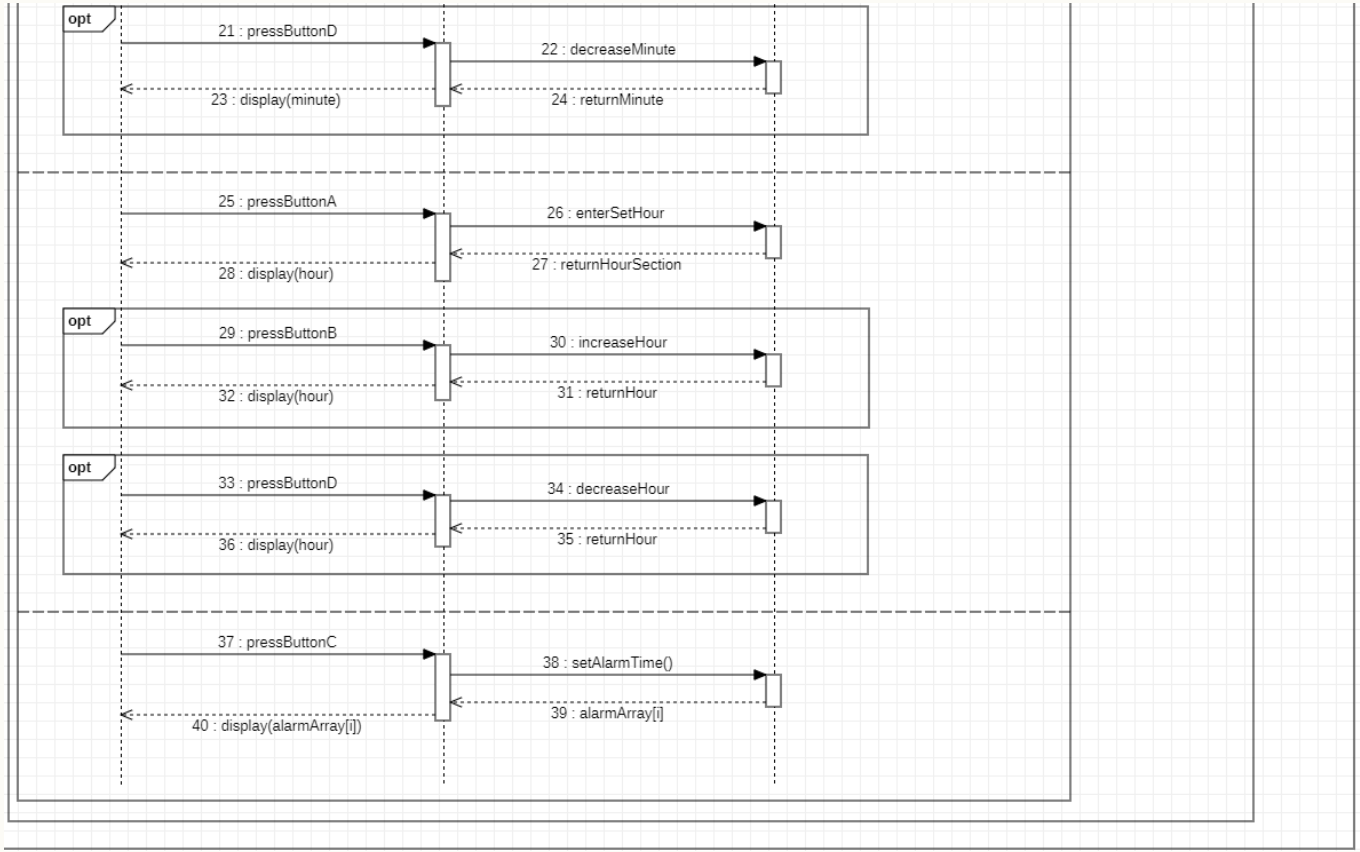

interaction 8.ResetStopwatch
DWS Controller
StopwatchMode
Lifeline1: User
1 : pressButtonD
2 : resetStopwatch
3 : returnStopwatchTime
4 : display(stopwatchTime)

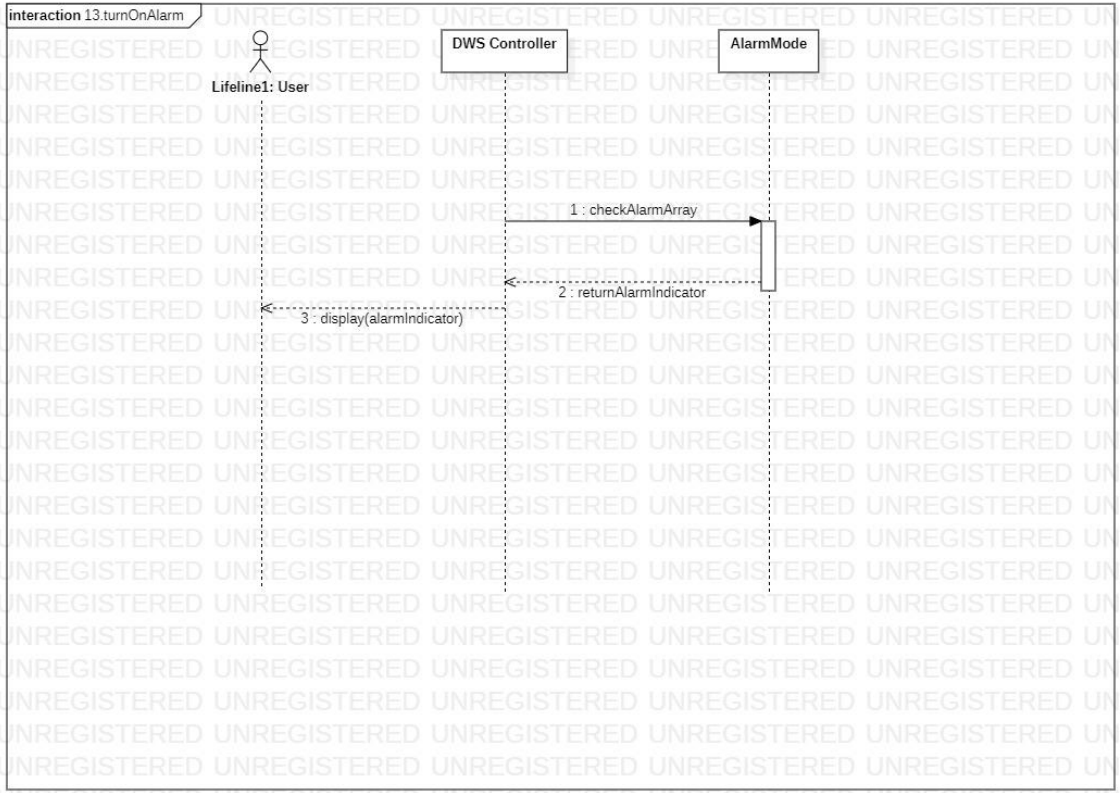

interaction 9.LapStopwatch
Lifeline1: User
DWS Controller
StopwatchMode
1 : pressButtonA
2 : lapTime
loop calCurrentTime
3 : calCurrentTime
4 : display(laptime)
5 : returnLapTime

