

# SafeHome Team 4

Analysis Model

Version 1.2

Printed by Team4

Team 4 :

20000330 Sangyoung, Lee

20040033 Sejoong, Kwon

20050426 Jieun, Lee

20070485 Kanghee, Won

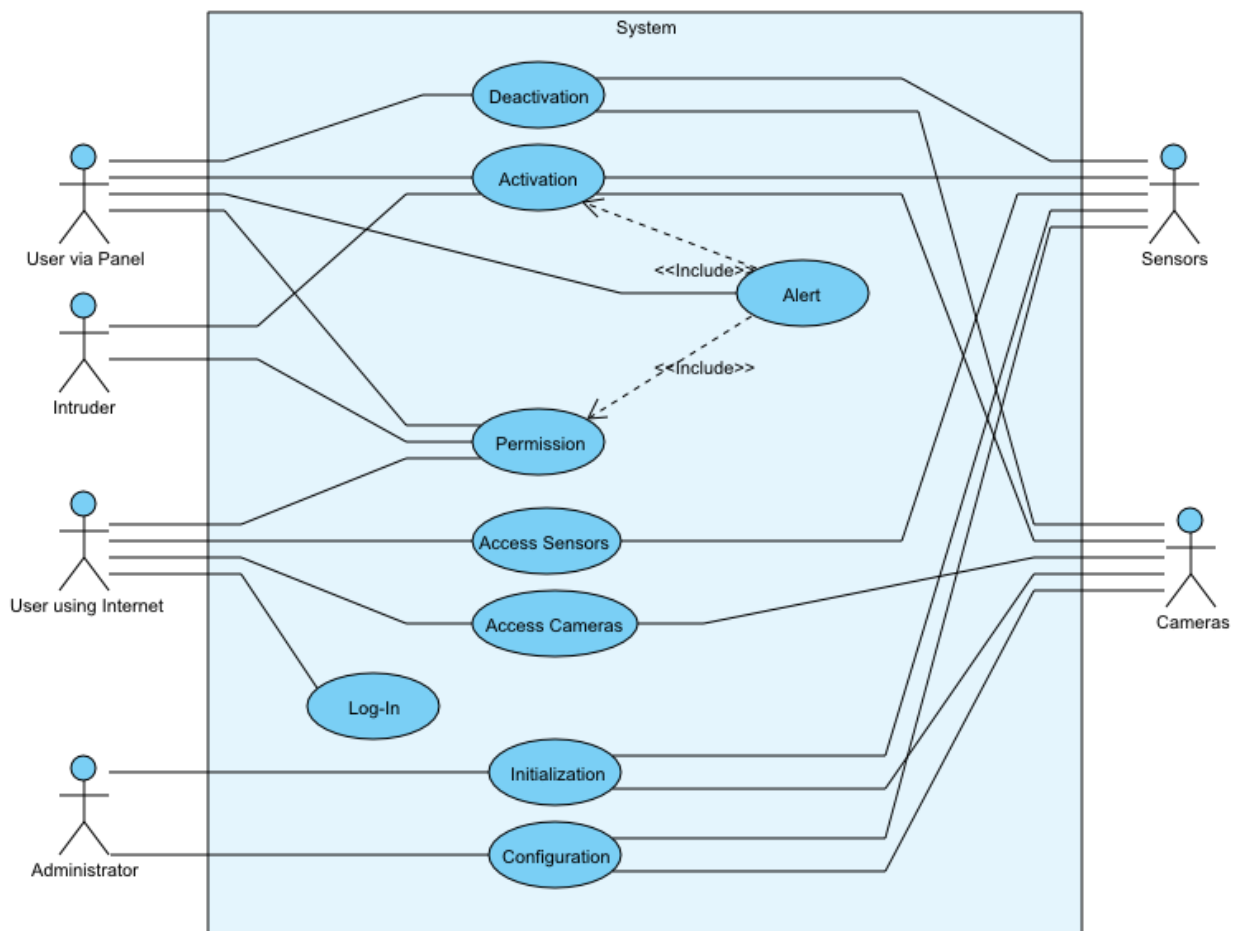
# Table of Contents

<b>Table of Contents .....</b>	<b>2</b>
<b>1. Main Use Case Diagram .....</b>	<b>5</b>
<b>2. Sub Use Case Diagram .....</b>	<b>6</b>
2.1 Initialization.....	6
2.2 Activation .....	7
2.3 Deactivation.....	8
2.4 Permission .....	9
2.5 Access Sensors .....	10
2.6 Access Cameras.....	11
2.7 Alert.....	12
2.8 Configuration.....	13
<b>3. Use Case Description .....</b>	<b>14</b>
3.1 Initialization.....	14
3.1.1 Revision History .....	14
3.1.2 Main Success Scenario .....	15
3.1.3 Extensions .....	15
3.1.4 Frequency of use .....	15
3.1.5 Priority .....	15
3.1.6 Traceability.....	15
3.2 Activation .....	17
3.2.1 Revision History .....	17
3.2.2 Main Success Scenario .....	18
3.2.3 Extensions .....	18
3.2.4 Frequency of use .....	19
3.2.5 Priority.....	19
3.2.6 Traceability.....	19
3.3 Deactivation.....	21
3.3.1 Revision History .....	21
3.3.2 Main Success Scenario .....	22
3.3.3 Extensions .....	22
3.3.4 Frequency of use .....	23
3.3.5 Priority.....	23
3.3.6 Traceability.....	23
3.4 Permission .....	25
3.4.1 Revision History .....	25
3.4.2 Main Success Scenario .....	26
3.4.3 Extensions .....	26
3.4.4 Frequency of use .....	26
3.4.5 Priority .....	26
3.4.6 Traceability.....	27
3.5 Access Sensors .....	28
3.5.1 Revision History .....	28
3.5.2 Main Success Scenario .....	28
3.5.3 Extensions .....	29
3.5.4 Frequency of use .....	29

