

# A Remind Watch

## Software Modeling & Analysis

### Team 3

201511251 김우진

201511263 박종엽

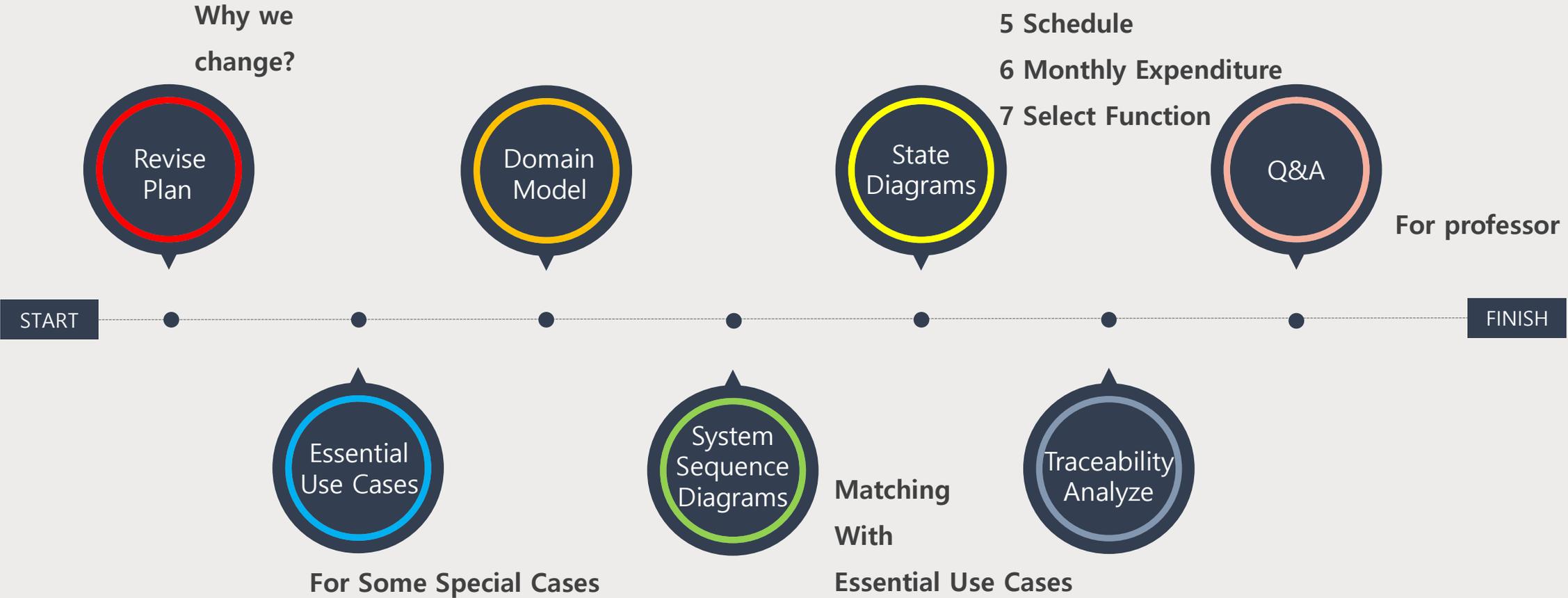
201511279 이상현

201511283 이정노

# INDEX

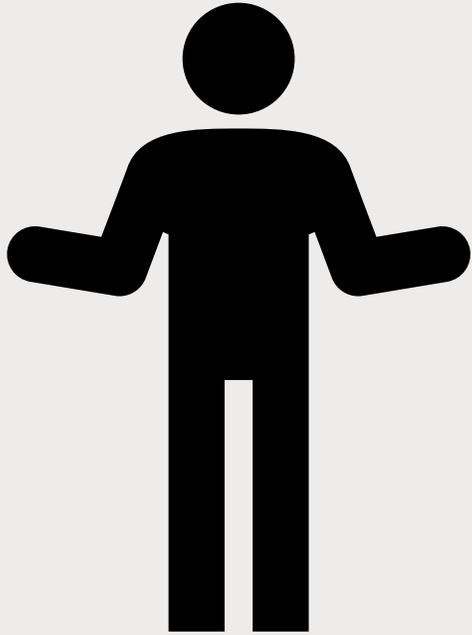
목차

- 1 Whole Diagram
- 2 Current Time
- 3 Timer
- 4 Alarm
- 5 Schedule
- 6 Monthly Expenditure
- 7 Select Function



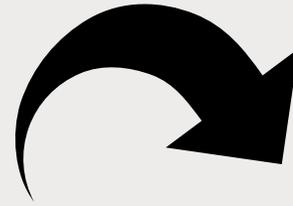
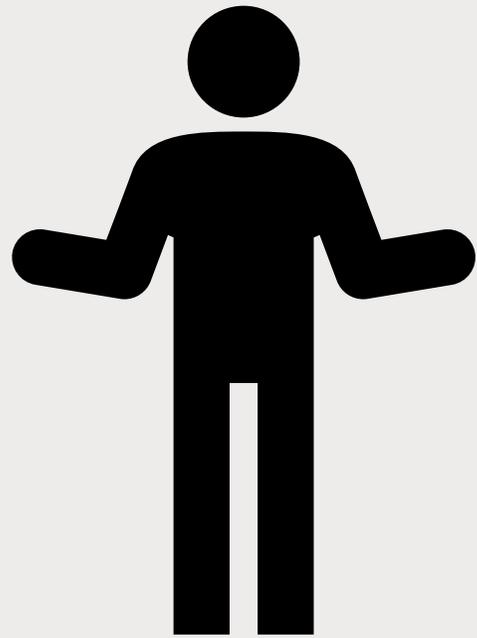
# *Revise Plan*

Why?

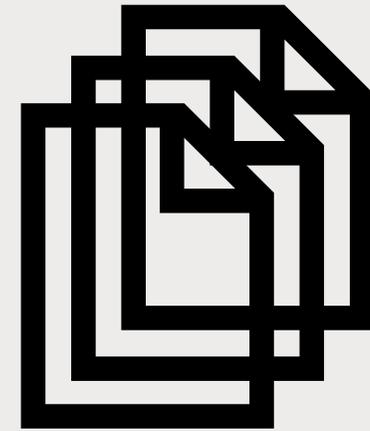
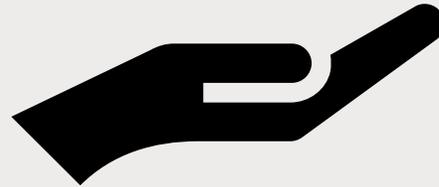


# Revise Plan

Why?



Only  
4 Buttons



Require  
Many  
Functions

# *Revise Plan*

- \* **CurrentTime**
- \* **Stopwatch**
- Timer**
- Alarm**
- \* **Schedule**
- \* **Expenditure**

# Select Function

# Define Essential Use Cases

Use Case	Change current time
<b>Actor</b>	User
<b>Purpose</b>	현재 시각을 새로 설정한다.
<b>Overview</b>	커서 이동 버튼을 눌러 디스플레이에 표시되는 현재 시간에서 수정할 항목으로 이동한다. 커서 이동 순서는 초/분/시/일/월/년 순이다. 값 증가 버튼을 눌러 현재 수정할 항목의 값을 증가시킨다.
<b>Type</b>	Evident
<b>Cross Reference</b>	
<b>Pre-Requisites</b>	각 버튼의 기능이 change current time에 사용되는 기능들로 설정되어 있어야 한다.
<b>Typical Courses of Events</b> <b>(A) : Actors (S) : System</b>	1.(A) : 커서 이동 버튼을 눌러 수정할 항목으로 커서를 옮긴다. 2.(A) : 값 증가 버튼을 눌러 현재 수정할 항목의 값을 증가시킨다. 3.(A) : 설정 완료 버튼을 눌러 현재 시각 설정을 완료한다. 4.(S) : 설정한 시각을 현재 시각으로 설정한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 값 증가 버튼을 눌렀을 때 그 항목의 최대값이면 최소값으로 돌아간다. E2. 값 증가 버튼을 눌렀을 때 현재 수정할 항목이 초이면 초를 0으로 초기화한다. E3. 커서 이동 버튼을 눌렀을 때 현재 수정할 항목이 년이면 초로 돌아간다. E4. 기능 전환 버튼을 눌렀을 때 현재 시각 수정을 하고 있었다면, 그 시각을 현재 시각으로 설정한다.