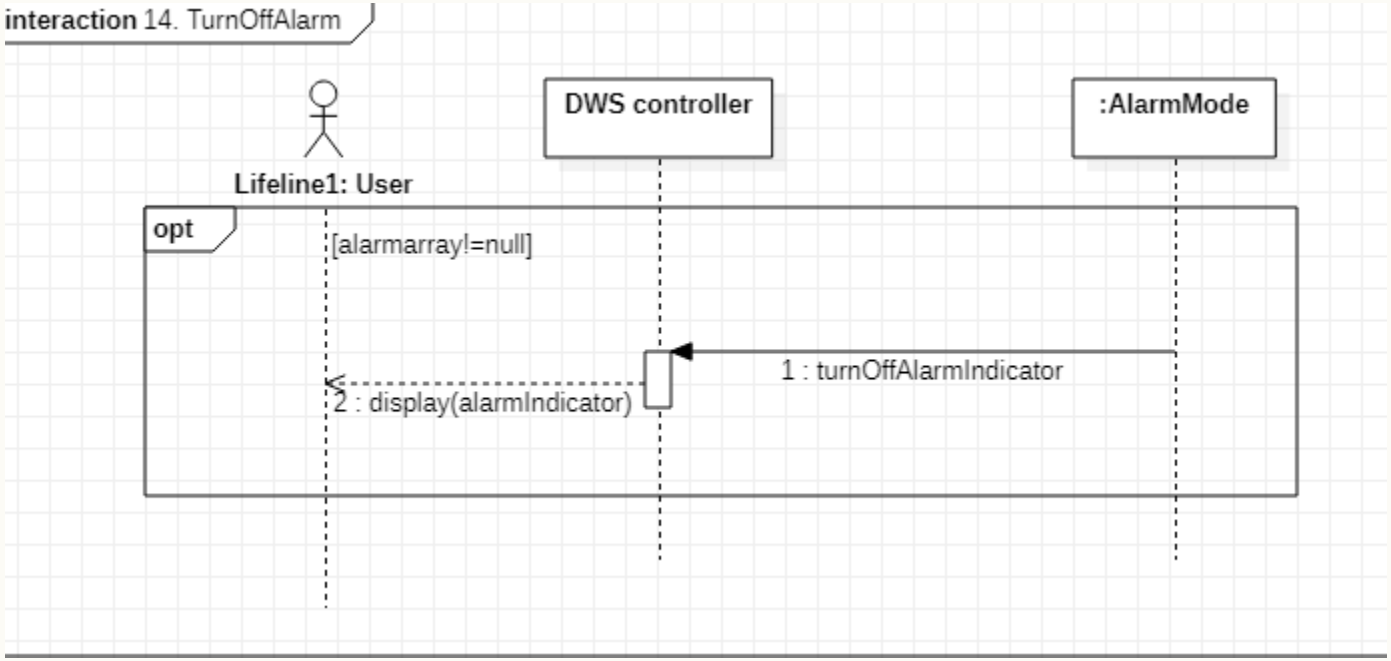

interaction 10.selectAlarmNum
DWS Controller
AlarmMode
Lifeline1: User
loop choice
1 : pressButtonB
2 : getAlarmtime(i)
3 : displaty(alarmArraay[i])
4 : returnAlarmtime

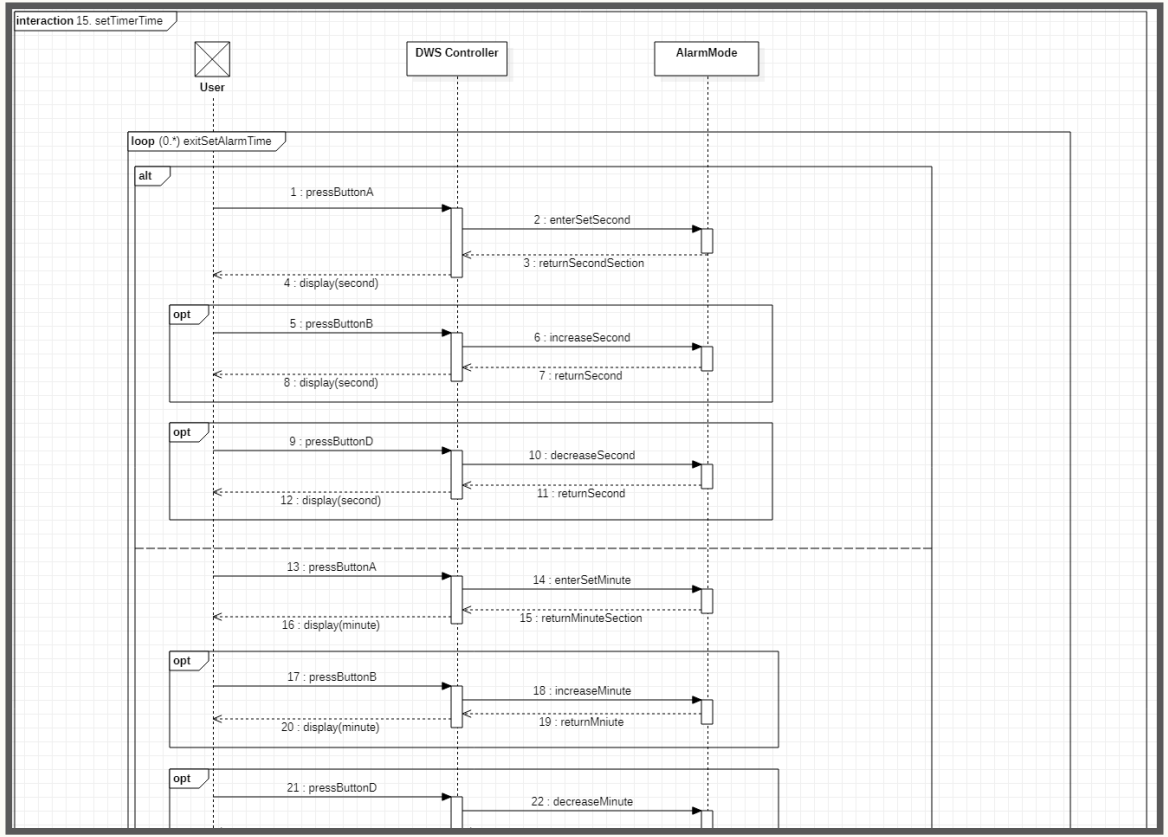

interaction 11.RemoveAlarmnum
Lifeline2: User
DWS Controller
AlarmMode
1 : pressButtonB
2 : removeAlarmNum(i)
3 : returnDefaultAlarmTime
4 : display(defaultTime)

