

# Traveler Digital Watch OOPT Stage 2040

소프트웨어 모델링 및 실습  
[19. 05.13]

201411295 이상훈  
201711394 민하은  
201711395 박성준  
201711423 정종화

# Index

---

**2041. Define Real Use Cases**

**2042. Define Reports, UI and Storyboards**

**2044. Define Interaction Diagrams**

**2045. Define Design Class Diagrams**

**2046. Define Traceability Analysis**

# Define Real Use Cases

	1.Set Time
Actors	User
Purpose	User가 시간/날짜를 설정한다.
Overview	User가 초, 시, 분, 년도, 월, 일 순서로 순환하며 설정한다.
Type	Evident
Cross Reference	Function : R.10.0 Use Case : "Alarm"
Pre-Requisites	Time Keeping 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	<p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> <li>1. (A) 모드변경버튼을 2초 동안 누른다.</li> <li>2. (S) 화면이 시간 설정 화면으로 바뀐다.</li> <li>3. (S) 초 부분이 깜빡거린다.</li> <li>4. (A) 시간설정 버튼을 눌러 초 설정을 하고 시간저장버튼을 누른다.</li> <li>5. (S) 초 설정이 저장된다.</li> <li>6. (S) 시 부분이 깜빡거린다.</li> <li>7. (A) 시간설정버튼을 눌러 시 설정을 하고 시간저장버튼을 누른다.</li> <li>8. (S) 시 설정이 저장된다.</li> <li>9. (S) 분 부분이 깜빡거린다.</li> <li>10. (A) 시간설정버튼을 눌러 분 설정을 하고 시간저장버튼을 누른다.</li> <li>11. (S) 분 설정이 저장된다.</li> <li>12. (S) 년도 부분이 깜빡거린다.</li> <li>13. (A) 시간설정버튼을 눌러 년 설정을 하고 시간저장버튼을 누른다.</li> <li>14. (S) 년도 설정이 저장된다.</li> <li>15. (S) 월 부분이 깜빡거린다.</li> <li>16. (A) 시간설정버튼을 눌러 월 설정을 하고 시간저장버튼을 누른다.</li> <li>17. (S) 월 설정이 저장된다.</li> <li>18. (S) 일 부분이 깜빡거린다.</li> <li>19. (A) 시간설정버튼을 눌러 일 설정을 하고 시간저장버튼을 누른다.</li> <li>20. (S) 일 설정이 저장된다.</li> <li>21. (A) 모드변경버튼을 눌러 시간 설정을 종료한다.</li> <li>22. (S) 저장된 시간대로 Time Keeping 화면에 표시된다.</li> </ol>
Alternative Courses of Events	<p>E*. 시간 설정 화면의 어느 순간이던지 시간 설정 종료를 요청할 수 있다. 그 경우 마지막에 저장된 시간대로 Time Keeping 화면에 표시된다.</p> <p>E25. 일 설정에서 다음 설정을 요청할 경우 초 설정으로 순환한다.</p>
Exceptional Courses of Events	E2. User가 시간 설정 종료를 요청하지 않는다면 끝나지 않는다.

# Define Real Use Cases

Use Case	5.Start Stopwatch
Actors	User
Purpose	User가 스톱워치를 시작한다.
Overview	User가 스톱워치의 시간을 흐르게 한다.
Type	Evident
Cross Reference	Function : R.10.0 Use Case : "Alarm"
Pre-Requisites	Stopwatch 시간 멈춤 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 스톱워치 시작 버튼을 누른다. 2. (S) 스톱워치 시간을 흐르게 한다. 3. (S) 59분 59초가 되면 화면이 <분 초 0.01초>에서 <시 분 초>로 바뀐다.
Alternative Courses of Events	
Exceptional Courses of Events	E1. 스톱워치가 동작 상태이면 멈춘다. E4. 23시간 59분 59초가 되면 스톱워치를 0으로 초기화 해주고 멈춘다.

# Define Real Use Cases

Use Case	6.Stop Stopwatch
Actors	User
Purpose	User가 스톱워치를 멈추게 한다.
Overview	User가 스톱워치의 시간을 멈추게 한다.
Type	Evident
Cross Reference	Function : R.3.0 Use Case : "Start Stopwatch"
Pre-Requisites	Stopwatch 시간 동작 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 스톱워치 동작멈춤 버튼을 누른다. 2. (S) 스톱워치 시간의 흐름을 멈춘다.
Alternative Courses of Events	
Exceptional Courses of Events	E1. 스톱워치가 멈춤 상태이면 동작한다.

# Define Real Use Cases

Use Case	7. Reset Stopwatch
Actors	User
Purpose	User가 스톱워치를 초기화한다.
Overview	User가 스톱워치의 설정된 시간을 0으로 초기화한다.
Type	Evident
Cross Reference	Function : R.3.1 Use Case : "Stop Stopwatch"
Pre-Requisites	Stopwatch 시간 멈춤 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 스톱워치 초기화 버튼을 누른다. 2. (S) 스톱워치 시간을 0으로 초기화한다.
Alternative Courses of Events	
Exceptional Courses of Events	

# Define Real Use Cases

Use Case	12.Set Number Range
Actors	User
Purpose	User가 난수 숫자 범위를 설정한다.
Overview	User가 난수를 생성할 숫자 범위를 설정한다.
Type	Evident
Cross Reference	Function : R.5.1 Use Case : "Generate Number Range"
Pre-Requisites	난수 범위 설정 화면 상태, 올바른 기능 요청
Typical Courses of Events	1. (A) 범위 설정을 위해 UP button 을 누른다 2. (S) 저장되어 있는 숫자 범위에 1을 더해주고, '시간' 화면에 값을 표시한다.
Alternative Courses of Events	
Exceptional Courses of Events	E2. 숫자가 59이후에 1로 돌아온다.