# Define Essential Use Cases

Use Case	Set timer
Actor	User
Purpose	타이머를 설정한다.
Overview	<p>커서 이동 버튼을 눌러 수정시킬 항목으로 이동한다.</p> <p>커서 이동 순서는 초/분/시 순이다.</p> <p>증가 버튼을 눌러 현재 수정할 항목을 증가시킨다.</p> <p>수정 완료 버튼을 눌러 타이머 설정을 완료한다.</p>
Type	Evident
Cross Reference	
Pre-Requisites	각 버튼의 기능이 Set Timer의 기능들로 설정되어 있어야 한다.
Typical Courses of Events (A) : Actors (S) : System	1.(A) : 커서 이동 버튼을 눌러 수정시킬 항목으로 이동한다. 2.(A) : 증가 버튼을 눌러 현재 수정할 항목을 증가시킨다. 3.(A) : 수정 완료 버튼을 눌러 타이머 설정을 완료한다. 4.(S) : 현재 타이머 설정을 저장한다.
Alternative Courses of Events	N/A
Exceptional Courses of Events	E1. 기존에 저장된 타이머가 있는 경우, 해당 타이머 설정에서 설정을 재개한다. E2. 커서가 시를 가리키는 상태에서 커서를 이동시키면 커서를 초로 이동시킨다. E3. 각 항목이 표현하는 최대값을 넘기면 최소 표현 값으로 변환된다.

# Define Essential Use Cases

Use Case	Set Alarm
<b>Actor</b>	User
<b>Purpose</b>	알람을 설정한다.
<b>Overview</b>	알람 전환 버튼을 눌러 설정할 알람으로 이동한다. 커서 이동 버튼을 눌러 수정할 항목으로 이동한다. 커서 이동 순서는 ON/OFF/분/시/월/화/수/목/금/토/일 순이다. 값 증가 버튼을 눌러 현재 수정할 항목의 값을 증가한다.
<b>Type</b>	Evident
<b>Cross Reference</b>	
<b>Pre-Requisites</b>	각 버튼의 기능이 Set Alarm에 사용되는 기능들로 설정되어 있어야 한다.
<b>Typical Courses of Events</b> <b>(A) : Actors (S) : System</b>	1.(A) : 알람 전환 버튼을 눌러 수정할 알람으로 이동한다. 2.(A) : 커서 이동 버튼을 눌러 수정할 항목으로 이동한다. 3.(A) : 값 증가 버튼으로 수정할 항목의 값을 증가시킨다. 4.(A) : 알람 설정을 확정하기 위해 기능 전환/알람 전환 버튼을 누른다. 5.(S) : 알람 설정 값을 저장한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 네번째 알람을 보여주는 상태에서 알람 전환 버튼을 누르는 경우 첫번째 알람으로 돌아간다. E2. 커서가 '일요일'을 가리키는 상태에서 커서를 이동시키면 어느 커서도 가리키지 않는 처음 상태로 돌아간다. E3. 값 증가 버튼을 눌렀을 때 어떤 커서도 가리키지 않는 경우 아무 동작도 하지 않는다. E4. 값 증가 버튼을 눌렀을 때 현재 커서가 ON/OFF 이거나 요일을 가리키는 상태라면 값을 설정/해제한다.

# Define Essential Use Cases

Use Case	Enroll Schedule
<b>Actor</b>	User
<b>Purpose</b>	스케줄을 등록한다.
<b>Overview</b>	<p>커서 이동 버튼을 눌러 스케줄에서 수정할 항목으로 이동한다.</p> <p>커서 이동 순서는 분/시/일/월/년/일정종류 순이다.</p> <p>값 증가 버튼을 눌러 현재 수정할 항목의 값을 증가한다.</p> <p>스케줄 등록 버튼을 눌러 설정한 스케줄을 등록한다.</p>
<b>Type</b>	Evident
<b>Cross Reference</b>	
<b>Pre-Requisites</b>	각 버튼의 기능이 Enroll Schedule에 사용되는 기능들로 설정되어야 한다.
<b>Typical Courses of Events</b> <b>(A) : Actors (S) : System</b>	<p>1.(A) : 커서 이동 버튼을 눌러 수정할 항목으로 이동한다.</p> <p>2.(A) : 값 증가 버튼을 눌러 현재 수정할 항목의 값을 증가한다.</p> <p>3.(A) : 스케줄 등록 버튼을 눌러 설정한 스케줄을 등록시킨다.</p> <p>4.(S) : 저장된 스케줄을 등록한다.</p>
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	<p>E1. 스케줄 설정 화면에서 기능 전환을 하는 경우 설정하던 스케줄 값을 지우고 등록하지 않는다.</p> <p>E2. 값 증가 버튼을 눌렀을 때 현재 수정할 항목의 값이 일정종류라면 일정종류를 변경 한다. (일정 종류는 PROM/MEET/BRTH/ETC)</p>