3.5.5 Priority .....	29
3.5.6 Traceability .....	29
3.6 Access Cameras .....	31
3.6.1 Revision History .....	31
3.6.2 Main Success Scenario .....	32
3.6.3 Extensions .....	32
3.6.4 Frequency of use .....	32
3.6.5 Priority .....	32
3.6.6 Traceability .....	33
3.7 Alert .....	35
3.7.1 Revision History .....	35
3.7.2 Main Success Scenario .....	36
3.7.3 Extensions .....	36
3.7.4 Frequency of use .....	36
3.7.5 Priority .....	36
3.7.6 Traceability .....	36
3.8 Configuration .....	38
3.8.1 Revision History .....	38
3.8.2 Main Success Scenario .....	39
3.8.3 Extensions .....	39
3.8.4 Frequency of use .....	40
3.8.5 Priority .....	40
3.8.6 Traceability .....	40
3.9 Log-in .....	41
3.9.1 Revision History .....	41
3.9.2 Main Success Scenario .....	41
3.9.3 Extensions .....	42
3.9.4 Frequency of use .....	42
3.9.5 Priority .....	42
3.9.6 Traceability .....	42
<b>4. Sequence Diagram .....</b>	<b>43</b>
4.1 Initialization .....	43
4.2 Activation .....	45
4.3 Deactivation .....	47
4.4 Permission .....	49
4.5 Access Sensors .....	50
4.6 Access Cameras .....	51
4.7 Alert .....	53
4.8 Configuration .....	55
<b>5. Swimlane Diagram .....</b>	<b>56</b>
5.1 Initialization .....	56
5.1.1 Initialization functions in the camera module .....	57
5.1.1 Initialization functions in the sensor module .....	58
5.1.1 Initialization functions in the phonecall module .....	59
5.2 Activation .....	60
5.3 Deactivation .....	62
5.4 Permission .....	63
5.5 Access Sensors .....	64
5.6 Access Cameras .....	66
5.7 Alert .....	68

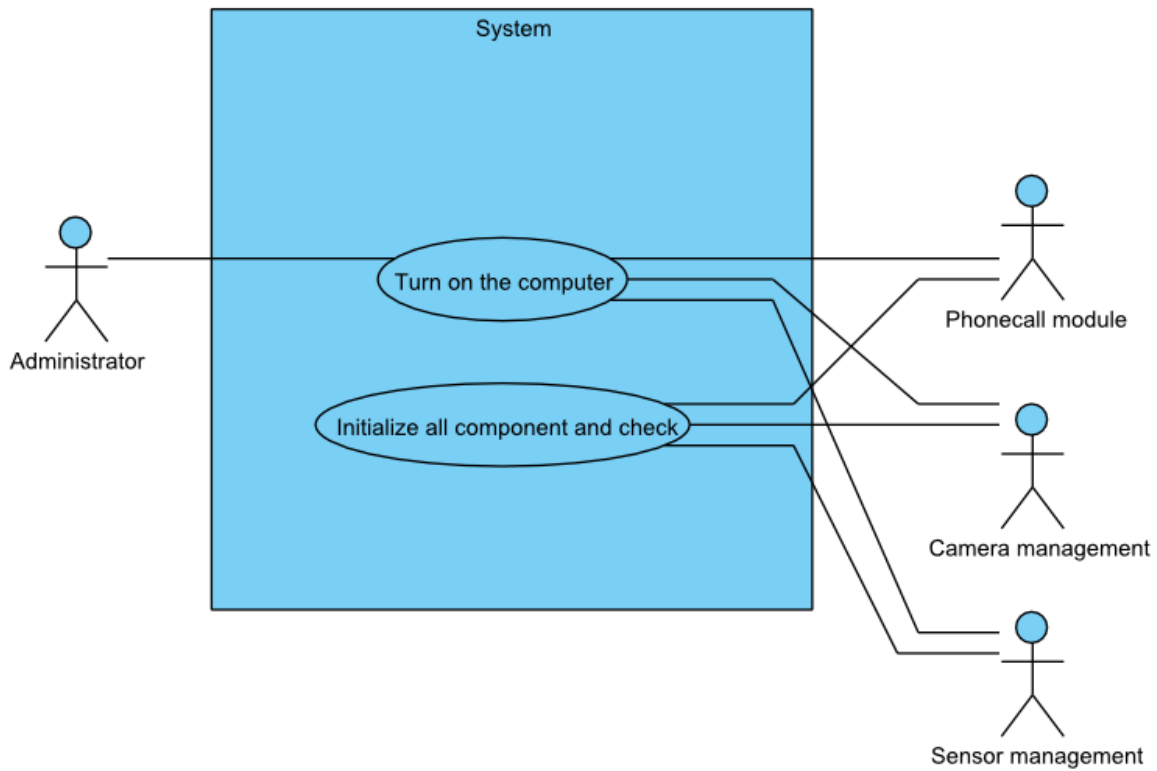
5.8 Configuration.....	69
5.9 Log-in.....	73
<b>Appendix A: Doors Links.....</b>	<b>74</b>
<b>Appendix B: Requirement specification .....</b>	