# Define Real Use Cases

Use Case	13.Generate Random Number
Actors	User
Purpose	User에게 난수를 생성해준다.
Overview	User가 난수를 요청하면 set number range로 저장되어 있는 수 범위내의 수를 복원 추출하여 보여준다.
Type	Evident
Cross Reference	Function : R.5.0 Use Case : "Set Number Range"
Pre-Requisites	난수 모드 상태, 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 난수 생성을 위해 make button 을 누른다. 2. (S) '분'을 표시하는 화면에 범위에 맞는 난수를 표시해준다.
Alternative Courses of Events	
Exceptional Courses of Events	



# Define Real Use Cases

Use Case	14.Reset Number Range
Actors	User
Purpose	난수의 생성 범위를 1로 초기화 한다.
Overview	User가 난수의 생성 범위 숫자를 1로 초기화한다.
Type	Evident
Cross Reference	Function : R.5.1 Use Case : "Generate Number Range"
Pre-Requisites	난수 모드 상태, 올바른 기능 요청
Typical Courses of Events	(A) : Actor, (S) : System 1. (A) 난수 생성 범위 초기화를 위해 reset button을 누른다. 2. (S) 난수 범위 값을 1로 초기화해주고, '시간' 화면에 표시해준다.
Alternative Courses of Events	
Exceptional Courses of Events	

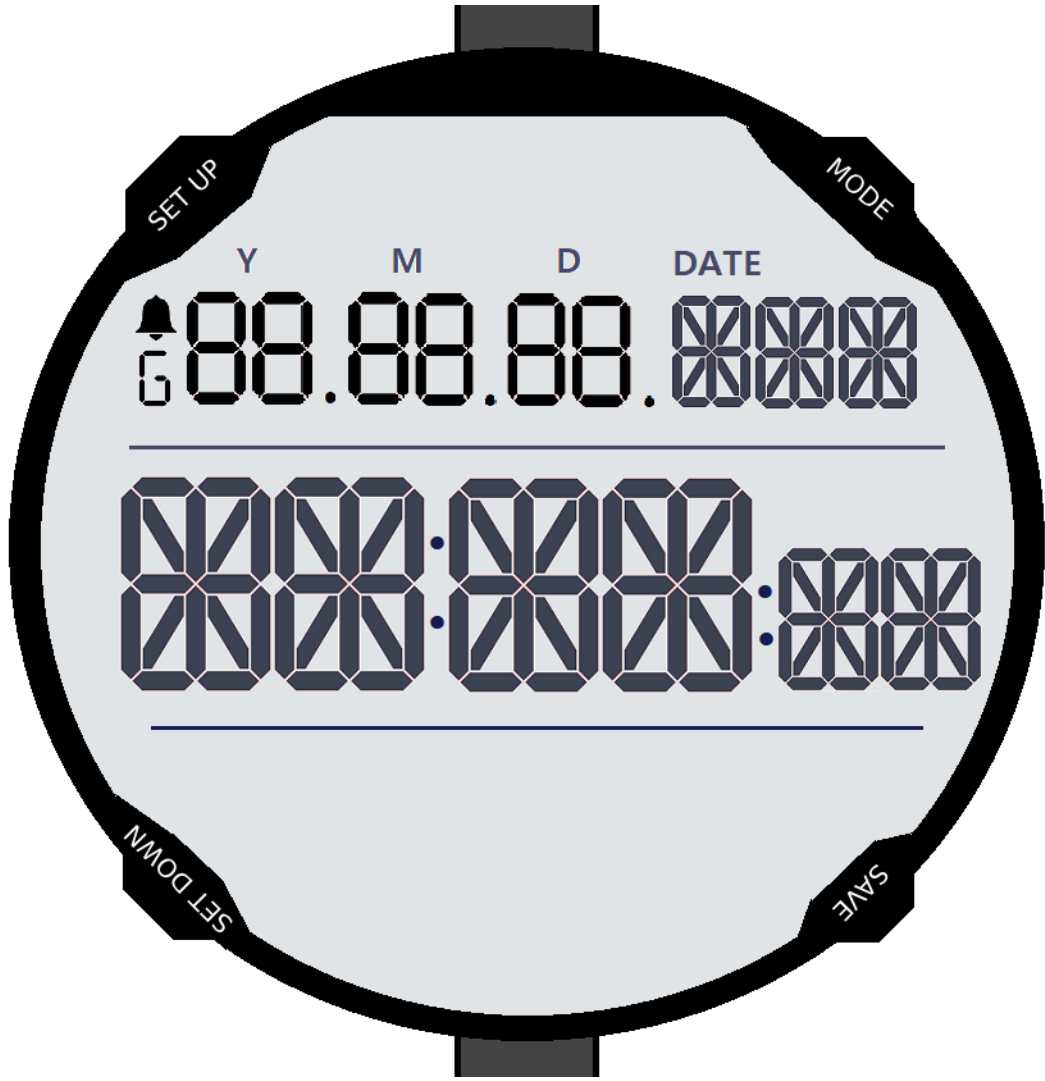
# Define Real Use Cases

Use Case	15.Set Global Time
Actors	User
Purpose	User가 여행지의 시간을 설정한다.
Overview	User가 여행지의 시, 분 순서로 순환하며 설정한다.
Type	Evident
Cross Reference	
Pre-Requisites	여행지 시간 설정 화면 상태, User의 올바른 기능 요청
Typical Courses of Events	<p>(A) : Actor, (S) : System</p> <ol style="list-style-type: none"> <li>1. (A) 여행지 시간 화면이 나올 때까지 모드변경버튼을 누른다.</li> <li>2. (S) 화면이 여행지 시간 화면으로 바뀐다.</li> <li>3. (A) 모드변경버튼을 2초 동안 누른다.</li> <li>2. (S) 화면이 시차 설정 화면으로 바뀐다.</li> <li>3. (S) 시간 부분이 깜빡거린다.</li> <li>4. (A) 시간설정버튼을 눌러 시 설정을 하고 시간저장버튼을 누른다.</li> <li>5. (S) 시간 설정이 저장된다.</li> <li>7. (S) 분 부분이 깜빡거린다.</li> <li>8. (A) 시간설정버튼을 눌러 분 설정을 하고 시간저장버튼을 누른다.</li> <li>9. (S) 분 설정이 저장된다.</li> <li>10. (A) 모드변경버튼을 눌러서 여행지 시차 설정 종료한다.</li> <li>11. (S) 기존 시간과 시차를 연산하여 여행지 시간 화면에 표시된다.</li> </ol>
Alternative Courses of Events	<p>E*. 시간 설정 화면의 어느 순간이던지 시간 설정 종료를 요청할 수 있다. 그 경우 마지막에 저장된 시간대로 global time 화면에 표시된다.</p> <p>E9. 분 설정에서 다음 설정을 요청할 경우 시 설정으로 순환한다.</p>
Exceptional Courses of Events	E2. User가 여행지 시간 설정 종료를 요청하지 않는다면 끝나지 않는다.