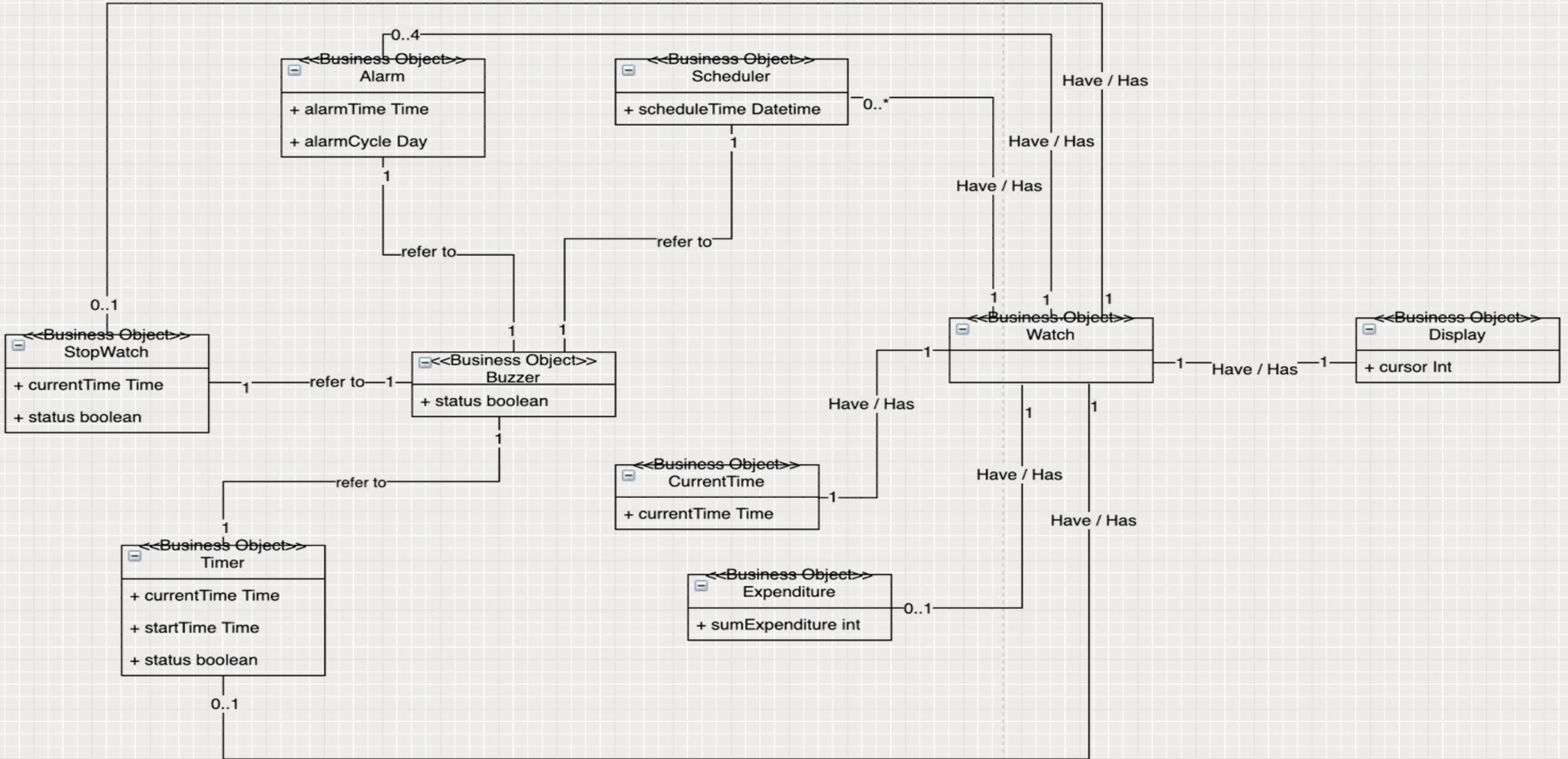
# Define Essential Use Cases

Use Case	Append Expenditure
<b>Actor</b>	User
<b>Purpose</b>	지출액을 추가한다.
<b>Overview</b>	첫 커서에서 증가 버튼을 눌러 +,-를 설정한다. 커서 이동 버튼을 눌러 설정한 숫자들을 왼쪽으로 shift 시킨다. 증가 버튼을 눌러 현재 커서의 숫자를 설정한다. 지출액 추가 버튼을 눌러 설정한 돈의 값을 월 지출액에 더해 저장한다.
<b>Type</b>	Evident
<b>Cross Reference</b>	
<b>Pre-Requisites</b>	각 버튼의 기능이 Append Expenditure에 사용되는 기능들로 설정되어야 한다.
<b>Typical Courses of Events</b> <b>(A) : Actors (S) : System</b>	1.(A) : 첫 커서에서 증가 버튼을 눌러 +,-를 설정한다. 2.(A) : 커서 이동 버튼을 눌러 금액 설정 커서로 이동한다. 2.(A) : 커서 이동 버튼을 눌러 설정한 숫자들을 왼쪽으로 shift 시킨다. 3.(A) : 증가 버튼을 눌러 현재 커서의 숫자를 증가시킨다. 4.(A) : 지출액 추가 버튼을 눌러 설정한 돈의 값을 월 지출액에 추가해 저장한다. 5.(S) : 월 지출액에 추가해 저장한다.
<b>Alternative Courses of Events</b>	N/A
<b>Exceptional Courses of Events</b>	E1. 기능 전환 버튼을 눌렀을 때에는 현재 설정하던 월 지출액을 저장하지 않는다.

# Define Essential Use Cases

Use Case	Show Expenditure
<b>Actor</b>	User
<b>Purpose</b>	월 지출액을 보여준다.
<b>Overview</b>	가장 큰 단위의 숫자부터 왼쪽 정렬하여 보여준다. 커서 이동 버튼을 눌러 오른쪽으로 이동해 7자리 이상을 볼 수 있다. 마지막 자리까지 왔을 때, WON을 출력해 마지막 숫자임을 알린다.
<b>Type</b>	Evident
<b>Cross Reference</b>	
<b>Pre-Requisites</b>	각 버튼의 기능이 Show Expenditure에 사용되는 기능들로 설정되어 있어야 한다.
<b>Typical Courses of Events</b> <b>(A) : Actors (S) : System</b>	1.(S) : 가장 큰 단위의 숫자부터 왼쪽 정렬하여 보여준다. 2.(A) : 커서 이동 버튼을 눌러 왼쪽으로 이동하며 월 지출액을 본다. 3.(S) : 커서 이동시 마지막 자리라면 WON을 출력한다.
<b>Alternative Courses of Events</b>	A1. 커서 이동이 중간에서 다른 기능으로 전환했다가 다시 이 기능으로 돌아오면 가장 큰 단위의 숫자부터 왼쪽 정렬하여 보여준다.
<b>Exceptional Courses of Events</b>	E1. 마지막 자리에서 커서 이동을 할 경우 다시 가장 큰 단위의 자리 숫자부터 왼쪽 정렬하여 보여준다.

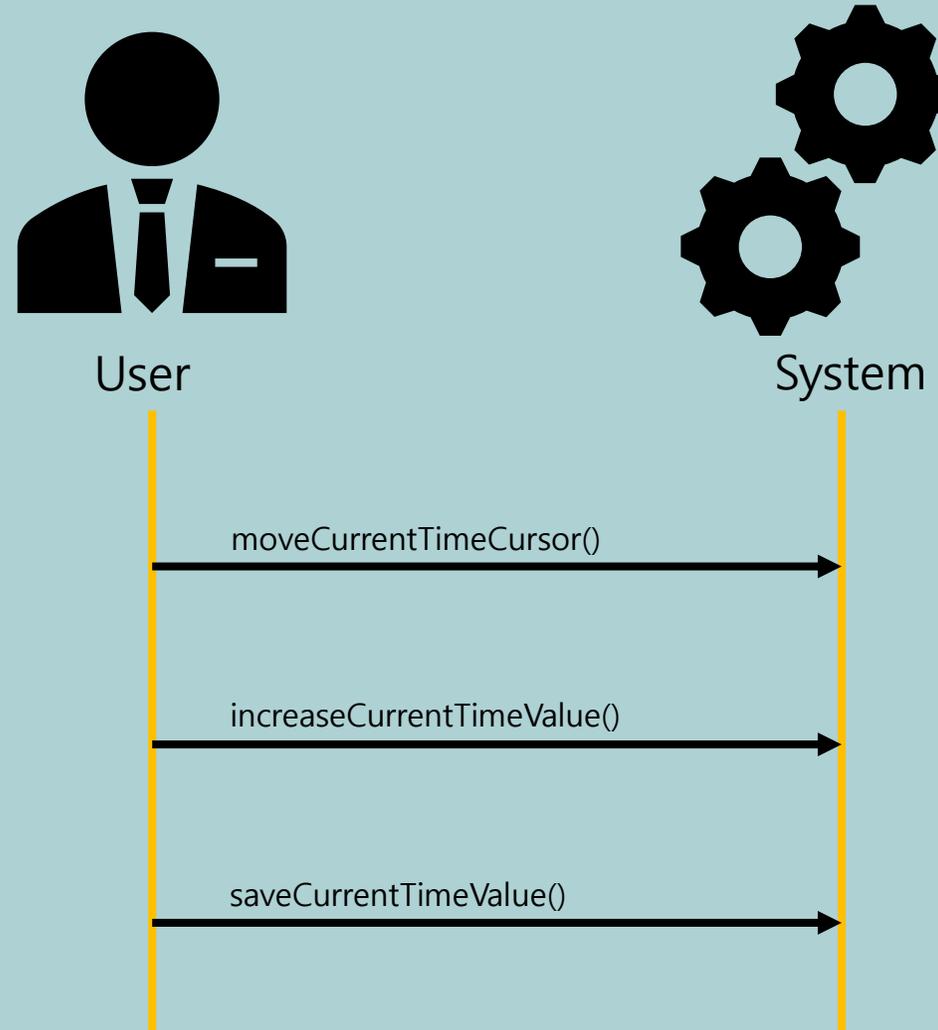
# Domain Model



# System Sequence Diagrams

## Use Case Change current time

1. Move cursor point
2. Increase value (if seconds -> initialize, else increase)
3. Save value
4. Show changed current time