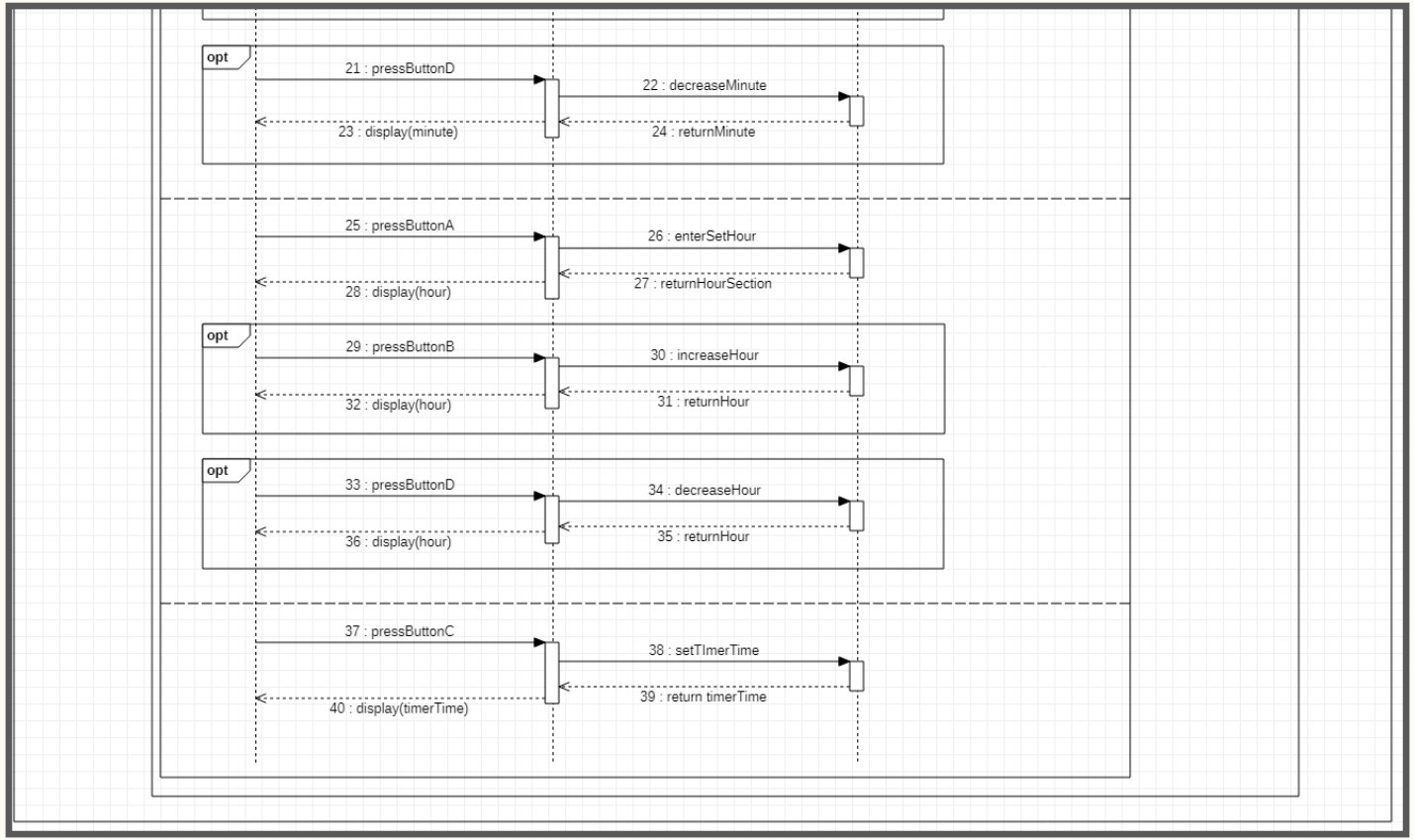

interaction 12.setAlarmTime
Lifeline1: User
DWS Controller
AlarmMode
loop (0.*) exitSetAlarmTime
alt
1 : pressButtonA
2 : enterSetSecond
3 : returnSecondSection
4 : blink(second)
opt
5 : pressButtonB
6 : increaseSecond
7 : returnSecond
8 : display(second)
opt
9 : pressButtonD
10 : decreaseSecond
11 : returnSecond
12 : display(second)
13 : pressButtonA
14 : enterSetMinute
15 : returnMinuteSection
16 : display(minute)
opt
17 : pressButtonB
18 : increaseMinute
19 : returnMniute
20 : display(minute)
opt
21 : pressButtonD
22 : decreaseMinute

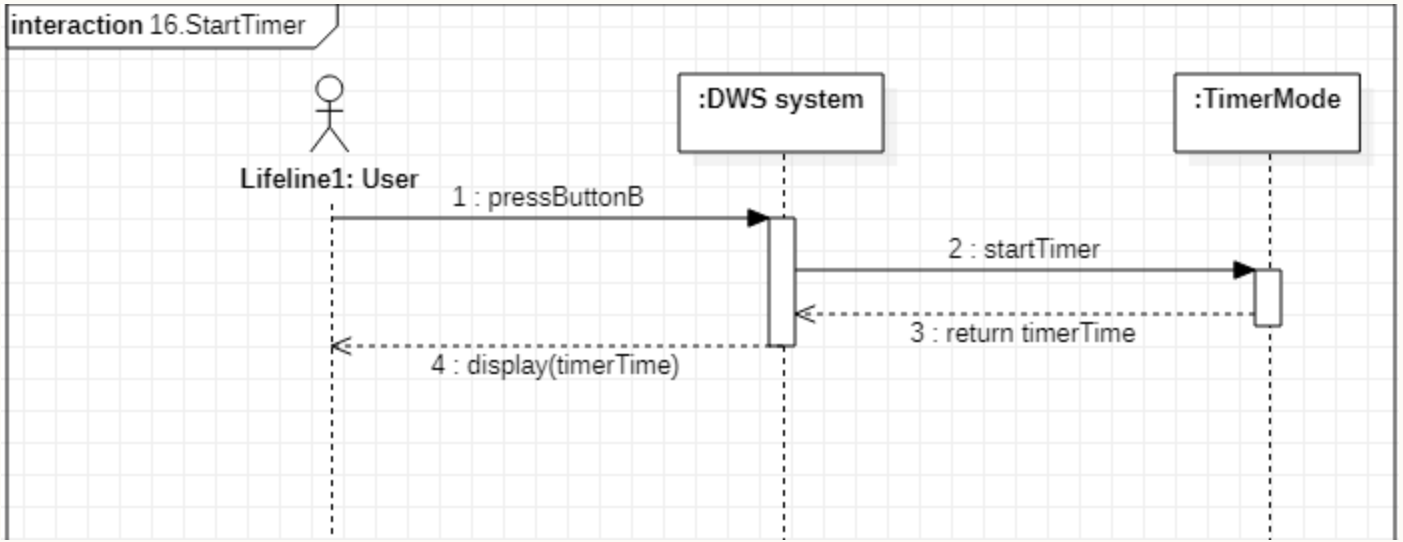

opt
21 : pressButtonD
22 : decreaseMinute
23 : display(minute)
24 : returnMinute
25 : pressButtonA
26 : enterSetHour
28 : display(hour)
27 : returnHourSection
opt
29 : pressButtonB
30 : increaseHour
32 : display(hour)
31 : returnHour
opt
33 : pressButtonD
34 : decreaseHour
36 : display(hour)
35 : returnHour
37 : pressButtonC
38 : setAlarmTime()
40 : display(alarmArray[i])
39 : alarmArray[i]