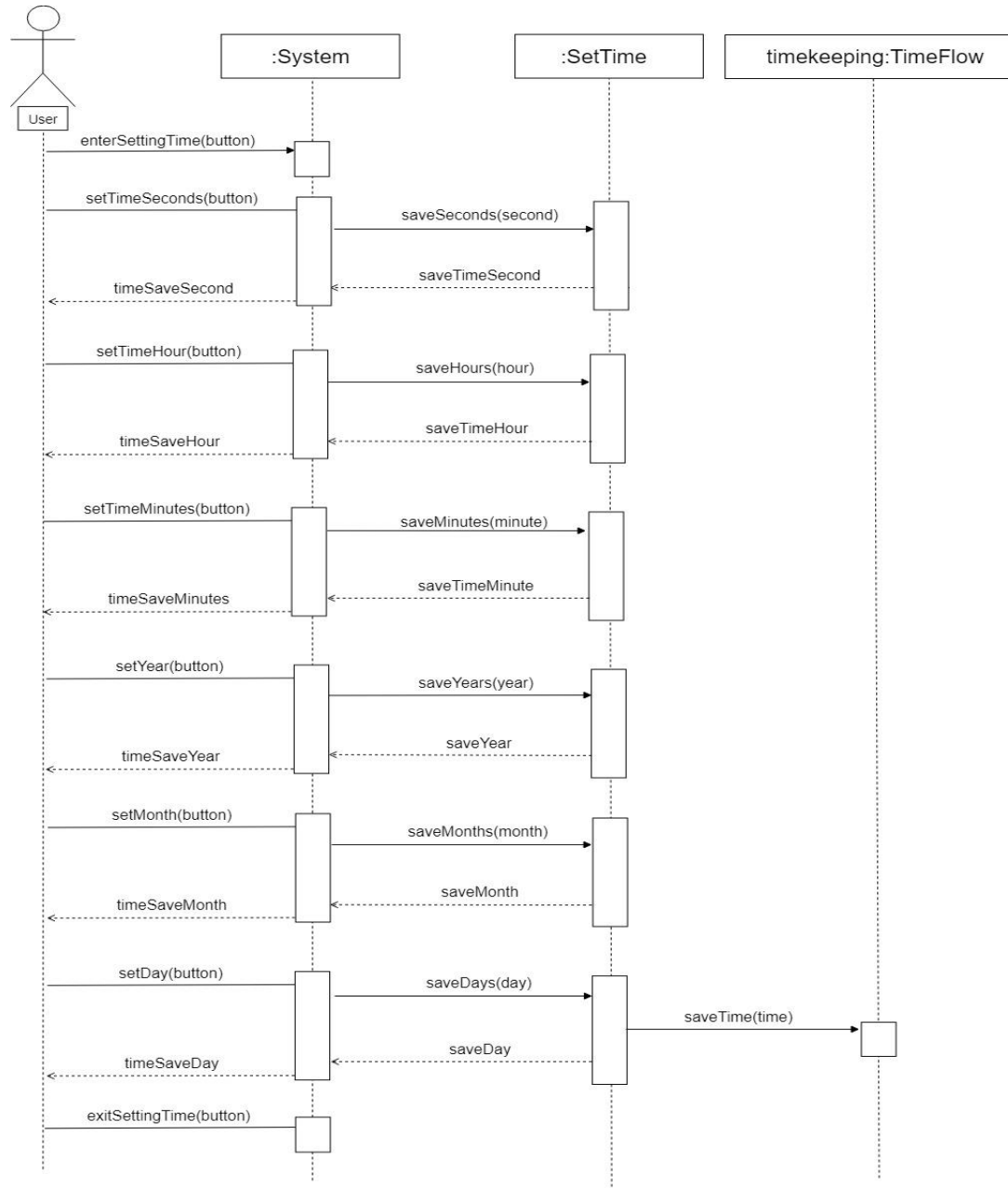
# Define Real Use Cases

Use Case	18.Display
Actors	System
Purpose	System이 User에게 현재 상태에 알맞은 화면을 나타내준다.
Overview	System이 User가 작동한 결과에 맞추어 User에게 알맞은 화면을 나타내준다. 시간의 흐름에 맞추어 알맞은 화면을 나타내준다.
Type	Hidden
Cross Reference	
Pre-Requisites	Any time
Typical Courses of Events	(A) : Actor, (S) : System 1. (S) System 이 정보를 저장 및 계산한다. 2. (S) System 현재 정보를 User에게 보여준다.
Alternative Courses of Events	
Exceptional Courses of Events	

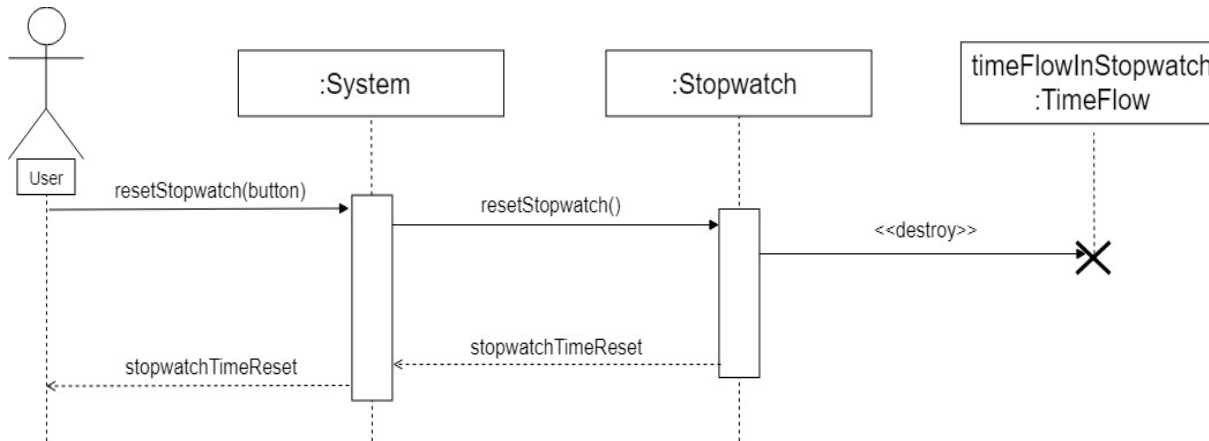
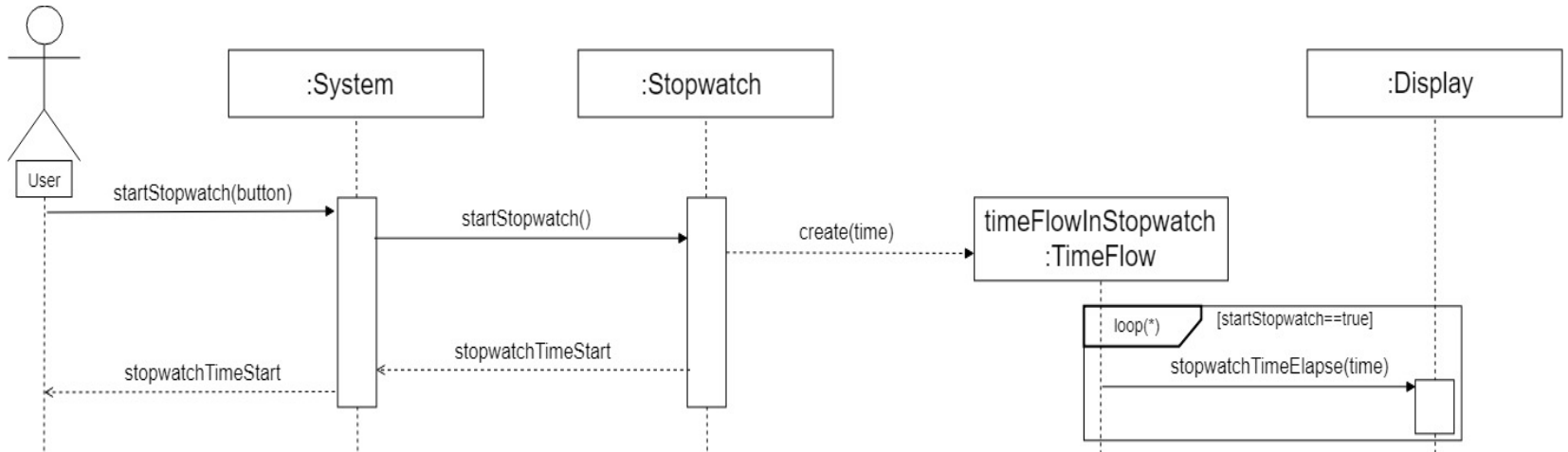
# Define Reports, UI and Storyboards