# 1. Main Use Case Diagram

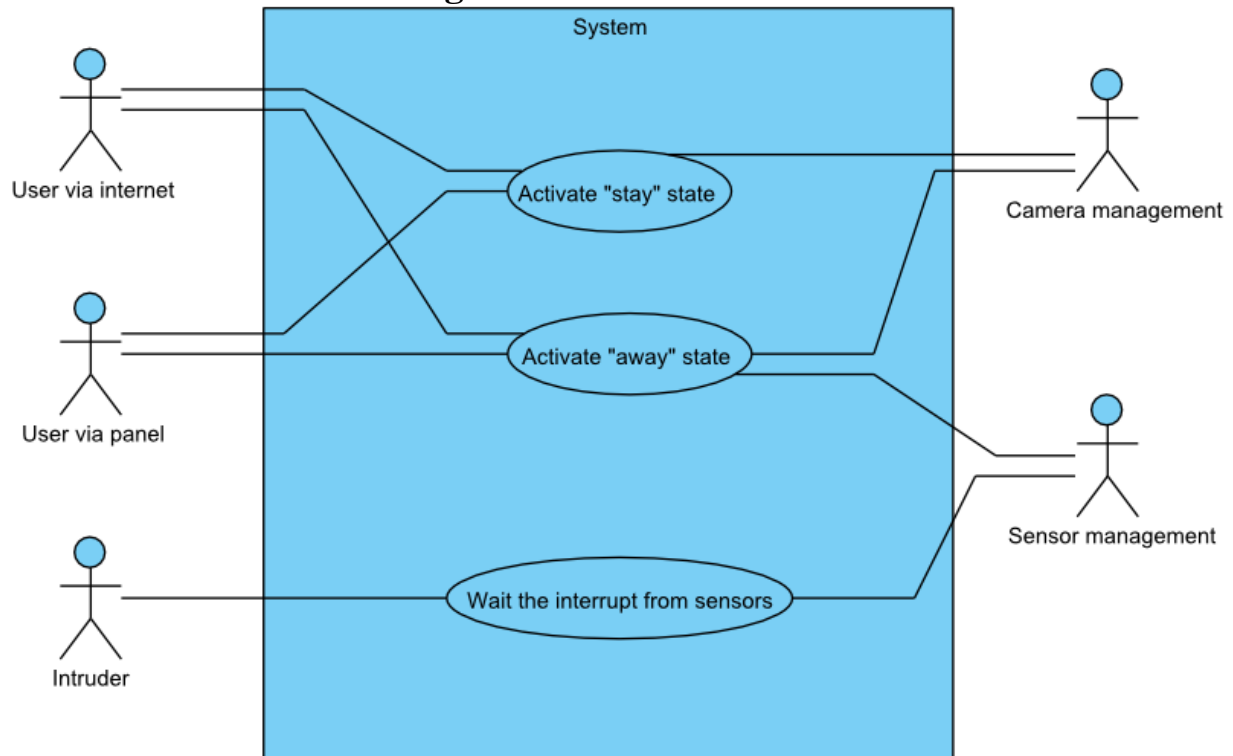


## 2. Sub Use Case Diagram

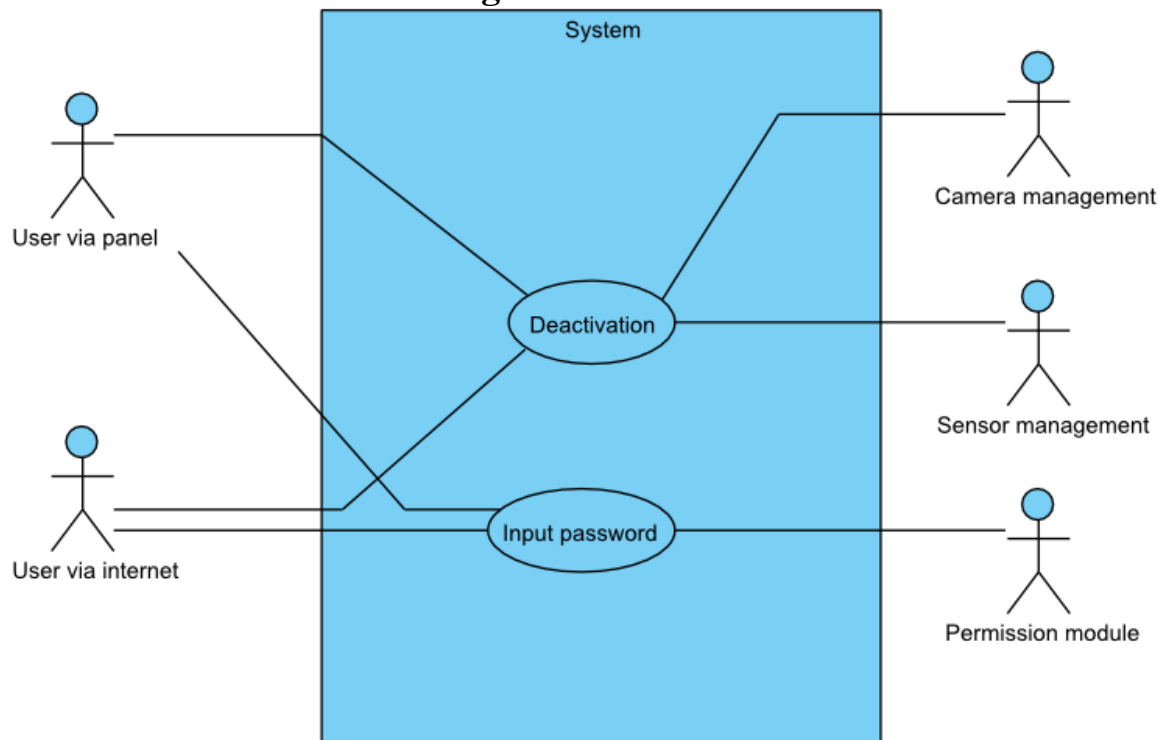
### 2.1 Initialization Use Case Diagram



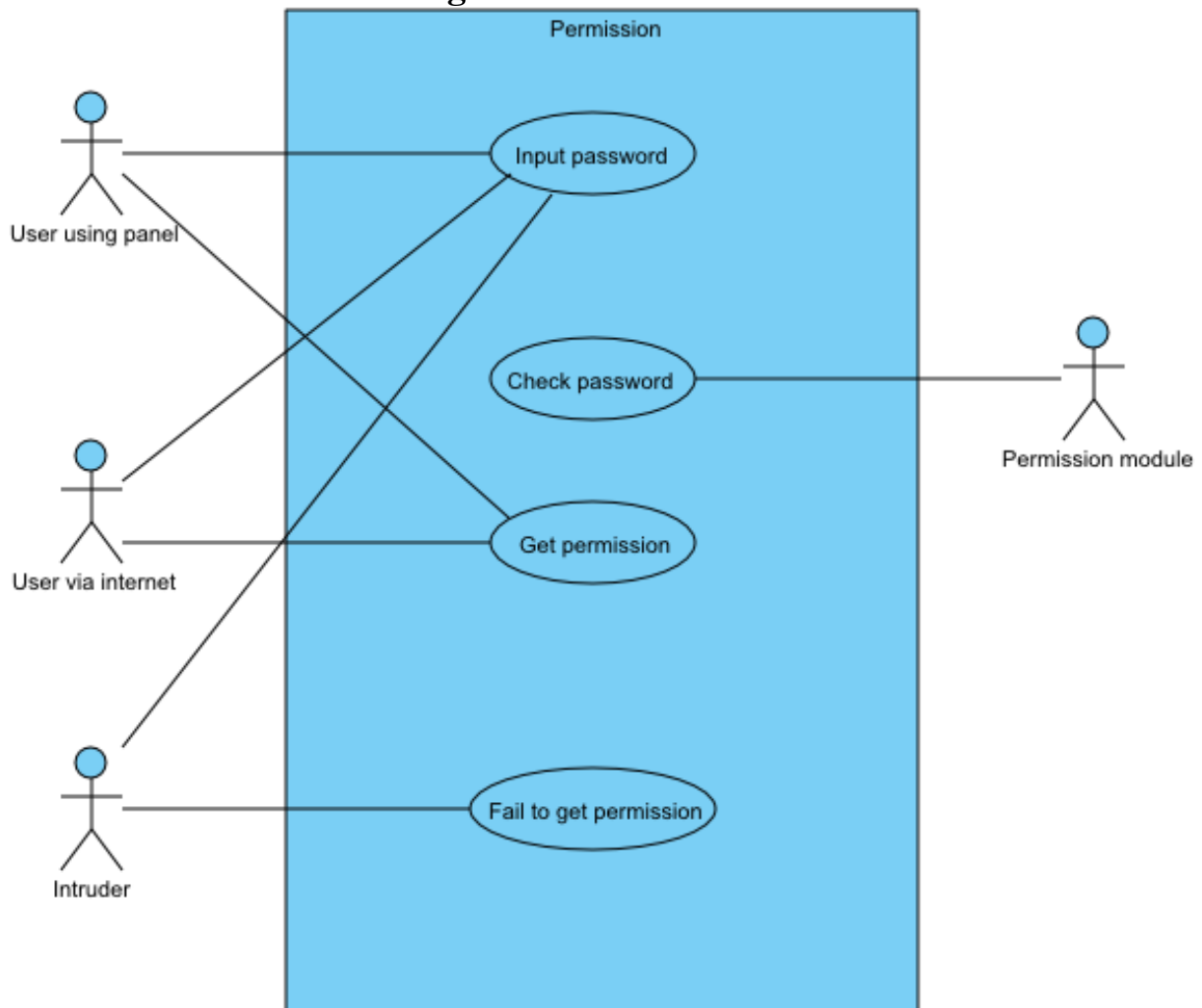
## 2.2 Activation Use Case Diagram



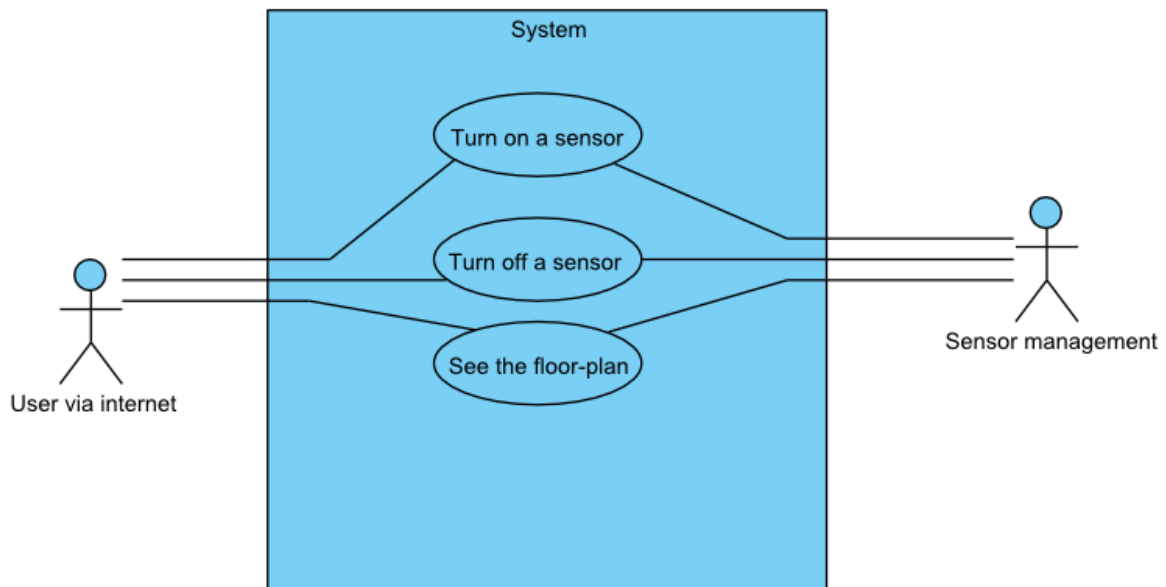
## 2.3 Deactivation Use Case Diagram



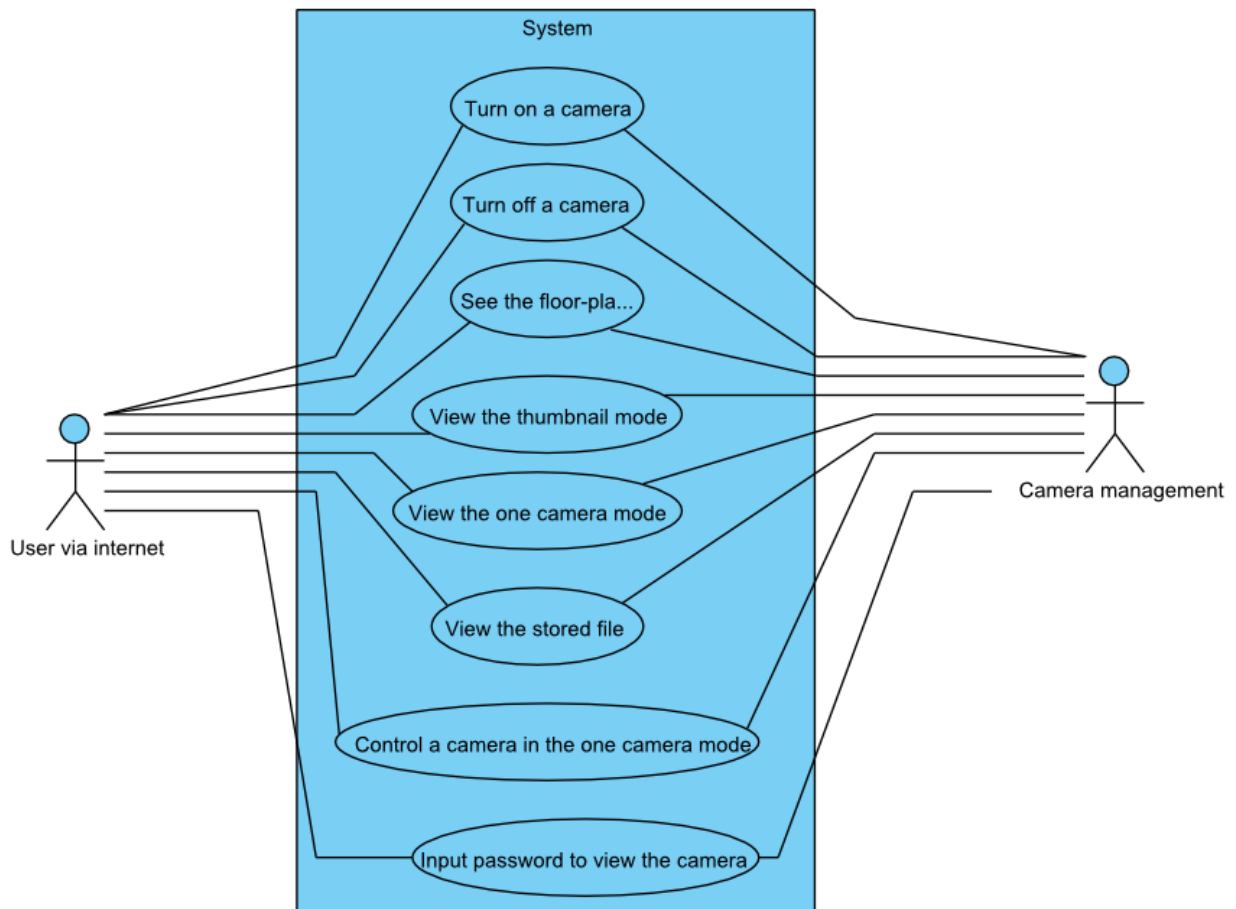
## 2.4 Permission Use Case Diagram



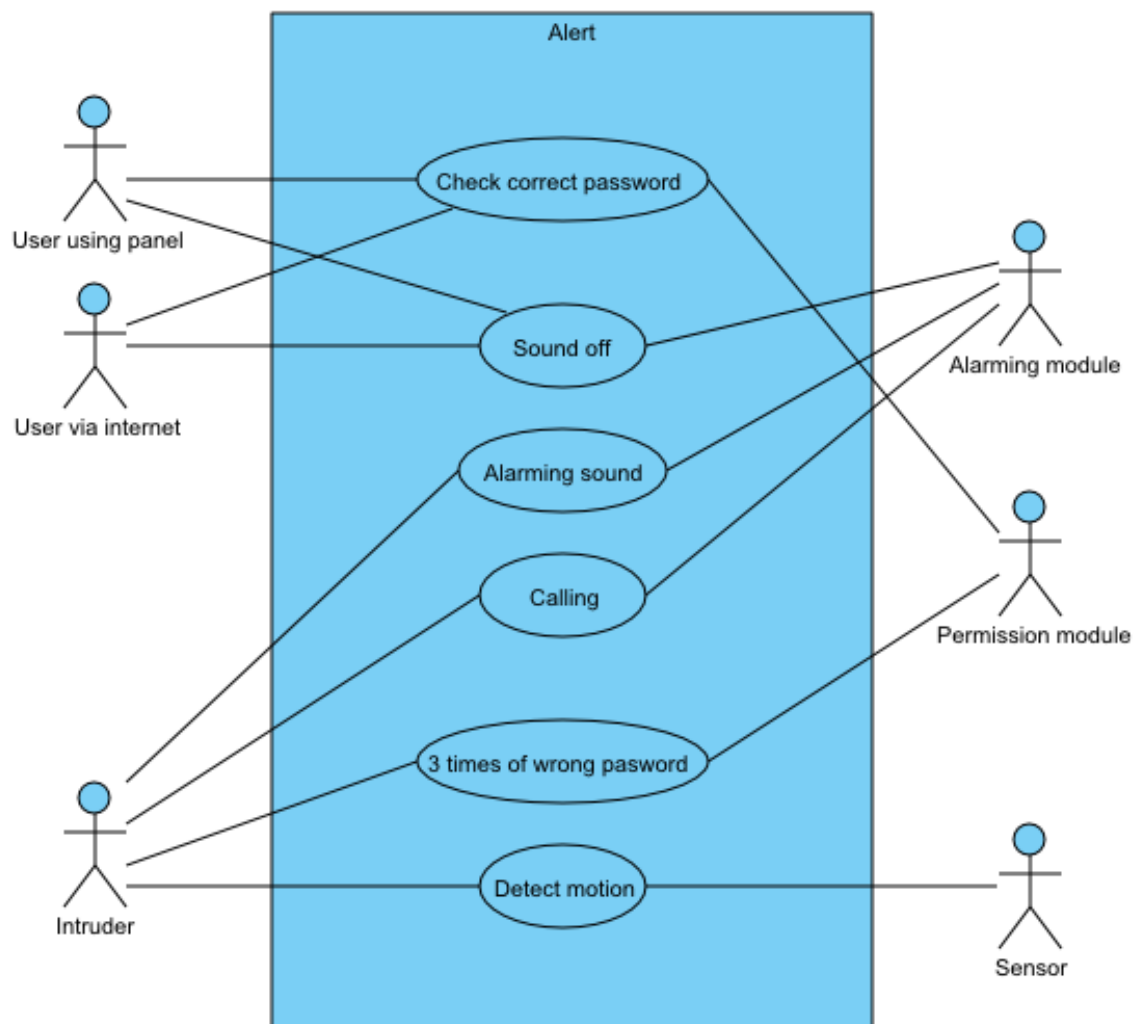
## 2.5 Access Sensors Use Case Diagram



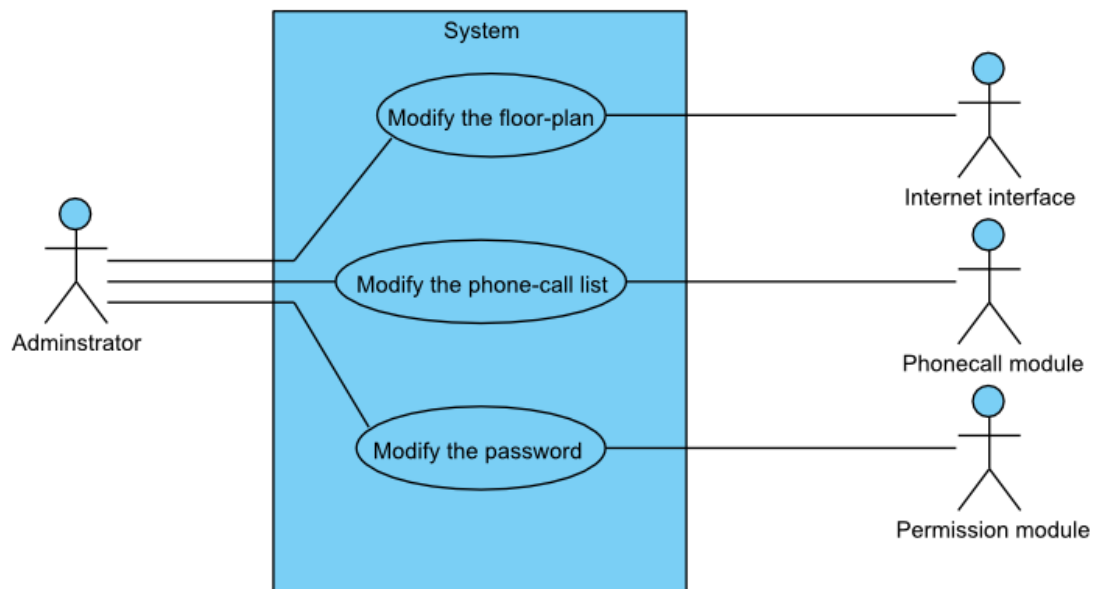
## 2.6 Access Cameras Use Case Diagram



## 2.7 Alert Use Case Diagram



## 2.8 Configuration Use Case Diagram



## 3. Use Case Description

### 3.1 Initialization

#### 3.1.1 Revision History

Date	Author	Description of change
08/04/23	Jieun, Lee, Sangyoung Lee	Started
08/04/25	Sejoong, Kwon	Modified
08/04/27	Jieun, Lee	Modified Extensions, Traceability
08/04/28	Sejoong, Kwon	Modified Main Success Scenario

#### 1. Use Case: Initialization

#### 2. Description

When the Safehome system starts to power on, system must check the camera management module, sensor management module, and Phonecall Module are fine. Only administrator can turn on or off the Safehome system via Internet by logging in. Initialization module only concern about turning on the Safehome system. And if some of those three modules are malfunction, Initialization order to display about malfunctioning module information, and cancel the starting Safehome system.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. Administrator

#### 5. Supporting Actors

1. Initialization module
2. Sensor management module
3. Camera management module
4. Phonecall module

#### 6. Pre-Conditions

1. Safehome system including panel, wireless box must be installed

#### 7. Post Conditions

##### Success end condition

1. Safehome system is turned on.

##### Failure end condition

1. Safehome system fails to be turned on.

## 8.Trigger

1. Administrator turns on the Safehome system.

### 3.1.2 Main Success Scenario

1. Administrator turns on the Safehome system.
2. Initialization module checks sensor management module, camera management module, phonecall module.
3. Safehome system starts.