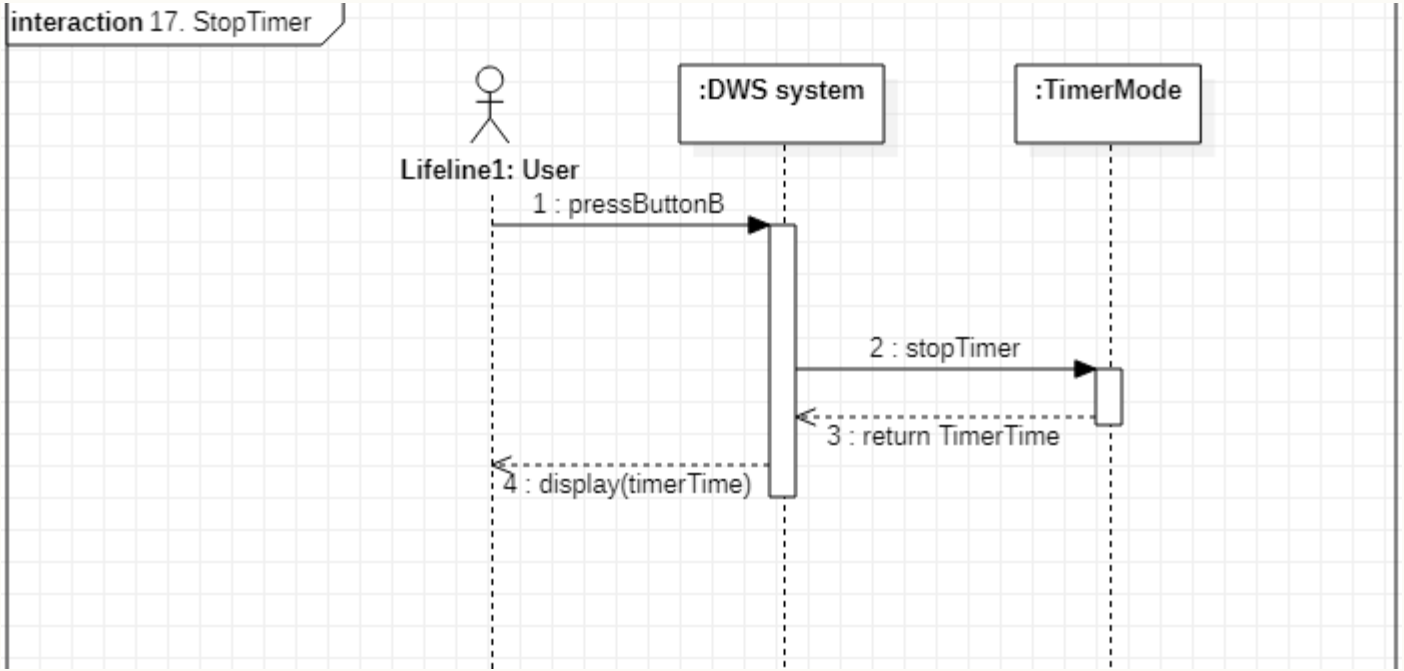

interaction 13.turnOnAlarm
Lifeline1: User
DWS Controller
AlarmMode
1 : checkAlarmArray
2 : returnAlarmIndicator
3 : display(alarmIndicator)

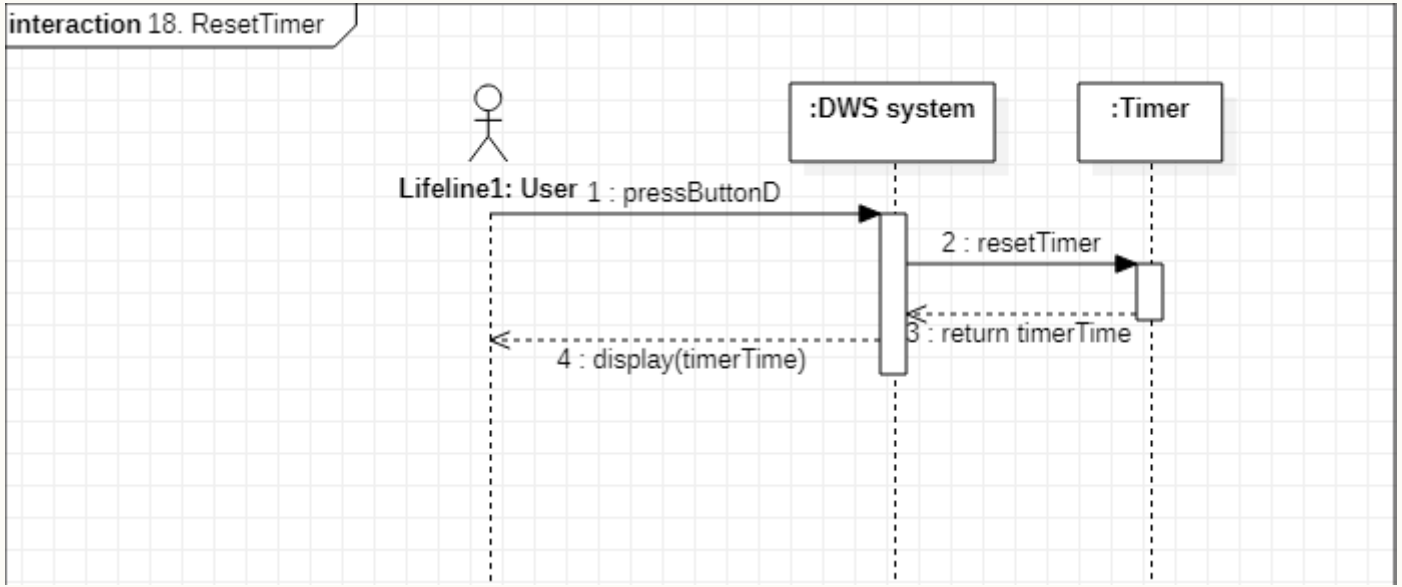

interaction 14. TurnOffAlarm
DWS controller
:AlarmMode
Lifeline1: User
opt
[alarmarray!=null]
1 : turnOffAlarmIndicator
2 : display(alarmIndicator)

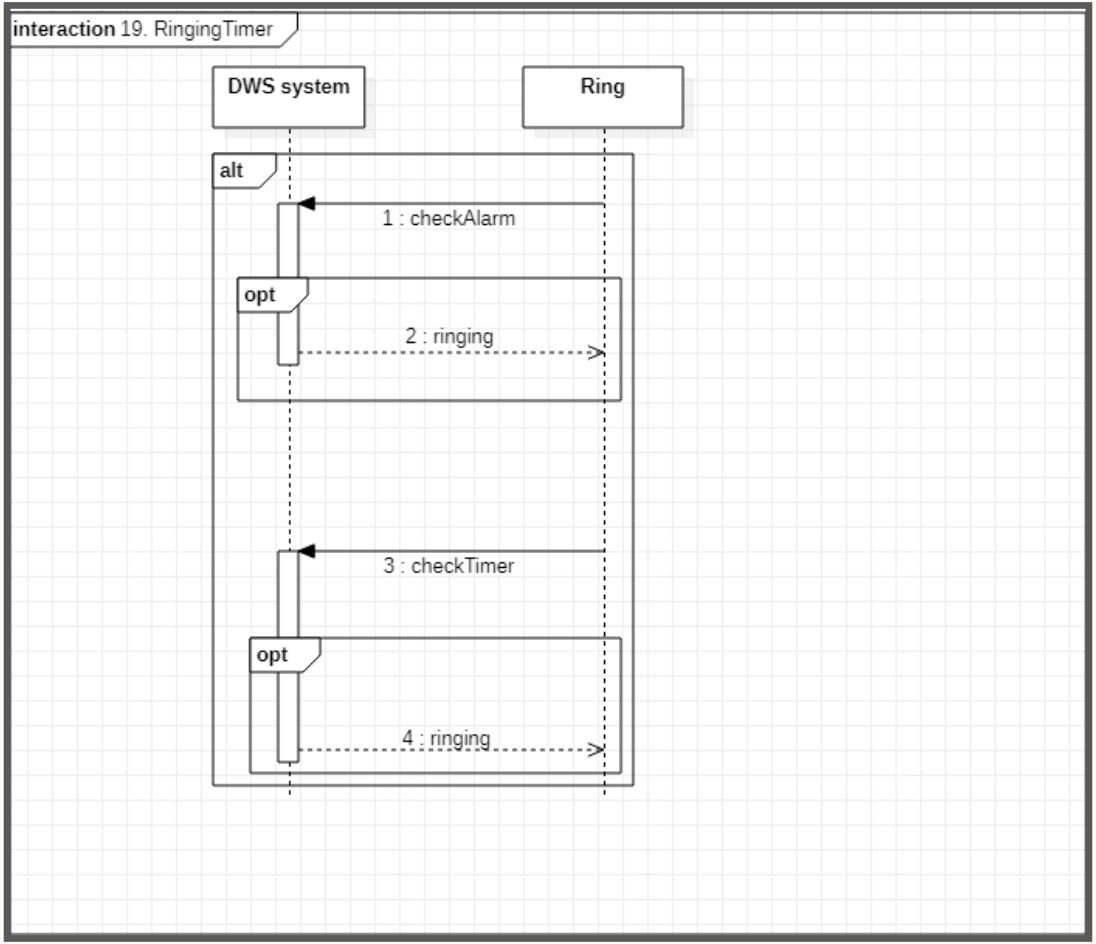

interaction 15. setTimerTime
User
DWS Controller
AlarmMode
loop (0.*) exitSetAlarmTime
alt
1 : pressButtonA
2 : enterSetSecond
3 : returnSecondSection
4 : display(second)
opt
5 : pressButtonB
6 : increaseSecond
7 : returnSecond
8 : display(second)
opt
9 : pressButtonD
10 : decreaseSecond
11 : returnSecond
12 : display(second)
13 : pressButtonA
14 : enterSetMinute
15 : returnMinuteSection
16 : display(minute)
opt
17 : pressButtonB
18 : increaseMinute
19 : returnMniute
20 : display(minute)
opt
21 : pressButtonD
22 : decreaseMinute