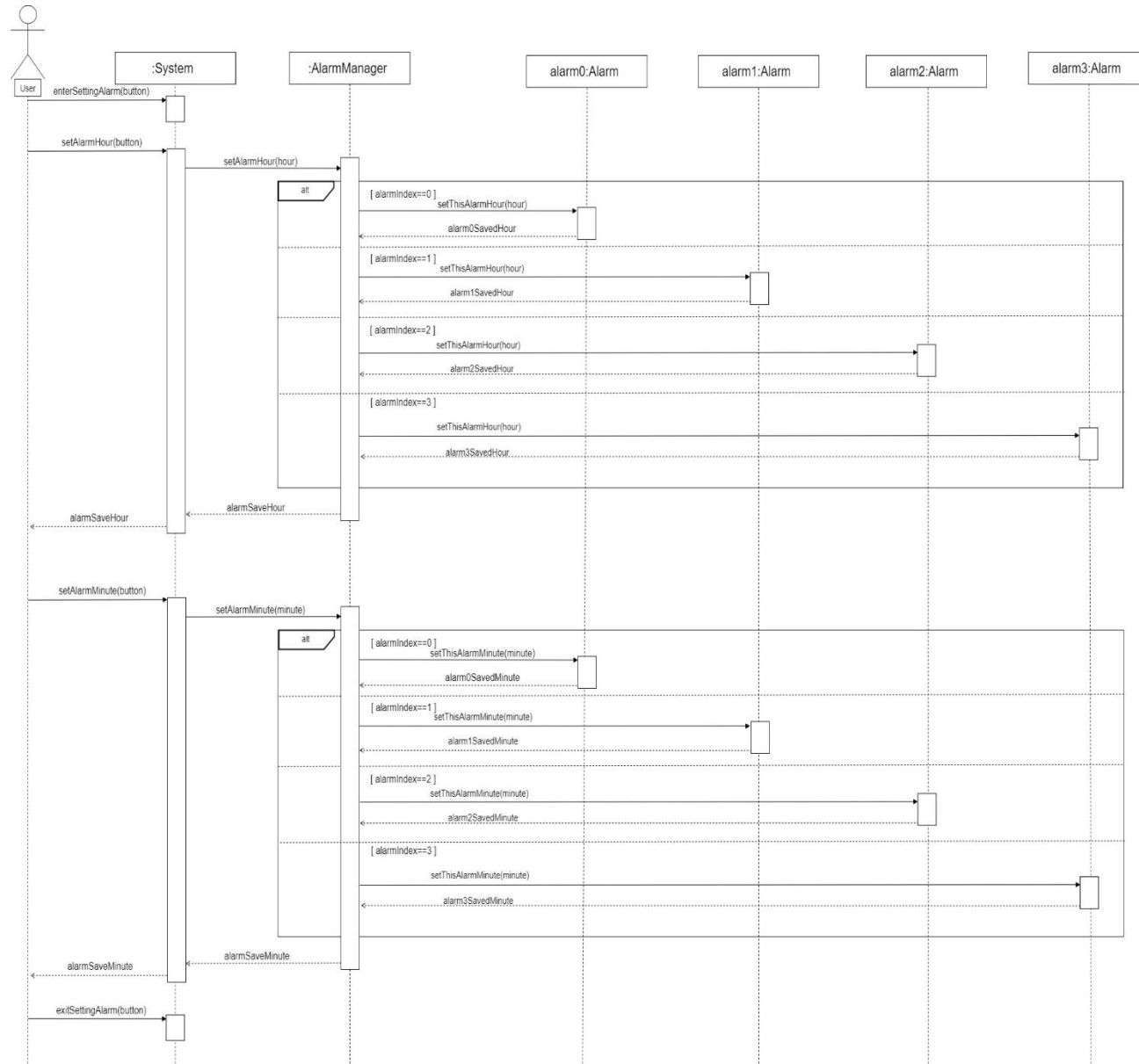
# Define Interaction Diagram



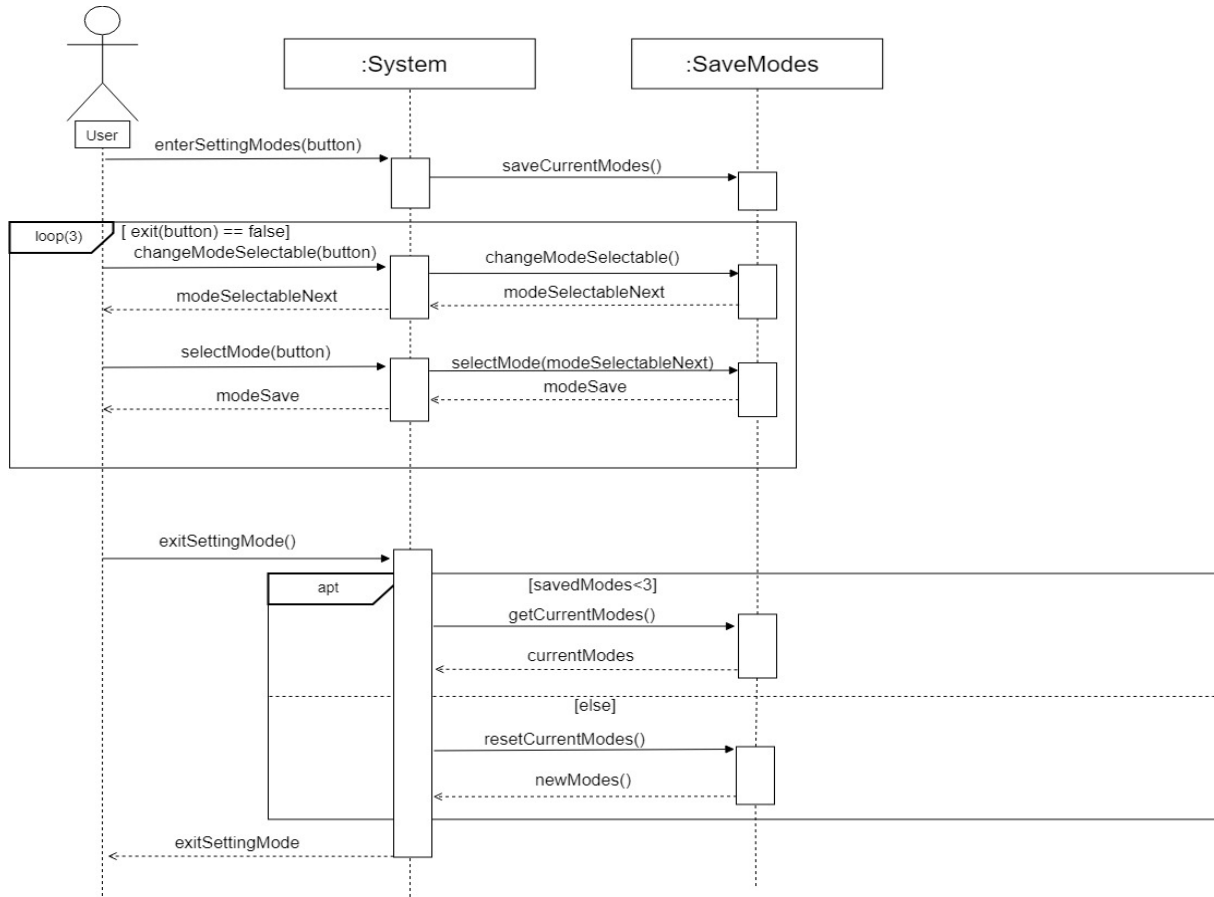
# Define Interaction Diagram



# Define Interaction Diagram

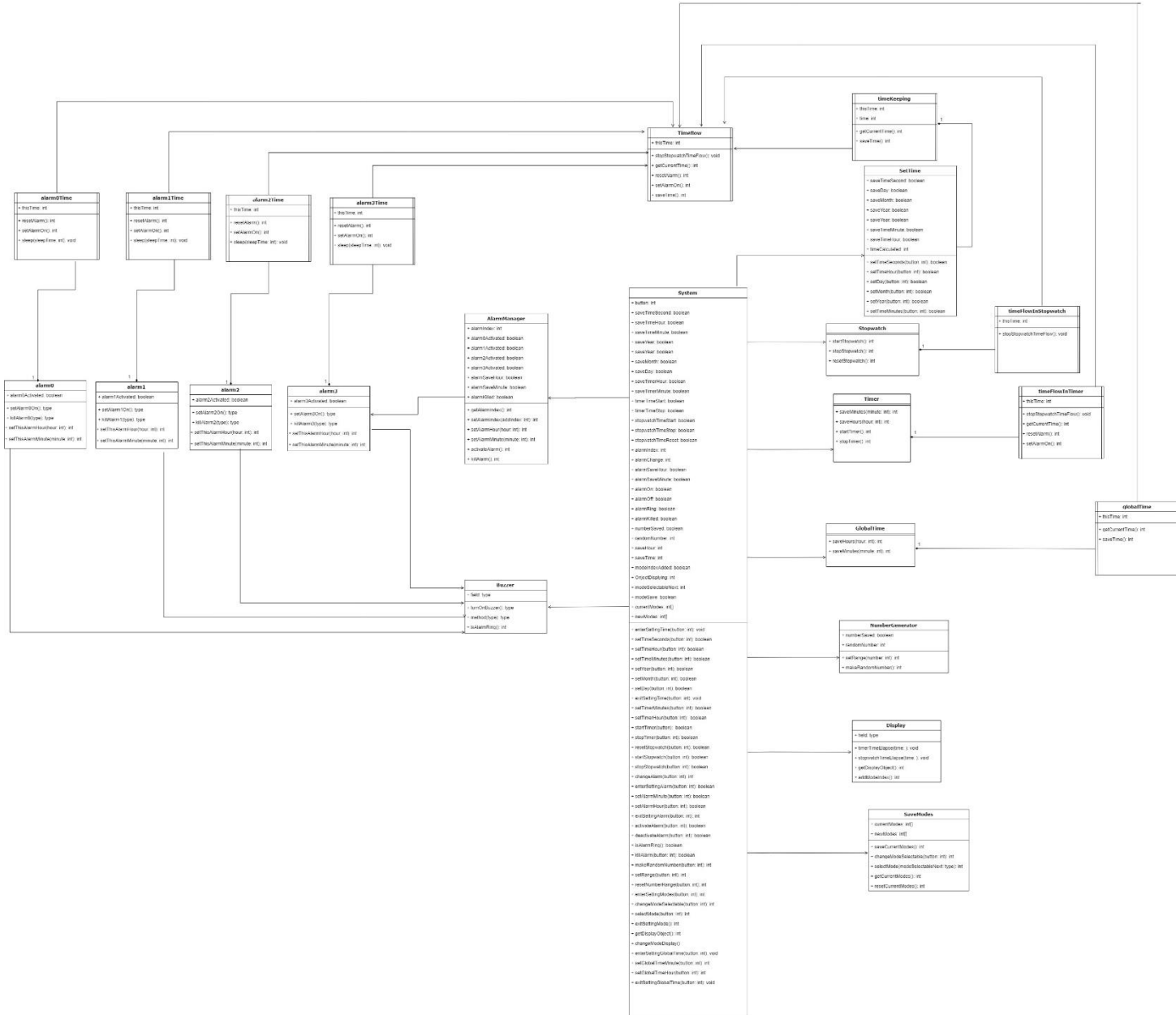


# Define Interaction Diagram

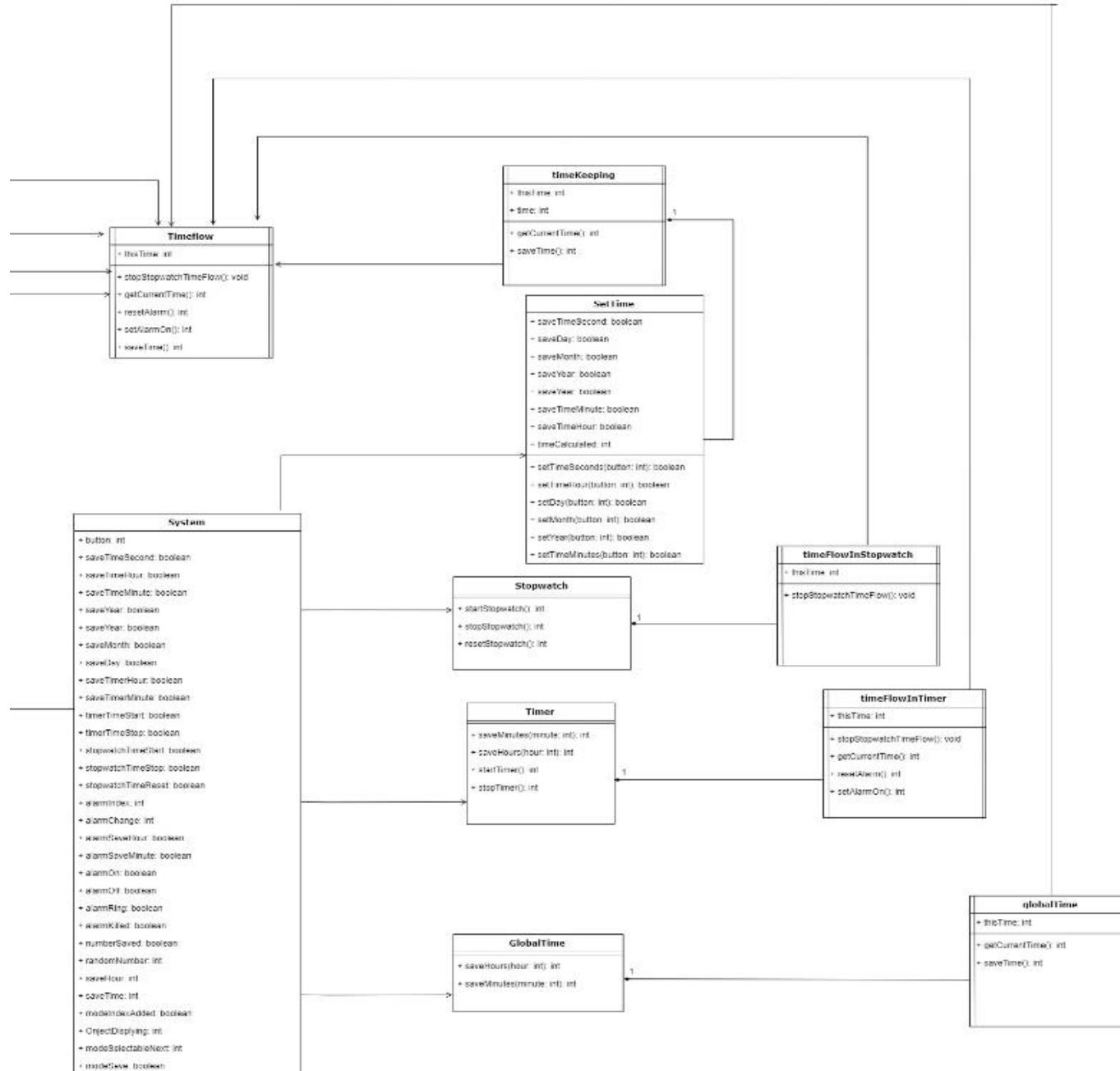




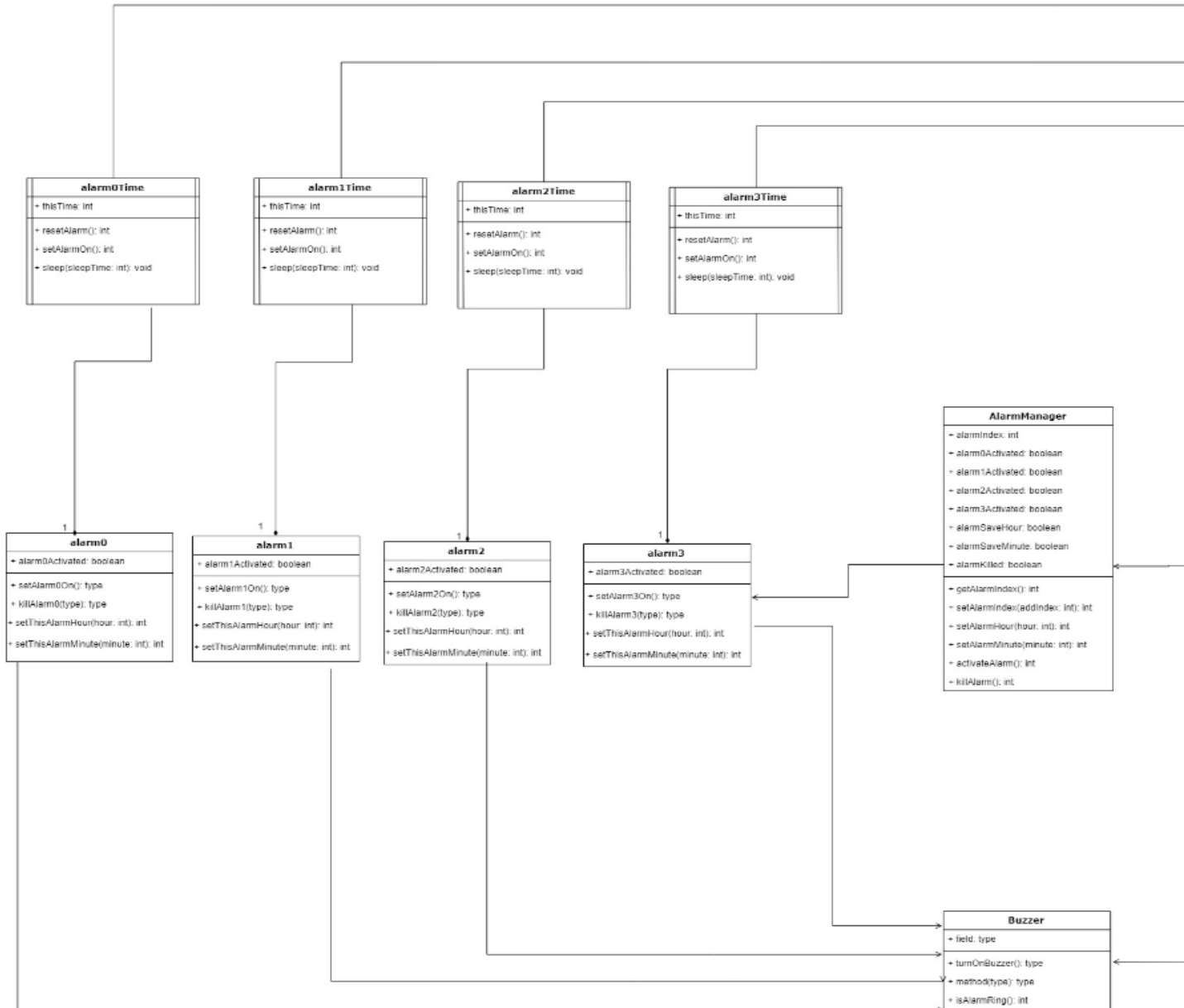
# Define Design Class Diagram



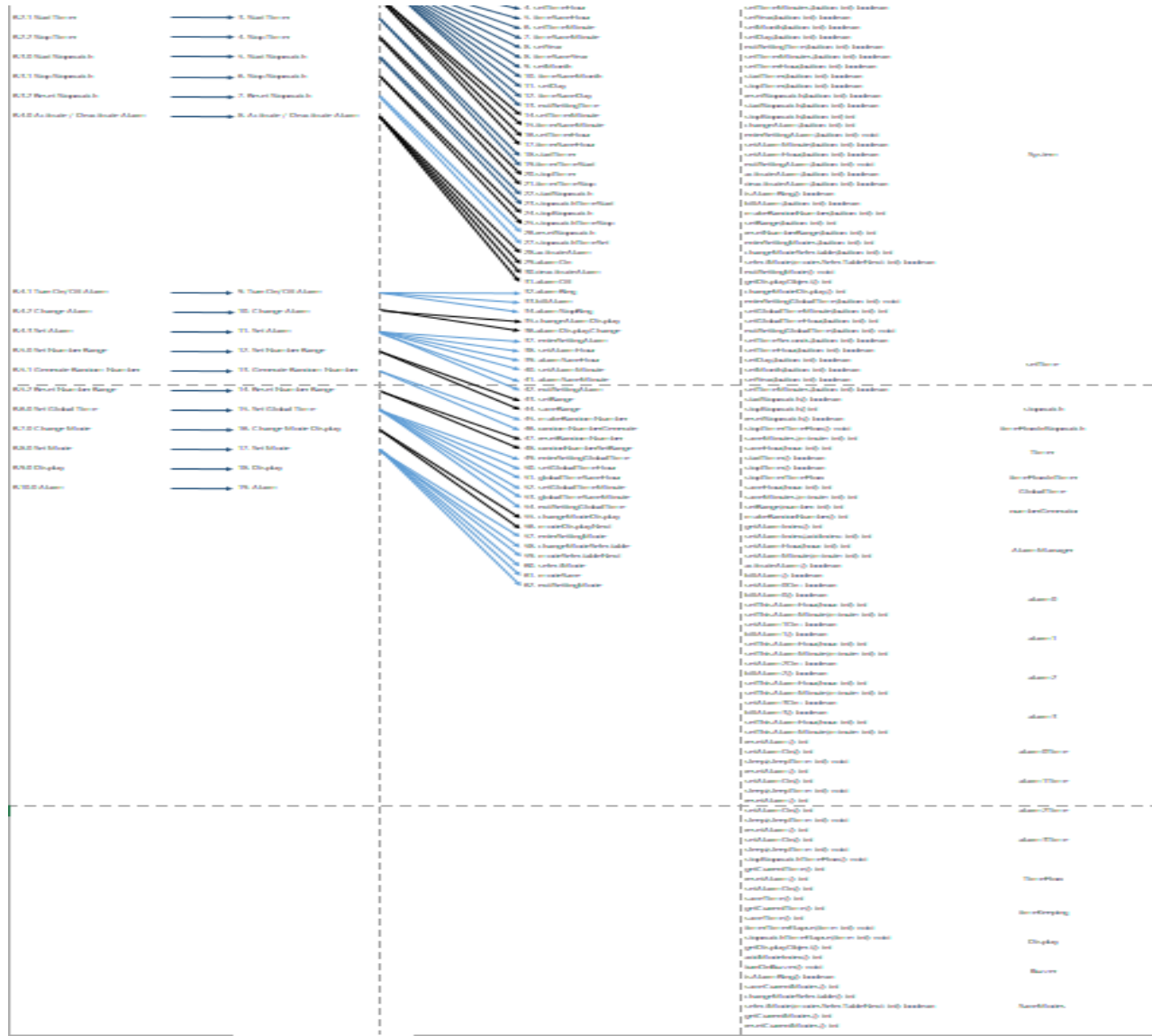
# Define Design Class Diagram



# Define Design Class Diagram



# Design Traceability Analysis



# Design Traceability Analysis

Functional Requirement	User Case	System Operation	method	class
R.1.0. Set Time	→ 1. Set Time	1. enterSettingTime	enterSettingTime(button: int): void	System
R.2.0 Set Timer	→ 2. Set Timer	2. setTimeSecond	setTimeSeconds(button: int): boolean	
R.2.1 Start Timer	→ 3. Start Timer	3. timeSaveSecond	setTimeHour(button: int): boolean	
R.2.2 Stop Timer	→ 4. Stop Timer	4. setTimeHour	setTimeMinutes(button: int): boolean	
R.3.0 Start Stopwatch	→ 5. Start Stopwatch	5. timeSaveHour	setYear(button: int): boolean	
R.3.1 Stop Stopwatch	→ 6. Stop Stopwatch	6. setTimeMinute	setMonth(button: int): boolean	
R.3.2 Reset Stopwatch	→ 7. Reset Stopwatch	7. timeSaveMinute	setDay(button: int): boolean	