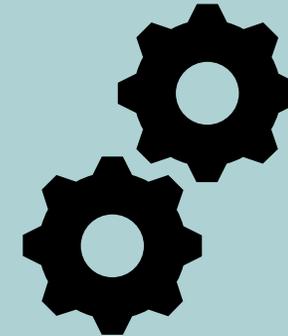
# System Sequence Diagrams

Use Case  
Set Timer

1. Move cursor point
2. Increase value
3. Save value
4. Show entered timer time



User



System

moveTimerCursor()

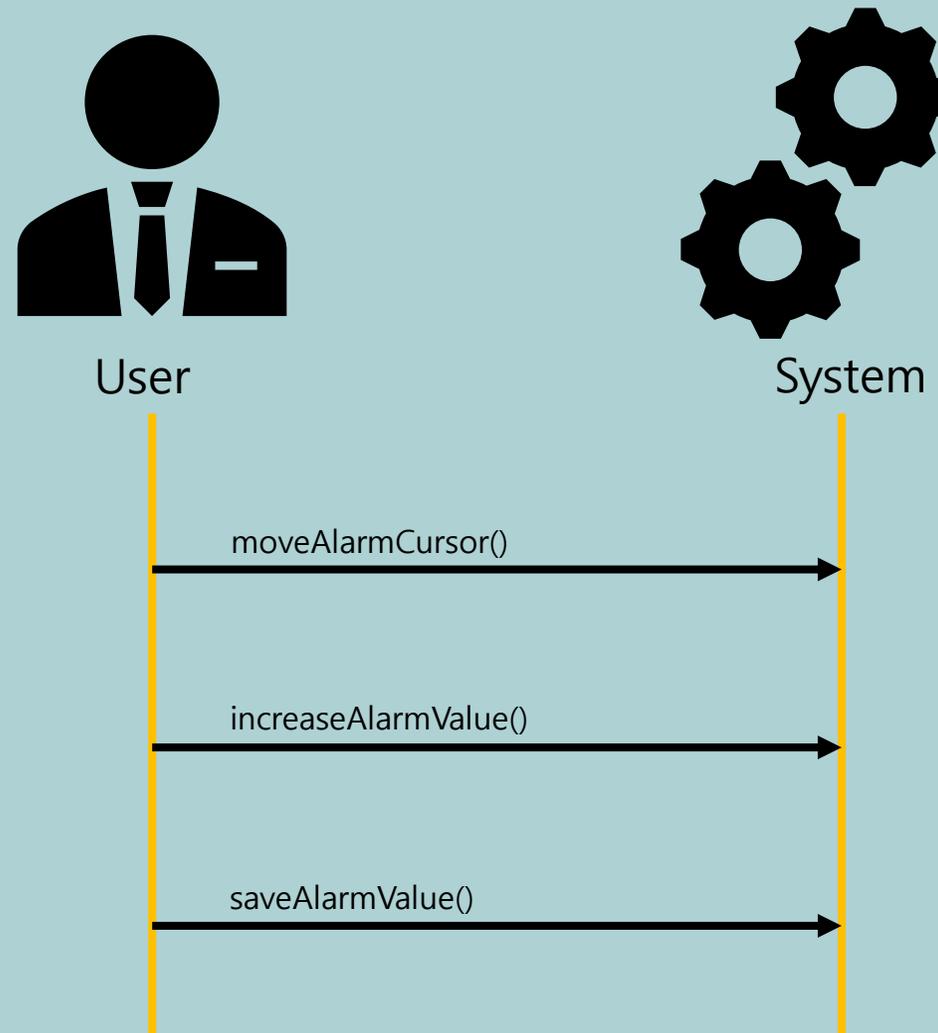
increaseTimerValue()

saveTimerValue()

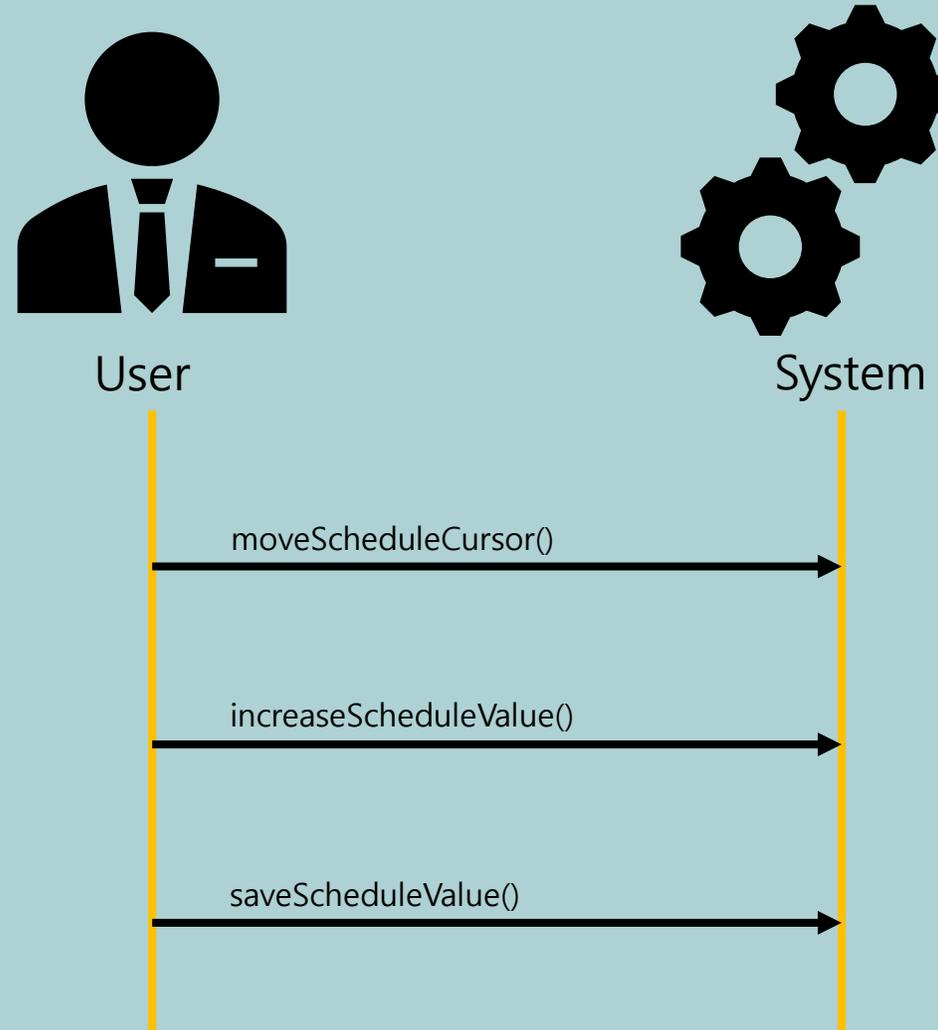
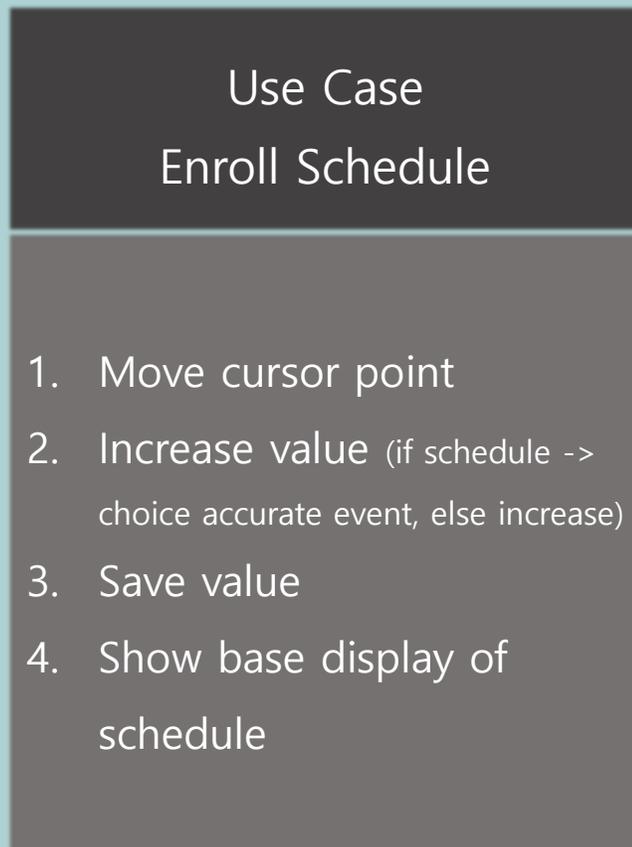
# System Sequence Diagrams