Initialize all the sensors. Scan wireless components, and compare this result with the information stored in this module

4. 1. If the result from scanning is same as the stored information, return success.
5. 2. If the result from scanning is not same as the stored information, return fail.

### 3.1.3 Extensions

- 2a. In step 2, any of above three modules can be malfunctioning.
  1. Initialization module request to module to display the malfunctioning module information.
  2. Initialization module stop to starting Safehome system.
- 2b. In step 2, Initialization module do followings
  1. Initialization module checks the stored values about sensors are same as the values in the sensor management module.
  2. Initialization module checks the stored values about cameras are same as the values in the camera management module.
  3. Initialization module checks the stored phone number list and phone line.

### 3.1.4 Frequency of use

Very low

### 3.1.5 Priority

Essential, must be implemented.

### 3.1.6 Traceability

Use case description about Initialization is closely related to Initialization module in Requirement specification.

Requirement Specification	Use Case Description
3.1 Initialization module	
3.1.1 Descriptions	
3.1.2 Functionality & Scenario	
3.1.2.1 call the initialize function in the sensor management module	3.1.2 Main Success Scenario 2

3.1.2.2 call the initialize function in the camera management module	3.1.2 Main Success Scenario 2
3.1.2.3 call the initialize function in the phonecall module	3.1.2 Main Success Scenario 2
3.1.3 .3 Frequency of use Very low	3.1.4 Frequency of use
3.1.4 Special conditions None	
3.1.5 Constraints None	

## 3.2 Activation

### 3.2.1 Revision History

Date	Author	Description of change
08/04/09	Jieun, Lee	Started
08/04/15	Sangyoung, Lee	Modified Extension 5a
08/04/23	Jieun, Lee	Modified Primary Actor
08/04/26	Jieun, Lee	Modified Trceability
08/04/26	Jieun, Lee	Modified Main Success Scenario, Extensions

#### 1. Use Case: Activation

#### 2. Description

SafeHome system has two types of sensors; First-type of sensors sticks to doors, and windows. And second-type of sensors is motion detector in the house. If all sensors are functioning, the system is in activated state. And if not, the system is deactivated. And these states need function which alters states.

Arming module is for switching the state between activated and deactivated state. There are two types of states in activated state. One is “stay” which permits only cameras work, the other is “away” which permits all components work additionally.

Activate function is accomplished by pushing “stay” or “away” button by user using panel. And user must choose one of those states when they activate the system.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. User using panel
2. User accessing via internet
3. Intruder

#### 5. Supporting Actors

1. Arming module

#### 6. Pre-Conditions

1. Activation only comes from deactivated state.
2. When user using panel push “stay” or “away” button, arming module sets to that state.

#### 7. Post Conditions

##### Success end condition

1. The system is changed from deactivated state to activated state.

##### Failure end condition

1. If user using panel pushed “stay” button and system can’t get any signal from camera management module, the panel displays information of malfunctioning cameras.
2. If user using panel pushed “away” button and system can’t get any signal from camera management module or sensor management module, the panel displays information of malfunctioning cameras or malfunctioning sensors.
3. In the case of 1,2, the states remains deactivated state.

## 8. Trigger

1. User using panel pushes “stay” or “away” button on panel.
2. User accessing via internet sets the system activated.

### 3.2.2 Main Success Scenario

- User using panel

1. User using panel pushes “stay” or “away” button.
2. If user using panel pushed “stay” button, only camera management module sends success signal to arming module.
3. If user using panel pushed “away” button, camera management module and sensor management module send success signal to arming module.
4. Arming module send signal to Sensor management module to wait for interrupt signal.
5. Armed LED on the panel turns on.
6. State changed from deactivated state to activated state.

- User accessing via internet

1. User accessing via internet selects “stay” or “away” activated state.
2. If user accessing via internet selects “stay” activated state, only camera management module sends success signal to arming module.
3. If user accessing via internet selects “away” activated state, camera management module and sensor management module send success signal to arming module.
4. Arming module send signal to Sensor management module to wait for interrupt signal.
5. Armed LED on the panel turns on.
6. State changed from deactivated state to activated state.

### 3.2.3 Extensions

- User using panel and User accessing via internet

23a. In step 2, 3, arming module, camera management module, sensor management module can be malfunctioning,

1. Resend the signal three times.
2. After 3 trials, Panel shows error messages about malfunctioning modules – which module is malfunctioning, what kind of malfunction it has

2a. In step 2, cameras can be malfunctioning.

1. The system can’t be “stay” activated.
2. Panel shows error messages about cameras’ malfunction – which camera is malfunctioning, what kind of malfunction it has.
3. User using panel should contact repairing service and user can get that service by its visiting.

- 3a. In step 3, cameras or sensors can be malfunctioning
1. The system can't be "away" activated.
  2. If cameras are malfunctioning panel shows error messages about cameras' malfunction – which cameras are malfunctioning, what kinds of malfunction it have.
  3. If sensors are malfunctioning panel shows error messages about sensors' malfunction – which sensors are malfunctioning, what kinds of malfunction it have.
  4. User using panel should contact repairing service and user can get that service by its visiting.
- 4a. In step4, when sensor detects motion in activate state
1. The moving object in the house is regarded as an intruder.

### 3.2.4 Frequency of use

Many times per day

### 3.2.5 Priority

Essential, must be implemented.

### 3.2.6 Traceability

Requirement specification about "Arming Module" is described in "Activation" and "Deactivation" use case diagram. Since Requirement specification 3.2.2.1 is Activation function, following traceability table will only show about 3.2.2.1. Requirement specification 3.2.2.2 is described in 2.3.6 Traceability.