R.4.0 Activate / Deactivate Alarm	→ 8. Activate / Deactivate Alarm	8. setYear	exitSettingTime(button: int): boolean	
		8. timeSaveYear	setTimerMinutes(button: int): boolean	
		9. setMonth	setTimerHour(button: int): boolean	
		10. timeSaveMonth	startTimer(button: int): boolean	
		11. setDay	stopTimer(button: int): boolean	
		12. timeSaveDay	resetStopwatch(button: int): boolean	
		13. exitSettingTime	startStopwatch(button: int): boolean	
		14.setTimerMinute	stopStopwatch(button: int): int	
		15.timerSaveMinute	changeAlarm(button: int): int	
		16.setTimerHour	enterSettingAlarm(button: int): void	
		17.timerSaveHour	setAlarmMinute(button: int): boolean	
		18.startTimer	setAlarmHour(button: int): boolean	
		19.timerTimeStart	exitSettingAlarm(button: int): void	
		20.stopTimer	activateAlarm(button: int): boolean	
		21.timerTimeStop	deactivateAlarm(button: int): boolean	
		22.startStopwatch	isAlarmRing(): boolean	
		23.stopwatchTimeStart	killAlarm(button: int): boolean	
		24.stopStopwatch	makeRandonNumber(button: int): int	
		25.stopwatchTimeStop	setRange(button: int): int	
		26.resetStopwatch	resetNumberRange(button: int): int	
		27.stopwatchTimeSet	enterSettingModes(button: int): int	
		28.activateAlarm	changeModeSelectable(button: int): int	
		29.alarmon	selectMode(modesSelectTableNext: int): boolean	