## Use Case Set Alarm

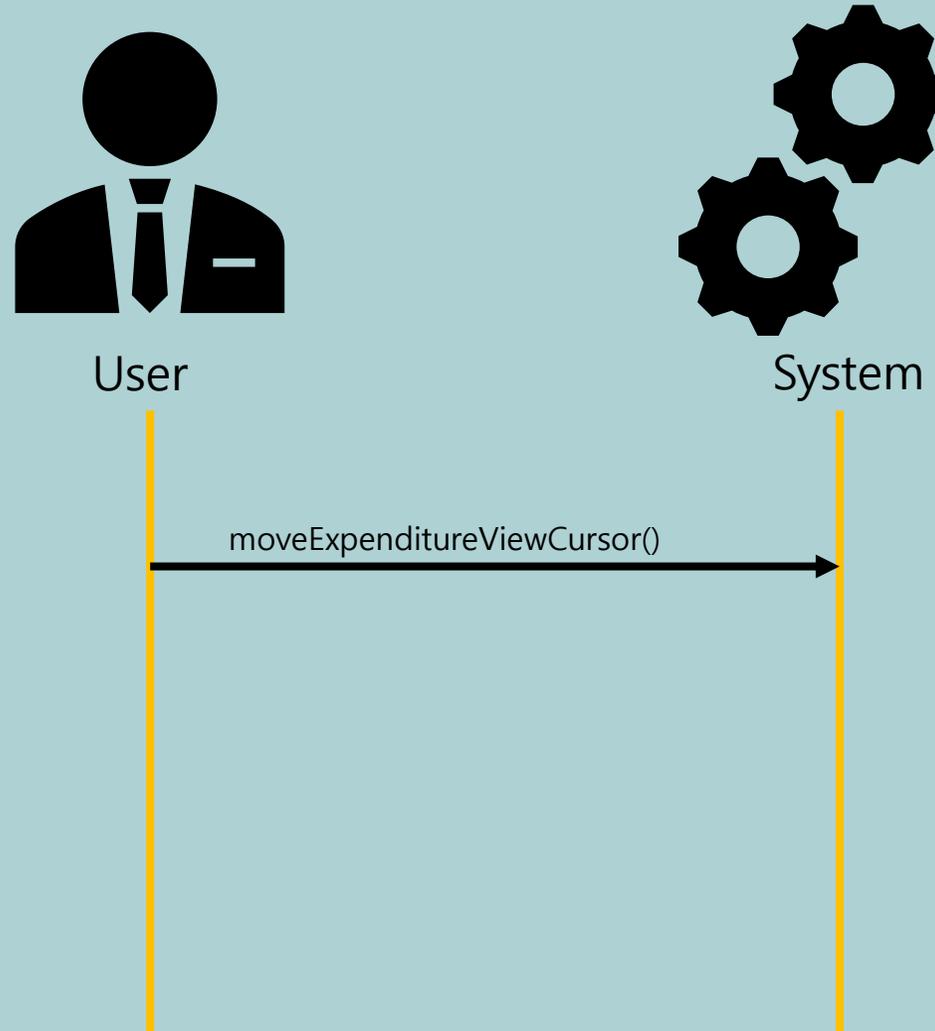
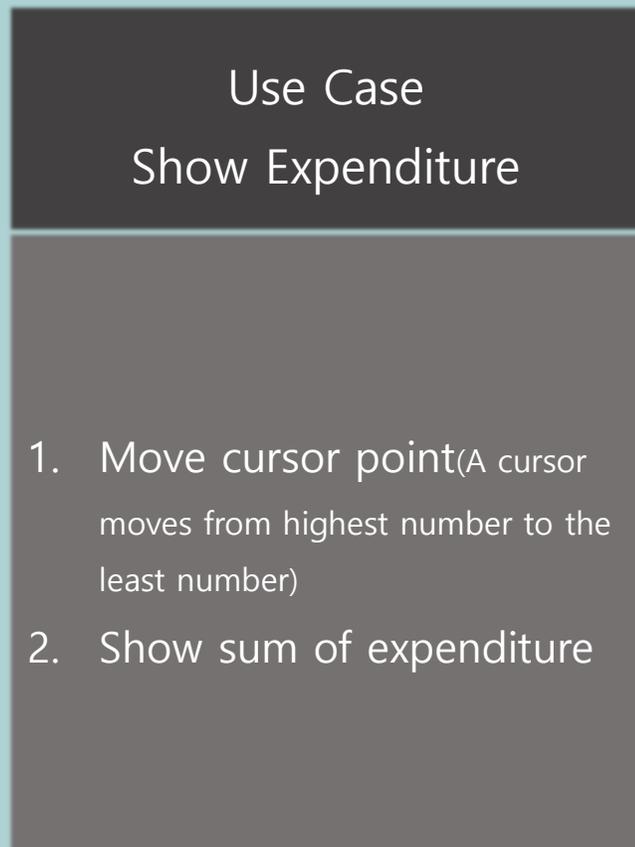
1. Move cursor point
2. Increase value(if ON/OFF -> two choices, else if day of the week select each ON/OFF, else increase)
3. Save value
4. Show entered alarm time



# System Sequence Diagrams



# System Sequence Diagrams



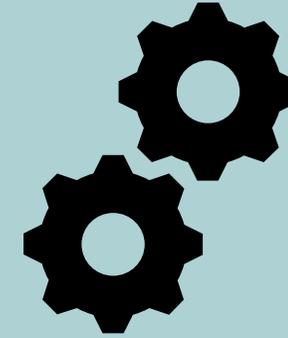
# System Sequence Diagrams

## Use Case Append Expenditure

1. Move cursor point(added +/- compared with show expenditure)
2. Increase value(If +/- -> two choices, else increase)
3. Save value
4. Show sum of expenditure