Requirement Specification	Use Case Description
3.2 Arming Module	
3.2.1 Descriptions	
3.2.2 Functionality & Scenario	
3.2.2.1 Activate function	3.2.2 Main Success Scenario
3.2.2.1.1 Send signal - If the state that user wants is "stay", conduct only #2. - If the previous state is "stay", don't need to conduct #2.  1. To enable all sensors, send a signal to the sensor management module. 2. To turn on all cameras, send a signal to the camera management module.	3.2.2 Main Success Scenario - User using panel 2, 3
3.2.2.1.2 Receive the response signals	3.2.2 Main Success Scenario

	- User using panel 2, 3
3.2.2.1.3 Wait interrupts of sensors.	3.2.2 Main Success Scenario - User using panel 4, User accessing via internet 4
3.2.2.1.4 Lock the panel(If the state is "stay", skip)	
3.2.2.1.5 If sensors send interrupt signals,  Detected motions can be from Safehome users who want to deactivate, system should not send signal to alarm on the spot. 1. Classify signals 1-1. When door sensors send signal, Wait for deactivate in constant1 time. 1-2. After waiting time, follow next step. 2. Send signal to the 'phoncall module' and 'alarming module' on the spot	
3.2.3 Frequency of use  Very High	3.2.4 Frequency of use
3.2.4 Special conditions	
3.2.4.1 When the one of functions related to this alarm module don't send the response messages in constant2 time,  1. Resend the signal three times. 2. If the module doesn't respond, call the 'fail function' with the message, what kind of module.	3.2.3 Extensions  - User using panel and User accessing via internet 23a
3.2.4.2 When the response message represent that some components are not working or not ready,  1. To be aware of the components that are not working, send a signal again 2. Receive a message about what kind of components is not working. 3. Display 'not ready'	3.2.3 Extensions  - User using panel and User accessing via internet 2a, 3a
3.2.4.3 fail function( with message )	3.2.3 Extensions  - User using panel and User accessing via

	internet 2a, 3a
--	-----------------

### 3.3 Deactivation

#### 3.3.1 Revision History

Date	Author	Description of change
08/04/09	Jieun, Lee	Started
08/04/12	Sangyoung, Lee	Modified scenario
08/04/26	Sejoong, Kwon	Modified Main Success Scenario
08/04/27	Jieun, Lee	Modified Traceability

#### 1. Use Case: Deactivation

#### 2. Description

SafeHome system has two types of sensors; First-type of sensors sticks to doors, and windows. And second-type of sensors is motion detector. If all sensors are functioning, the system is in activated state. And if not, the system is deactivated. And these states need function which alters states.

Arming module is for switching the state between activating and deactivating arming. There are two states in activated state. One is “stay” which permits only cameras work, the other is “away” which permits all components work.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. User using panel
2. User accessing via internet

#### 5. Supporting Actors

1. Arming module
2. Permission module

#### 6. Pre-Conditions

1. User must have permission for deactivated state.
2. Deactivation only comes from activated state.

#### 7. Post Conditions

##### Success end condition

1. The system is changed to deactivated state.
2. Sensor management module turns off all sensors.
3. Camera management module turns off all cameras.
4. Deactivation never fails.
5. Number of failed trial is initialized at 0.

### Failure end condition

None

### **8. Trigger**

1. User using panel finishes inputting 4-digit password and gets permission.
2. User accessing via internet sets the system deactivated.

### **3.3.2 Main Success Scenario**

- User using panel
  1. User inputs password
  2. Permission module checks whether password is right or wrong.
  3. If the password is right, the state is changed to deactivated state.
  4. Camera management module turns off all cameras.
  5. Sensor management module turns off all sensors.
  6. If the password is not right, panel requires user to input password again.
- User accessing via internet
  1. User inputs password
  2. Permission module checks whether password is right or wrong.
  3. If the password is right, the state is changed to deactivated state.
  4. Sensor management module turns off all sensors.
  5. Camera management module turns off all cameras.
  6. If the password is not right, interface requires user to input password again.

### **3.3.3 Extensions**

- User using panel
  - 23a. In step 2,3, camera management module or sensor management module can be malfunctioning.
    1. Arming module resend the signal three times
    2. After 3 trials, Panel shows error messages about malfunctioning modules – which module is malfunctioning, what kind of malfunction it has
  - 2a. In step 2, cameras or sensors can be malfunctioning.
    1. Panel displays error information – which cameras or sensors are malfunctioning.
    2. The system tries to turn off those components.
    3. If that fails, ignore any signal from the components.
  - 3a. In step 3,
    1. The system lets cameras and sensors turn off.
    2. The state is changed to deactivated state.
  - 4a. In step 4
    1. If password which is checked in permission module fails 3 times, alarm turns on
    2. If password which is checked in permission module fails 3 times, let phonecall module make call.
    3. Whether alarm turns on or off, panel requests user using panel to input password again.
- User accessing via internet

- 23a. In step 2,3, camera management module or sensor management module can be malfunctioning.
1. Arming module resend the signal three times
  2. After 3 trials, Panel shows error messages about malfunctioning modules – which module is malfunctioning, what kind of malfunction it has
- 2a. In step 2, cameras or sensors can be malfunctioning.
1. Interface displays error information – which cameras or sensors are malfunctioning.
  2. The system tries to turn off those components.
  3. If that fails, ignore any signal from the components.
- 3a. In step 3,
1. The system lets cameras and sensors turn off.
  2. The state is changed to deactivated state.
- 4a. In step 4
1. If password which is checked in permission module fails 3 times, alarm turns on
  2. If password which is checked in permission module fails 3 times, let phonecall module make call.
  3. Whether alarm turns on or off, panel requests user using panel to input password again.

### 3.3.4 Frequency of use

Many times per day

### 3.3.5 Priority

Essential, must be implemented.

### 3.3.6 Traceability

Requirement specification about “Arming Module” is described in “Activation” and “Deactivation” use case diagram. Since Requirement specification 3.2.2.2 is Deactivation function, following traceability table will only show about 3.2.2.2. Requirement specification 3.2.2.1 is described in 2.2.6 Traceability.