		30.deactivateAlarm	exitSettingMode(): void	
		31.alarloff	getDisplayObject(): int	
R.4.1 Turn On/Off Alarm	→ 9. Turn On/Off Alarm	32.alarMring	changeModeDisplay(): int	
R.4.2 Change Alarm	→ 10. Change Alarm	33.killAlarm	enterSettingGlobalTime(button: int): void	
		34.alarMstopRing	setGlobalTimeMinute(button: int): int	
		35.changeAlarmDisplay	setGlobalTimeHour(button: int): int	
R.4.3 Set Alarm	→ 11. Set Alarm	36.alarMdisplayChange	exitSettingGlobalTime(button: int): void	

# Design Traceability Analysis

R.4.1 Turn On/Off Alarm	→	9. Turn On/Off Alarm	→	32.alarmRing	→	changeModeDisplay(): int	setTime
R.4.2 Change Alarm	→	10. Change Alarm	→	33.killAlarm	→	enterSettingGlobalTime(button: int): void	
R.4.3 Set Alarm	→	11. Set Alarm	→	34.alarmStopRing	→	setGlobalTimeMinute(button: int): int	
R.5.0 Set Number Range	→	12. Set Number Range	→	35.changeAlarmDisplay	→	setGlobalTimeHour(button: int): int	
R.5.1 Generate Random Number	→	13. Generate Random Number	→	36.alarmDisplayChange	→	exitSettingGlobalTime(button: int): void	
R.5.2 Reset Number Range	→	14. Reset Number Range	→	37. enterSettingAlarm	→	setTimeSeconds(button: int): boolean	
R.6.0 Set Global Time	→	15. Set Global Time	→	38. setAlarmHour	→	setTimeHour(button: int): boolean	