User



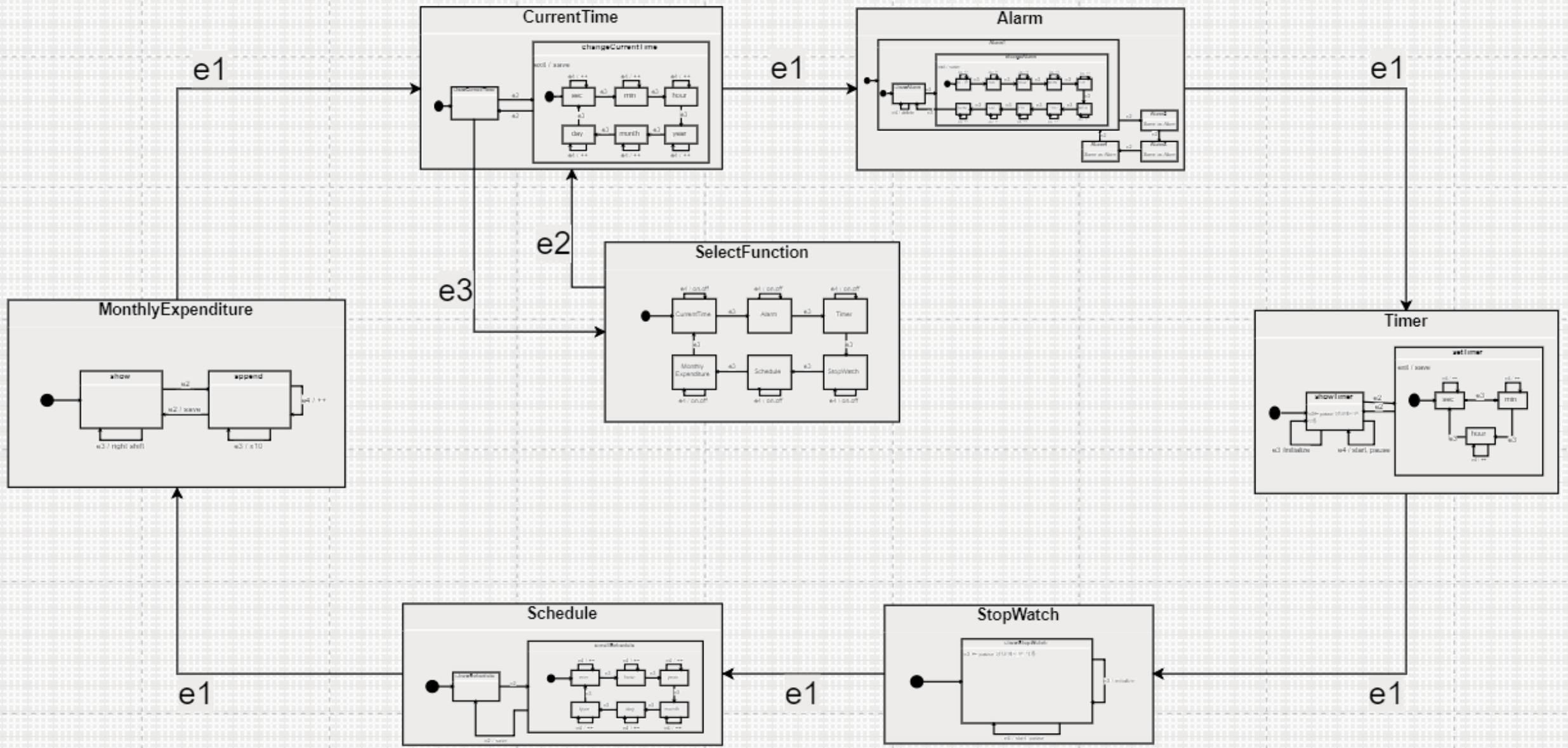
System

moveExpenditureCursor()

increaseExpenditureValue()

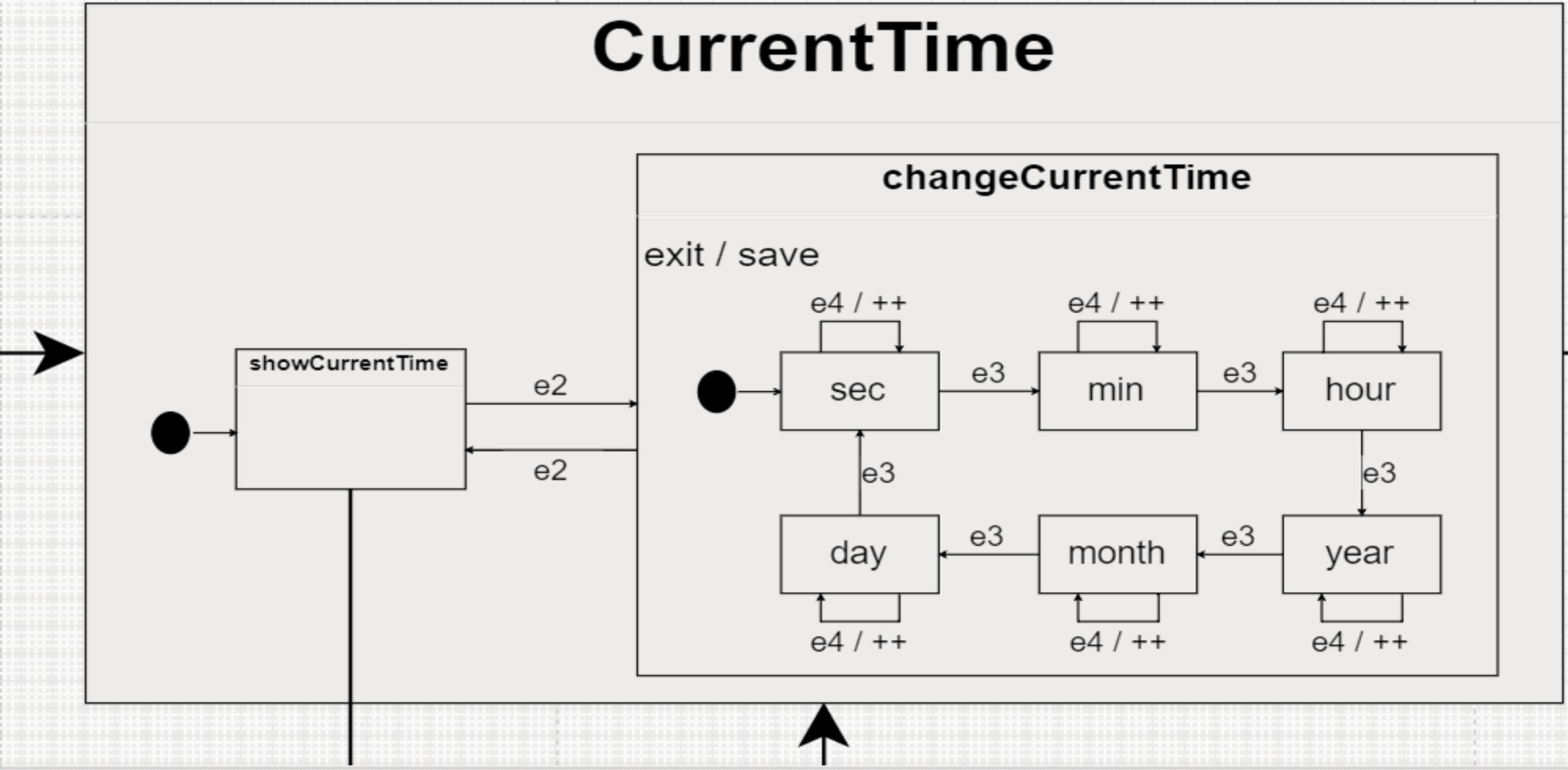
saveExpenditureValue()