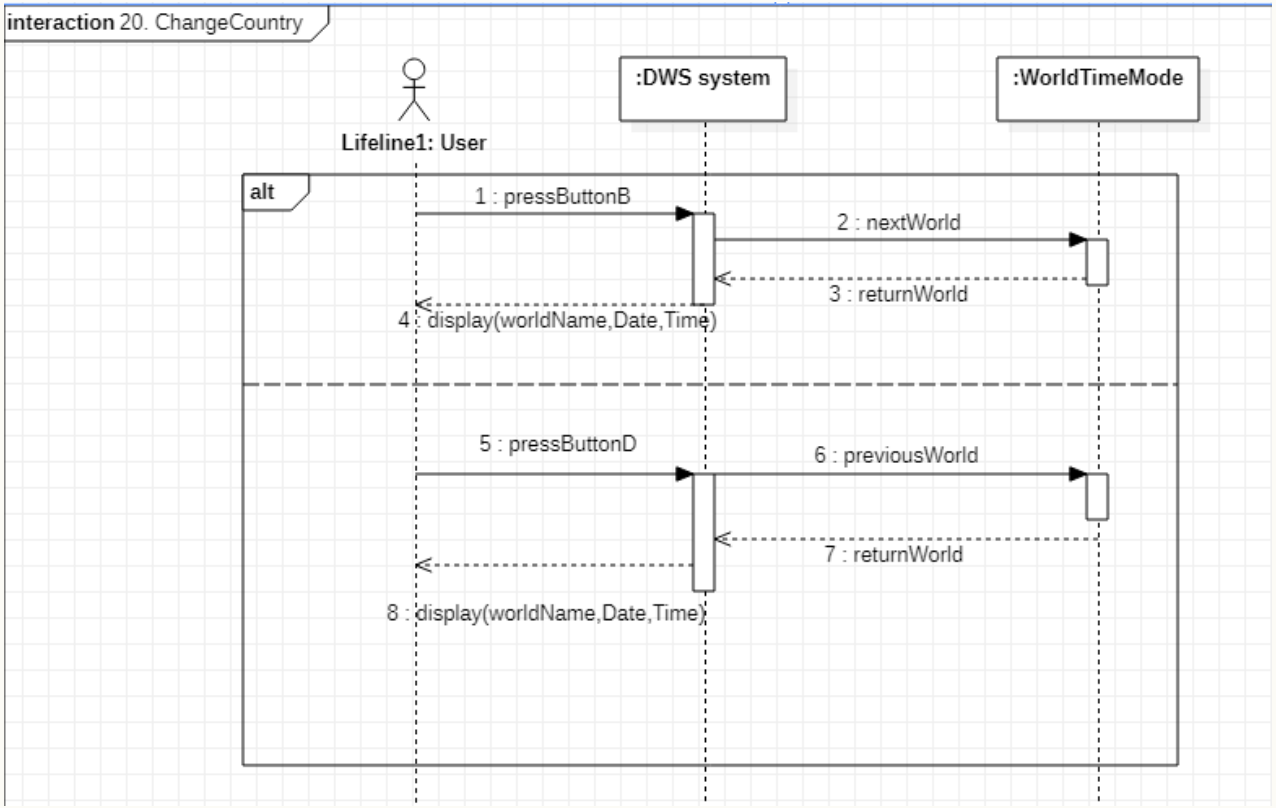

opt
21 : pressButtonD
22 : decreaseMinute
23 : display(minute)
24 : returnMinute
25 : pressButtonA
26 : enterSetHour
28 : display(hour)
27 : returnHourSection
opt
29 : pressButtonB
30 : increaseHour
32 : display(hour)
31 : returnHour
opt
33 : pressButtonD
34 : decreaseHour
36 : display(hour)
35 : returnHour
37 : pressButtonC
38 : setTImerTime
40 : display(timerTime)
39 : return timerTime


interaction 16.StartTimer
:DWS system
:TimerMode
Lifeline1: User
1 : pressButtonB
2 : startTimer
3 : return timerTime
4 : display(timerTime)

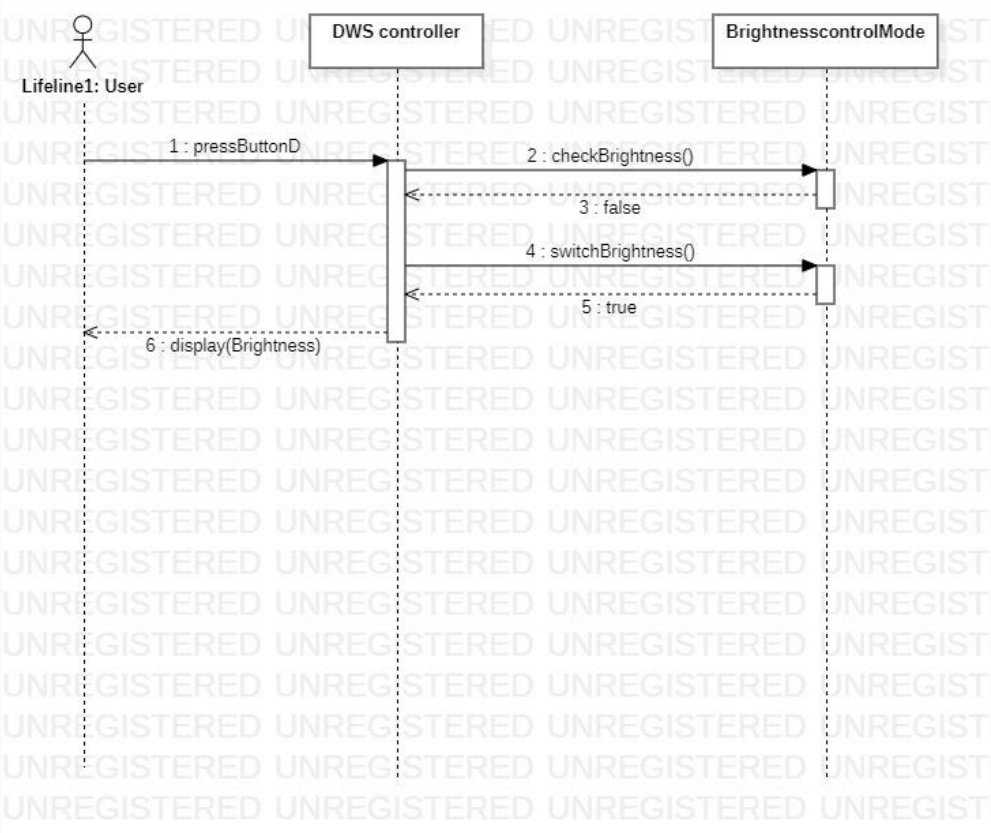

interaction 17. StopTimer
:DWS system
:TimerMode
Lifeline1: User
1 : pressButtonB
2 : stopTimer
3 : return TimerTime
4 : display(timerTime)


interaction 18. ResetTimer
:DWS system
:Timer
Lifeline1: User
1 : pressButtonD
2 : resetTimer
3 : return timerTime
4 : display(timerTime)