R.7.0 Change Mode	→	16. Change Mode Display	→	39. alarmSaveHour	→	setDay(button: int): boolean	
R.8.0 Set Mode	→	17. Set Mode	→	40. setAlarmMinute	→	setMonth(button: int): boolean	
R.9.0 Display	→	18. Display	→	41. alarmSaveMinute	→	setYear(button: int): boolean	
R.10.0 Alarm	→	19. Alarm	→	42. exitSettingAlarm	→	setTimeMinutes(button: int): boolean	
				43. setRange	→	startStopwatch(): boolean	stopwatch
				44. saveRange	→	stopStopwatch() int	
				45. makeRandomNumber	→	resetStopwatch(): boolean	timeFlowInStopwatch
				46. randomNumberGenerate	→	stopTimerTimeFlow(): void	
				47. resetRandomNumber	→	saveMinutes(minute: int): int	Timer
				48. randonNumberSetRange	→	saveHour(hour: int): int	
				49. enterSettingGlobalTime	→	startTimer(): boolean	timeFlowInTimer
				50. setGlobalTimeHour	→	stopTimer(): boolean	
				51. globalTimeSaveHour	→	stopTimerTimeFlow	GlobalTime
				52. setGlobalTimeMinute	→	saveHour(hour: int): int	
				53. globalTimeSaveMinute	→	saveMinutes(minute: int): int	numberGenerator
				54. exitSettingGlobalTime	→	setRange(number: int): int	
				55. changeModeDisplay	→	makeRandonNumber(): int	AlarmManager
				56. modeDisplayNext	→	getAlarmIndex(): int	
				57. enterSettingMode	→	setAlarmIndex(addIndex: int): int	
				58. changeModeSelectable	→	setAlarmHour(hour: int): int	