# State Diagrams



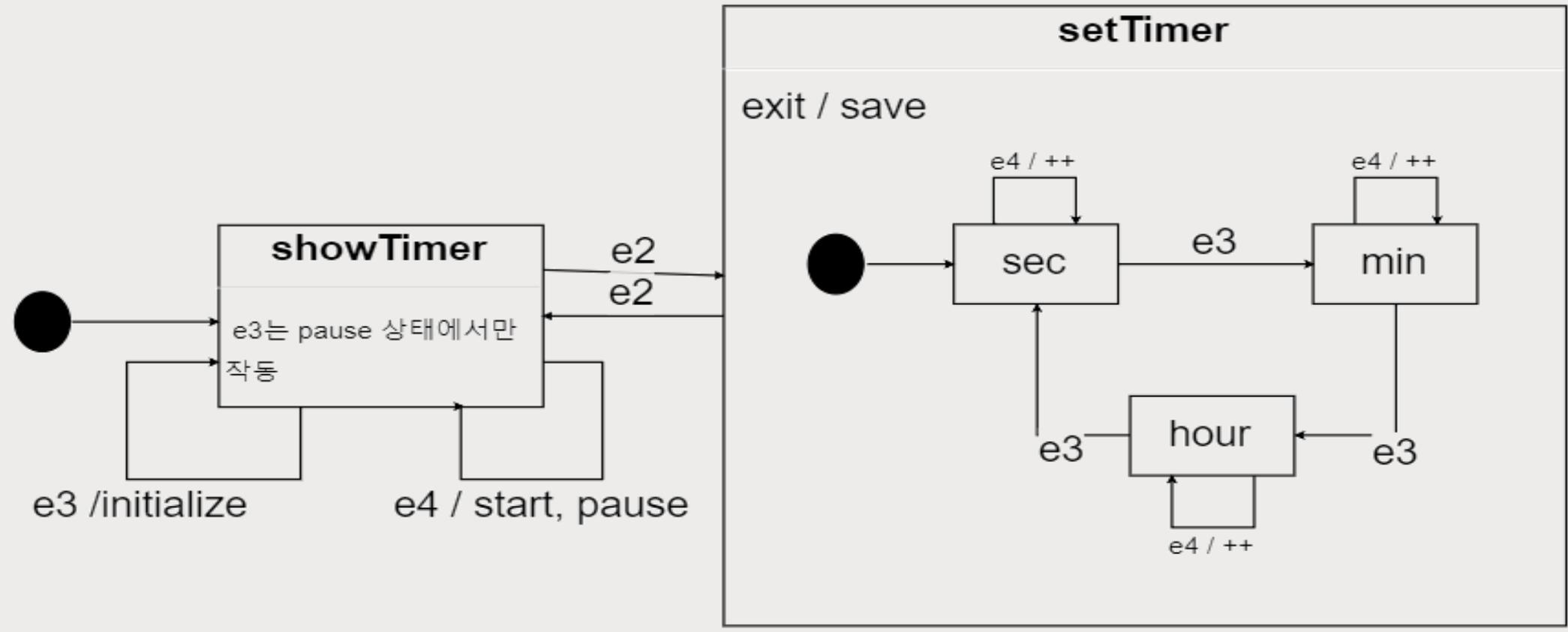
# State Diagrams

## CurrentTime



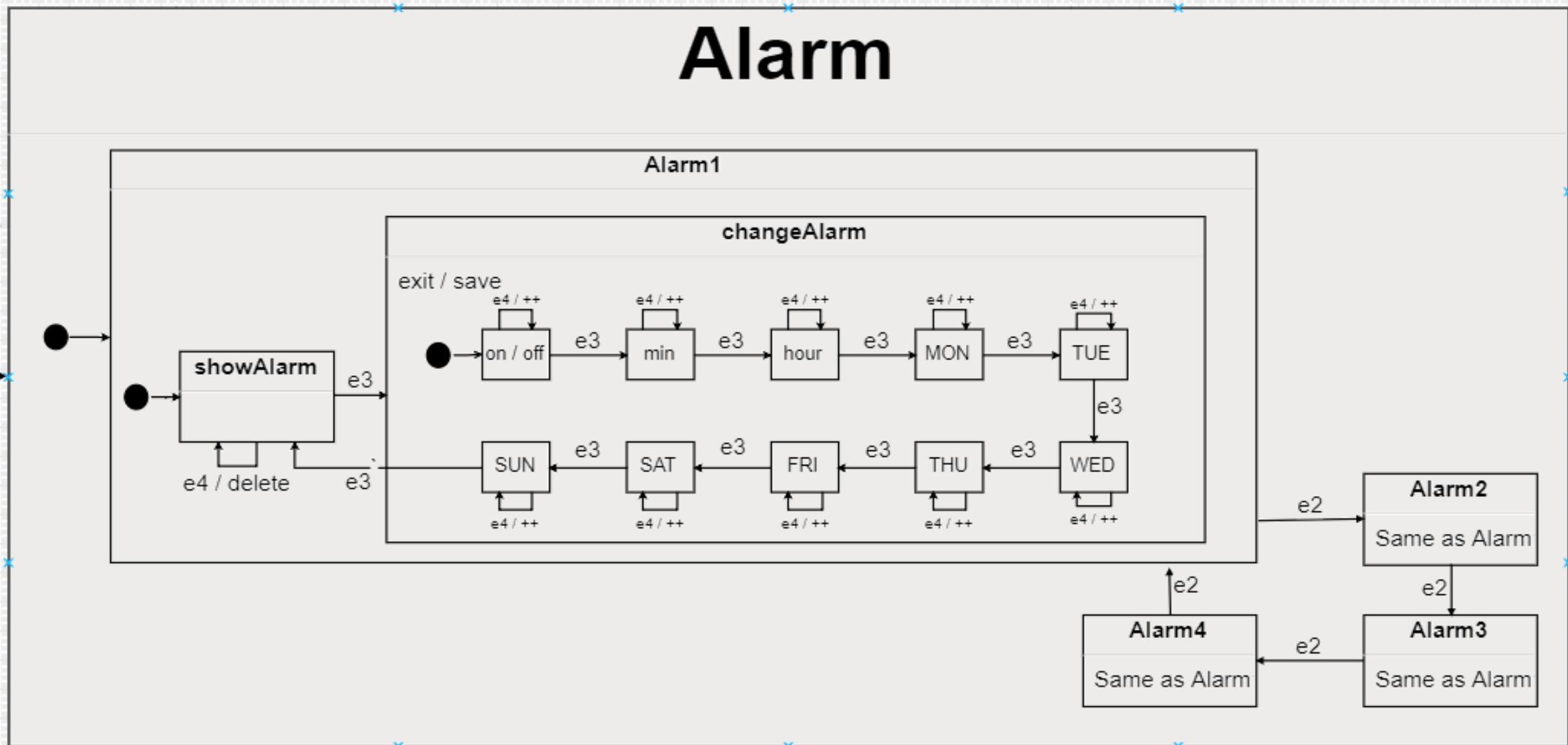
# State Diagrams

## Timer



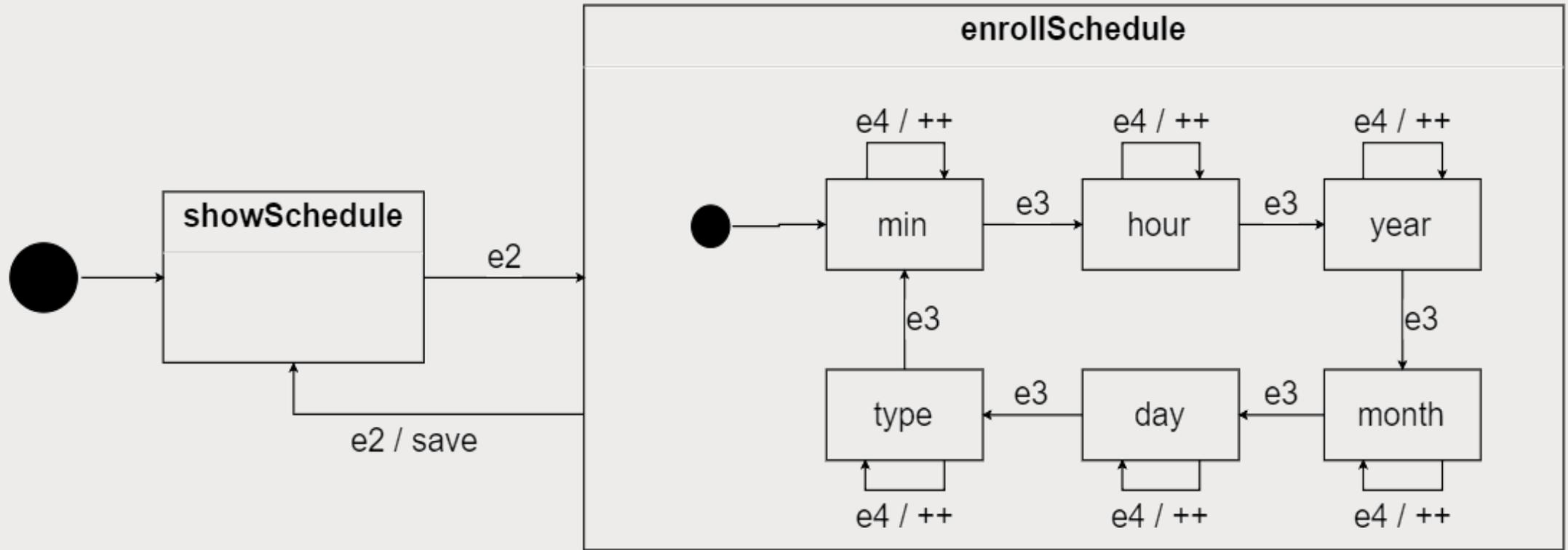