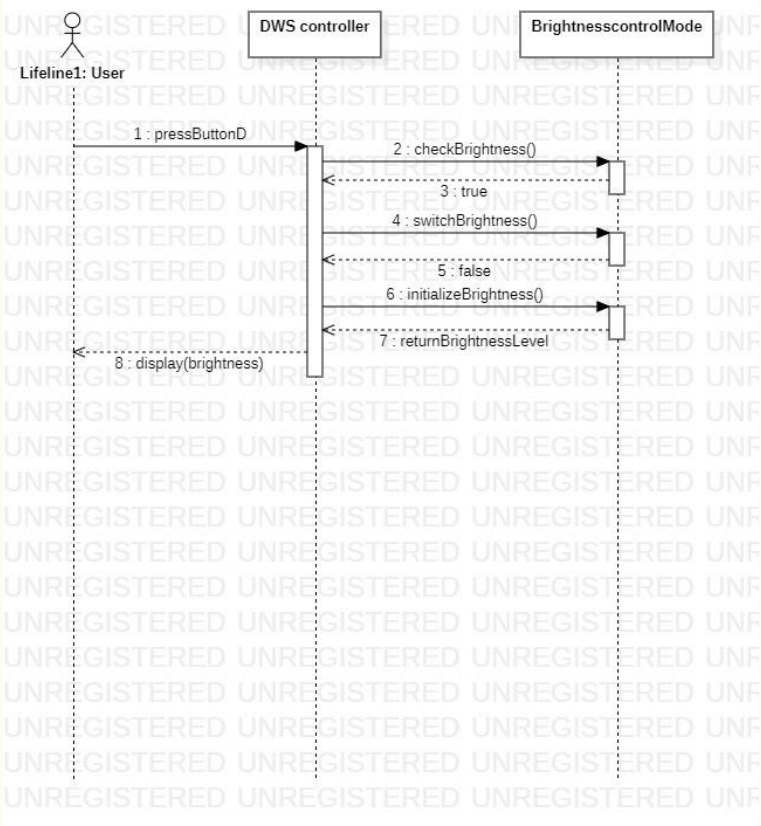

interaction 19. RingingTimer
DWS system
Ring
alt
1 : checkAlarm
opt
2 : ringing
3 : checkTimer
opt
4 : ringing


interaction 20. ChangeCountry
:DWS system
:WorldTimeMode
Lifeline1: User
alt
1 : pressButtonB
2 : nextWorld
3 : returnWorld
4 : display(worldName,Date,Time)
5 : pressButtonD
6 : previousWorld
7 : returnWorld
8 : display(worldName,Date,Time)

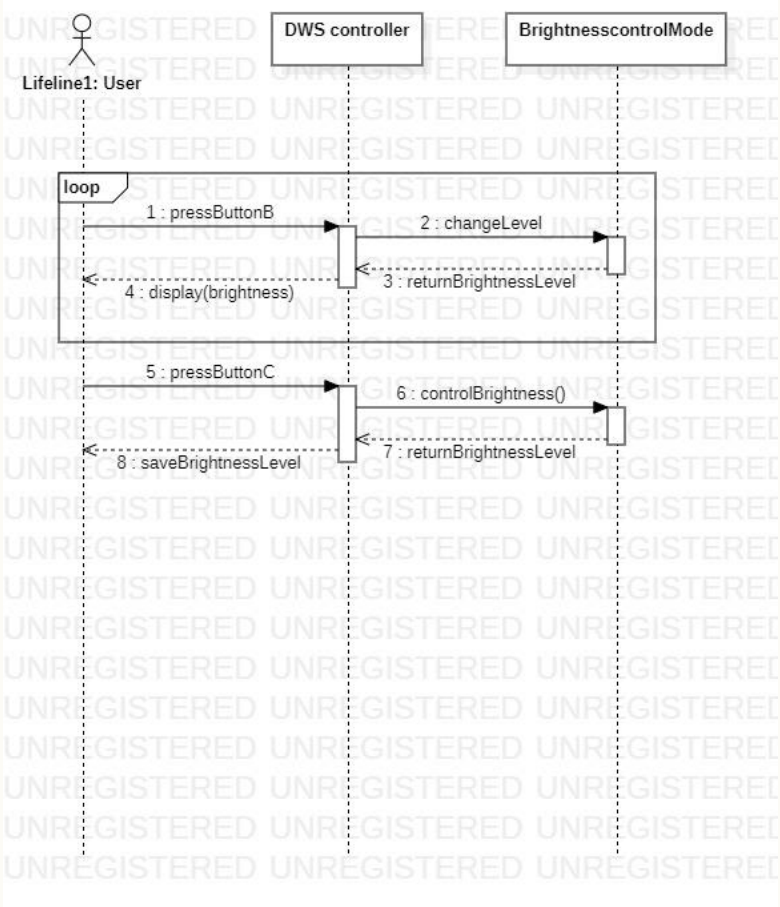
21.


DWS controller
BrightnesscontrolMode
Lifeline1: User
1 : pressButtonD
2 : checkBrightness()
3 : false
4 : switchBrightness()
5 : true
6 : display(Brightness)

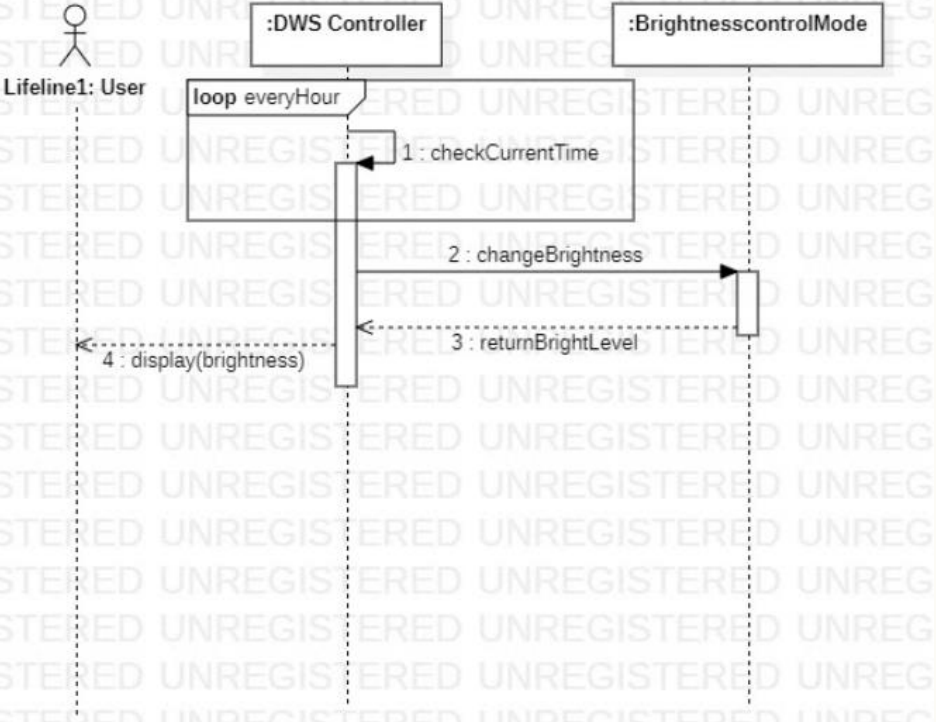
22.