				59. modeSelectableNext	→	setAlarmMinute(minute: int): int	
				60. selectMode	→	activateAlarm(): boolean	alarm0
				61. modeSave	→	killAlarm(): boolean	
				62. exitSettingMode	→	setAlarm0On : boolean	
						killAlarm0(): boolean	alarm1
						setThisAlarmHour(hour: int): int	
						setThisAlarmMinute(minute: int): int	alarm2
						setAlarm1On : boolean	
						killAlarm1(): boolean	alarm2
						setThisAlarmHour(hour: int): int	
						setThisAlarmMinute(minute: int): int	
						setAlarm2On : boolean	
						killAlarm2(): boolean	
						setThisAlarmHour(hour: int): int	
						setThisAlarmMinute(minute: int): int	

# Design Traceability Analysis

resetAlarm(): int	alarm3Time
setAlarmOn(): int	
sleep(sleepTime: int): void	
stopStopwatchTimeFlow(): void	TimeFlow
getCurrentTime(): int	
resetAlarm(): int	
setAlarmOn(): int	
saveTime(): int	timeKeeping
getCurrentTime(): int	
saveTime(): int	Display
timerTimerElapse(time: int): void	
stopwatchTimeElapse(time: int): void	
getDisplayObject(): int	
addModeIndex(): int	
turnOnBuzzer(): void	Buzzer
isAlarmRing(): boolean	SaveModes
saveCurrentModes(): int	
changeModeSelectable(): int	
selectMode(modesSelectTableNext: int): boolean	
getCurrentModes(): int	
resetCurrentModes(): int	

---

감사합니다

Q & A

---