# State Diagrams

## Alarm



# State Diagrams

## Schedule



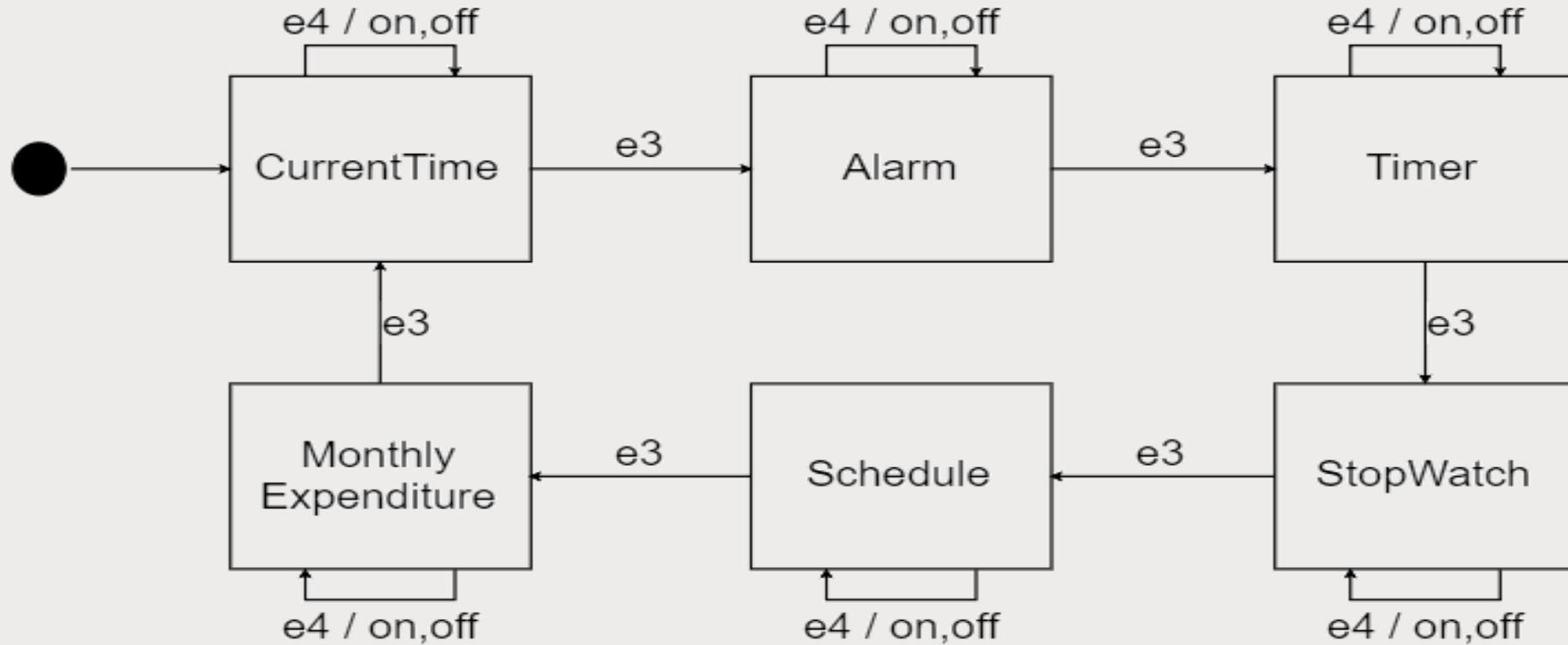
# State Diagrams

## MonthlyExpenditure

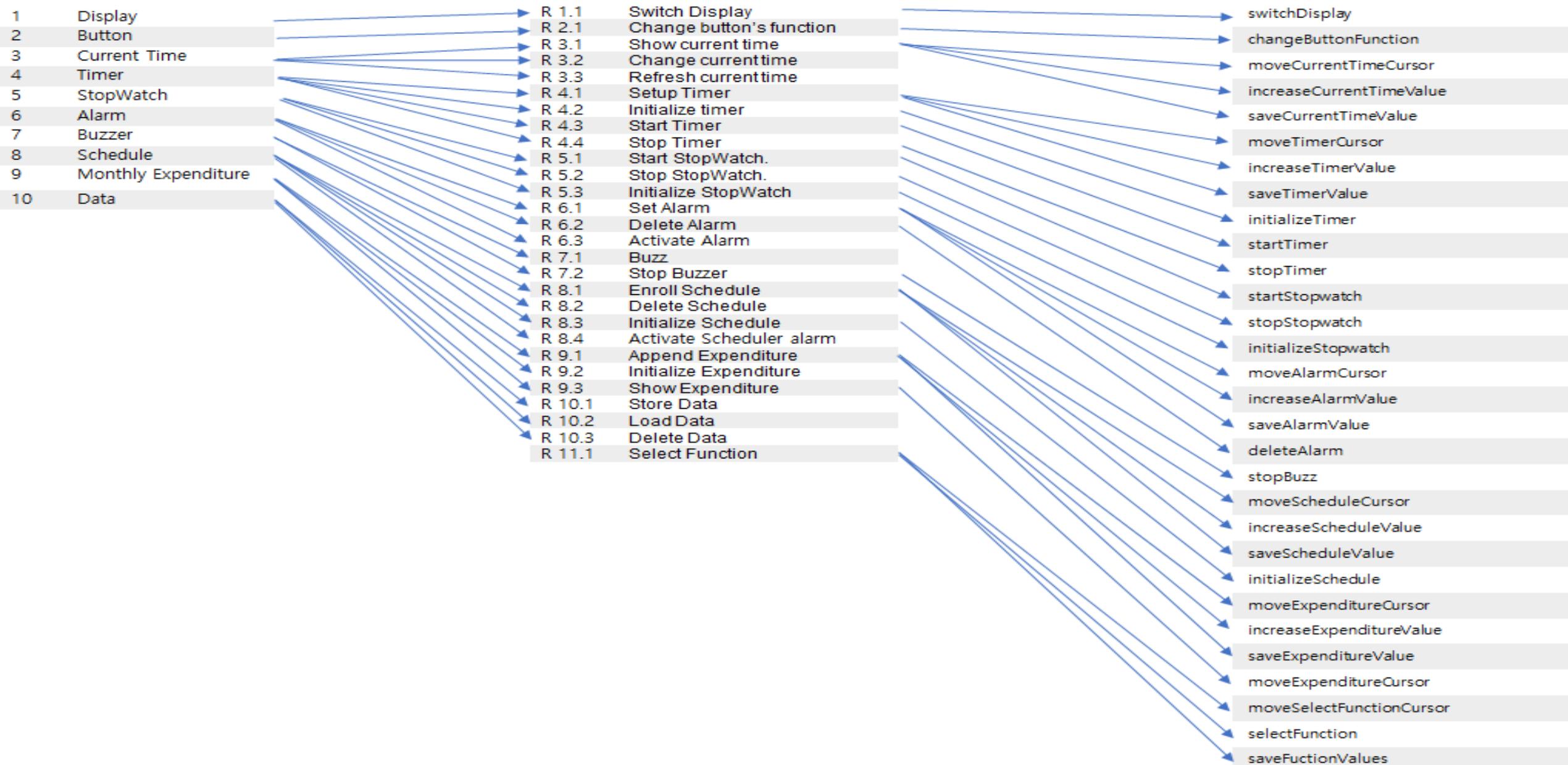


# State Diagrams

## SelectFunction



# Traceability Analyze



***Q&A***

***Thank you***