DWS controller
BrightnesscontrolMode
Lifeline1: User
1 : pressButtonD
2 : checkBrightness()
3 : true
4 : switchBrightness()
5 : false
6 : initializeBrightness()
7 : returnBrightnessLevel
8 : display(brightness)

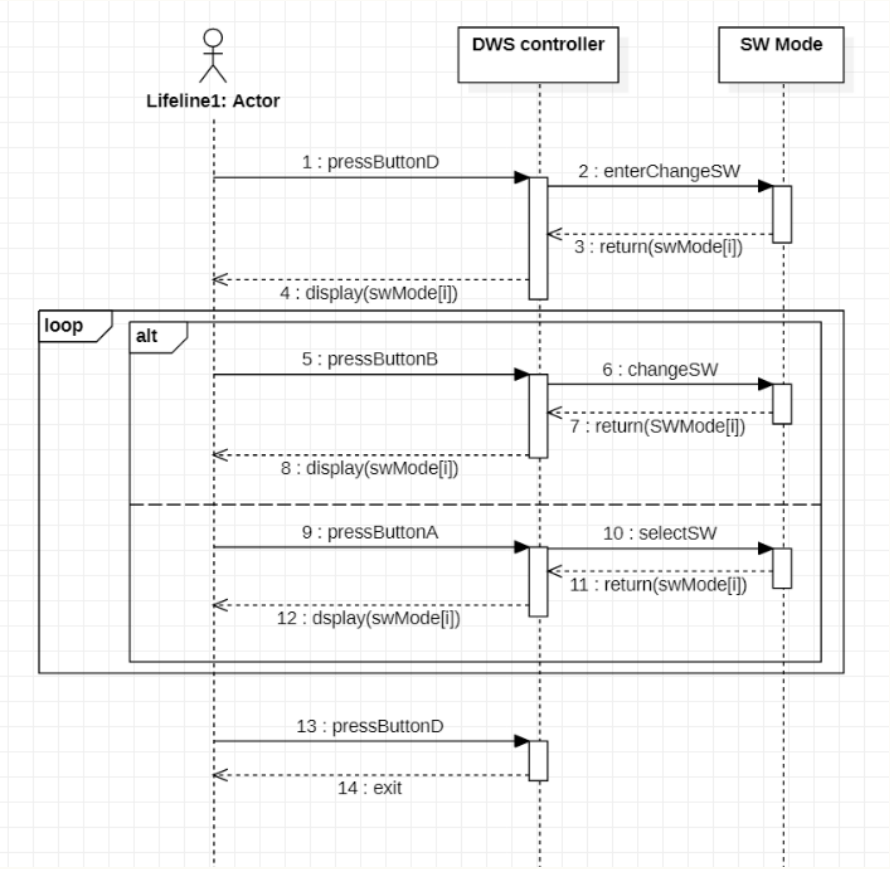
23.


DWS controller
BrightnesscontrolMode
Lifeline1: User
loop
1 : pressButtonB
2 : changeLevel
3 : returnBrightnessLevel
4 : display(brightness)
5 : pressButtonC
6 : controlBrightness()
7 : returnBrightnessLevel
8 : saveBrightnessLevel

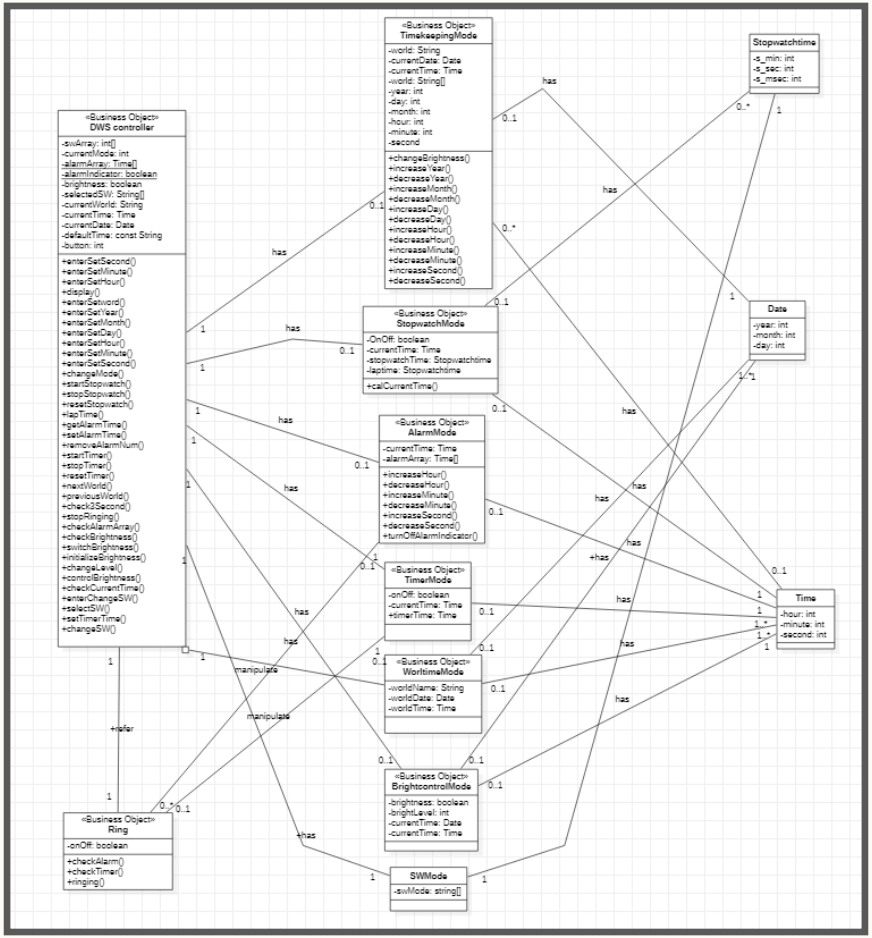
24.


:DWS Controller
:BrightnesscontrolMode
Lifeline1: User
loop everyHour
1 : checkCurrentTime
2 : changeBrightness
3 : returnBrightLevel
4 : display(brightness)

25.


Lifeline1: Actor
DWS controller
SW Mode
1 : pressButtonD
2 : enterChangeSW
3 : return(swMode[i])
4 : display(swMode[i])
loop
alt
5 : pressButtonB
6 : changeSW
7 : return(SWMode[i])
8 : display(swMode[i])
9 : pressButtonA
10 : selectSW
11 : return(swMode[i])
12 : dsplay(swMode[i])
13 : pressButtonD
14 : exit

Activity 2044 :
Define Define Class Diagrams


«Business Object»
TimekeepingMode
-world: String
-currentDate: Date
-currentTime: Time
-world: String[]
-year: int
-day: int
-month: int
-hour: int
-minute: int
-second
+changeBrightness()
+increaseYear()
+decreaseYear()
+increaseMonth()
+decreaseMonth()
+increaseDay()
+decreaseDay()
+increaseHour()
+decreaseHour()
+increaseMinute()
+decreaseMinute()
+increaseSecond()
+decreaseSecond()
Stopwatchtime
-s_min: int
-s_sec: int
-s_msec: int
«Business Object»
DWS controller
-swArray: int[]
-currentMode: int
-alarmArray: Time[]
-alarmIndicator: boolean
-brightness: boolean
-selectedSW: String[]
-currentWorld: String
-currentTime: Time
-currentDate: Date
-defaultTime: const String
-button: int
+enterSetSecond()
+enterSetMinute()
+enterSetHour()
+display()
+enterSetword()
+enterSetYear()
+enterSetMonth()
+enterSetDay()
+enterSetHour()
+enterSetMinute()
+enterSetSecond()
+changeMode()
+startStopwatch()
+stopStopwatch()
+resetStopwatch()
+lapTime()
+getAlarmTime()
+setAlarmTime()
+removeAlarmNum()
+startTimer()
+stopTimer()
+resetTimer()
+nextWorld()
+previousWorld()
+check3Second()
+stopRinging()
+checkAlarmArray()
+checkBrightness()
+switchBrightness()
+initialiseBrightness()
+changeLevel()
+controlBrightness()
+checkCurrentTime()
+enterChangeSW()
+selectSW()
+setTimerTime()
+changeSW()
«Business Object»
StopwatchMode
-OnOff: boolean
-currentTime: Time
-stopwatchTime: Stopwatchtime
-laptime: Stopwatchtime
+calCurrentTime()
Date
-year: int
-month: int
-day: int
«Business Object»
AlarmMode
-currentTime: Time
-alarmArray: Time[]
+increaseHour()
+decreaseHour()
+increaseMinute()
+decreaseMinute()
+increaseSecond()
+decreaseSecond()
+turnOffAlarmIndicator()
«Business Object»
TimerMode
-onOff: boolean
-currentTime: Time
+timerTime: Time
Time
-hour: int
-minute: int
-second: int
«Business Object»
WorltimeMode
-worldName: String
-worldDate: Date
-worldTime: Time
«Business Object»
BrightcontrolMode
-brightness: boolean
-brightLevel: int
-currentTime: Date
-currentTime: Time
«Business Object»
Ring
-onOff: boolean
+checkAlarm()
+checkTimer()
+ringing()
SWMode
-swMode: string[]
has
manipulate
+refer
+has

Activity 2047 :
Perform 2040 Traceability Analysis

Essential UseCase	S-Link	SID	Operation in Sequence Diagram	M-Link	MID	Method	Class
Set time	S1~S8, S2.* ~ S8.*	S1	display	M1	M1	display(): Time	DWS Controller
Show current time	S1	S2	enterSetWorld	M1,M2	M2	enterSetWorld () : void	
Show current date	S1	S3	enterSetSecond	M1,M3	M3	enterSetSecond () : void	
Change mode	S1, S10	S4	enterSetMinute	M1,M4	M4	enterSetMinute () : void	
Stop ringing	S11, S11.1	S5	enterSetHour	M1,M5	M5	enterSetHour () : void	
Start stopwatch	S1, S12	S6	enterSetYear	M1,M6	M6	enterSetYear () : void	
Stop stopwatch	S1, S13	S7	enterSetMonth	M1,M7	M7	enterSetMonth () : void	
Reset stopwatch	S1, S14	S8	enterSetDay	M1,M8	M8	enterSetDay () : void	
Lap stopwatch	S1, S15, S15.1	S9	enterChangeSW	M1,M9	M9	enterChangeSW () : void	
Select alarm number	S1, S16	S10	changeMode	M1,M10	M10	changeMode () : int	
Remove alarm number	S1, S17	S11	stopRinging	M11, M11.1	M11	stopRinging () : void	
Set Alarm Time	S1, S3~S8, S3.* ~ S8.*, S18	S12	startStopwatch	M1,M12	M12	startStopwatch () : void	
Turn on alarm	S1, S19	S13	stopStopwatch	M1,M13	M13	stopStopwatch () : void	
Turn off alarm	S1, S20	S14	resetStopwatch	M1,M14	M14	resetStopwatch () : void	
Set timer time	S1, S3~S8, S3.* ~ S8.*, S21	S15	lapTime	M1,M15,M28	M15	lapTime () : Time	
Start timer	S1, S22	S16	getAlarmTime	M1,M16	M16	getAlarmTime () : Time	
Stop timer	S1, S23	S17	removeAlarmNum	M1,M17	M17	removeAlarmNum () : void	
Reset timer	S1, S24	S18	setAlarmTime	M1, M18	M18	setAlarmTime(Time) : void	
Ringing timer	S25, S25.*	S19	checkAlarmArray	M1,M19	M19	checkAlarmArray : boolean	
Change country	S1, S2.*	S20	turnOffAlarmIndicator	M1,M29	M20	setTimerTime () : void	
Turn on brightness control	S1, S26, S26.1	S21	setTimerTime	M1,M20	M21	startTimer () : void	
Turn off brightness control	S1, S26, S26.*	S22	startTimer	M1,M21	M22	stopTimer () : void	
Control brightness	S1, S27, S27.1	S23	stopTimer	M1,M22	M23	resetTimer () : void	
Change brightness	S1, S28, S28.1	S24	resetTimer	M1,M23	M24	checkBrightness () : boolean	
Change SW	S1, S9, S9.*	S25	checkAlarm	M30, M32	M25	changeLevel (int) : void	
		S26	checkBrightness	M1,M27	M26	checkCurrentTime () : boolean	
		S27	changeLevel	M1,M25	M2.1	nextWorld () : void	
		S28	checkCurrentTime	M26	M2.2	previousWorld () : void	
		S2.1	nextWorld	M1,M2.1	M9.1	selectSW () : void	
		S2.2	previousWorld	M1,M2.2	M9.2	changeSW () : void	
		S3.1	increaseSecond	M1,M3.1	M11.1	check3Second () : boolean	

		S2.1	nextWorld	M1,M2.1	M9.1	selectSW () : void	
		S2.2	previousWorld	M1,M2.2	M9.2	changeSW () : void	
		S3.1	increaseSecond	M1,M3.1	M11.1	check3Second () : boolean	
		S3.2	decreaseSecond	M1,M3.2	M24.1	switchBrightness () : void	
		S4.1	increaseMinute	M1,M4.1	M24.2	initializeBrightness () : void	
		S4.2	decreaseMinute	M1,M4.2	M24.3	controlBrightness () : void	
		S5.1	increaseHour	M1,M5.1	M27	changeBrightness () : void	**TimekeepingMode**
		S5.2	decreaseHour	M1,M5.2	M3.1	increaseSecond () : void	
		S6.1	increaseYear	M1,M6.1	M3.2	decreaseSecond () : void	
		S6.2	decreaseYear	M1,M6.2	M4.1	increaseMinute () : void	
		S7.1	increaseMonth	M1,M7.1	M4.2	decreaseMinute () : void	
		S7.2	decreaseMonth	M1,M7.2	M5.1	increaseHour () : void	
		S8.1	increaseDay	M1,M8.1	M5.2	decreaseHour () : void	
		S8.2	decreaseDay	M1,M8.2	M6.1	increaseYear () : void	
		S9.1	changeSW	M1,M9.1	M6.2	decreaseYear () : void	
		S9.2	selectSW	M1,M9.2	M7.1	increaseMonth () : void	
		S11.1	check3Second	M11.1	M7.2	decreaseMonth () : void	
		S15.1	calCurrentTime	M28	M8.1	increaseDay () : void	
		S25.1	checkTimer	M31,M32	M8.2	decreaseDay () : void	
		S25.2	ringing	M32	M28	calCurrentTime () : Time	**StopwatchMode**
		S26.1	switchBrightness	M1,M24.1	M3.3	increaseSecond () : void	**AlarmMode**
		S26.2	initializeBrightness	M1,M24.2	M3.4	decreaseSecond () : void	
		S27.1	controlBrightness	M24.3	M4.3	increaseMinute () : void	
		S27.2	saveBrightnessLevel	M24.3	M4.4	decreaseMinute () : void	
		S28.1	changeBrightness	M1,M27	M5.3	increaseHour () : void	
					M5.4	decreaseHour () : void	
					M29	turnOffAlarmIndicator () : void	
					M30	checkAlarm () : boolean	**Ring**
					M31	checkTimer () : boolean	
					M32	ringing () : void	
							TimerMode
							WorldtimeMode
							BrightcontrolMode