Requirement Specification	Use Case Description
3.2 Arming Module	
3.2.1 Descriptions	
3.2.2 Functionality & Scenario	
3.2.2.2 Deactivate function	3.3.2 Main Success Scenario
3.2.2.2.1 To stop all sensors, send a signal to the sensor management module.	3.3.2 Main Success Scenario - User using panel 4 - User accessing via internet 4
3.2.2.2.2 To turn off all cameras, send a signal to the camera management module	3.3.2 Main Success Scenario - User using panel 5 - User accessing via internet 5
3.2.2.2.3 If the alarm is operating, send a signal to the "Arming module"	
3.2.3 Frequency of Use	3.3.4 Frequency of Use

3.2.4 Special conditions	
3.2.4.1 When the one of functions related to this alarm module don't send the response messages in constant2 time,	3.3.3. Extensions - User using panel 23a, User accessing via internet 23a
3.2.4.2 When the response message represent that some components are not working or not ready	3.3.3. Extensions - User using panel 2a, 3a, User accessing via internet 2a, 3a
3.2.4.3 fail function( with message )	3.3.3. Extensions - User using panel 2a, 3a, User accessing via internet 2a, 3a

## 3.4 Permission

### 3.4.1 Revision History

Date	Author	Description of change
08/04/09	Sangyoung, Lee	Started
08/04/12	Kanghee, Won	Modified Scenario, extension 1d
08/04/23	Sangyoung, Lee	Modified Primary Actor
08/04/26	Jieun, Lee	Modified Main Success Scenario, Extensions
08/04/27	Sejoong, Kwon	Modified Traceability

#### 1. Use Case: Permission

#### 2. Description

When the user using panel changes to activated state, system should be activated, to prevent anyone who doesn't have permission, for instances, thieves, handing the SafeHome system. All users and administrator can deactivate system by 4-digit password. If actor fails to input right password 3 times, the system will regards the actor as an intruder and perform proper action.

#### 3. Abstraction Level: Low Level

#### 4. Primary Actor

1. User using panel
2. User accessing via internet
3. Intruder

#### 5. Supporting Actors

1. Permission module

#### 6. Pre-Conditions

1. User using panel must input 4-digit password.
2. Input of permission module is 4-digit password.
3. Output of permission module is true or false.

#### 7. Post Conditions

##### Success end condition

1. Permission module returns true value.

##### Failure end condition

1. Permission module returns false value.
2. The number of failed trial increases.
3. If the number of failed trial reaches 3, SafeHome regards that user as an intruder.

#### 8. Trigger

1. User using panel or accessing via internet input passwords.

### **3.4.2 Main Success Scenario**

- User using panel

1. User using panel inputs 4-digit password.
2. Inputted password passed to Permission module.
3. Permission module compares the inputted password to true password.

- User accessing via internet

1. User accessing via internet inputs 4-digit password through internet.
2. Inputted password passed to Permission module.
3. Permission module compares the inputted password to true password.

### **3.4.3 Extensions**

- User using panel

1a. In step 1, if user intends to input less than 4-digit number, the system will perform below actions.

1. The system will wait until it gets the 4<sup>th</sup> number.

1b. In step 1, if user intends to input more than 4-digit number, the system will perform below actions.

1. The system gets only first 4 digits.
2. If first 4 digits are right, the system will ignore the rest.
3. If first 4 digits are wrong, the system recognizes the rest as new input.

3a. In step 4, if password is right.

1. Permission module returns true value.

3b. In step 4, if password is wrong

1. Permission module returns false value.
2. Increase number of failed trial.
3. Panel displays number of left chances.
4. If number of failed trial is 3, user using panel is considered as an intruder
  - A. Permission module let alarm module to turn on alarm.
  - B. Permission module let phonecall module to call the phonecall list.

- User accessing via internet

3a. In step2, if password is right

1. Permission module returns true value.

3b. In step2, if password is wrong

1. Permission module returns false value.

### **3.4.4 Frequency of use**

Many times per day

### **3.4.5 Priority**

Essential, must be implemented.

### 3.4.6 Traceability

Use case description about Permission is closely related to Permission module in Requirement specification.

Requirement specification	Use Case Description
3.3 Permission Module	
3.3.1 Descriptions	
3.3.2 Functionality & Scenario	3.4.2 Main Success Scenario
3.3.2.1 When user using panel or accessing via inputs 4-digit password, the password is passed to Permission module.	3.4.2 Main Success Scenario - User using panel 2, User accessing via internet
3.3.2.2 Compare the 4-digit with the true password.	3.4.2 Main Success Scenario - User using panel 3, User accessing via internet 3
3.3.2.4 If the password which is inputted via panel is not true, user has 2 more chance before alarming.	3.4.3 Extensions - User using panel 3b
3.3.2.5 If the password which is inputted via internet is not true, user have infinite chance to retry.	3.4.3 Extensions - User accessing via internet 3b
3.3.2.6 Control panel displays number of left chances to insert password.	3.4.3 Extensions - User using panel 3b3
3.3.2.7 But if the password still is not true, the system considers the user as an intruder.	3.4.3 Extensions - User using panel 3b4
3.3.2.7 Call the function in the 3.8 and Call the function in the 3.4	3.4.3 Extensions - User using panel 3b4
3.3.3 Frequency of use Very high	3.4.4 Frequency of use
3.3.4 Special conditions	
3.3.4.1 When the panel don't send the response a signal in constant time, 1. Resend the signal 3 times. 2. If the module doesn't respond, call the 'fail function'	
3.3.5 Constraints None	
3.3.5.1 HW Features Control panel.	

## 3.5 Access Sensors

### 3.5.1 Revision History

Date	Author	Description of change
04/08/23	Sejoong, Kwon	Started
04/08/24	Sejoong, Kwon	Modified Supporting Actor, Scenario
04/08/26	Sejoong, Kwon	Modified Traceability

#### 1. Use Case: Access Sensors

#### 2. Description

User can modify the state of sensors via internet. In Activation or Deactivation module control all the sensors at a time. But user can turn on or off each sensor.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. User accessing via internet

#### 5. Supporting Actors

1. Sensor management module
2. Sensors
3. Internet interface

#### 6. Pre-Conditions

1. Initializing of sensor must be conducted
2. User needs to get permission via internet

#### 7. Post Conditions

##### Success end condition

1. User can change states of sensors that user want to control

##### Failure end condition:

1. If there are some sensors that don't work, those components can't be used, but the others can be used. Also, notify the malfunctioning of sensors.
2. If the management system is not working, user can control anything.

#### 8. Trigger

1. User accessing via internet request to turn on or off the sensors.

### 3.5.2 Main Success Scenario

1. User accessing via internet request Internet Interface a floor plan of user's home.

2. User accessing via internet can see information about the floor plan.
3. User accessing via internet can see location information in the floor plan.
4. User accessing via internet can choose a sensor that user want to change the state.
5. The state will be changed and system will display a success message via internet.

### 3.5.3 Extensions

- 2a. If there is no floor plan or some configuration data, display error messages
- 3a. If the selected sensor is malfunctioning, display information about malfunctioning sensors.
- 4a. User accessing via internet can turn on or off the sensor.

### 3.5.4 Frequency of use

Many times per day

### 3.5.5 Priority

Not essential.

### 3.5.6 Traceability

Use case description about Access Sensors is closely related to sensor management module in Requirement specification.

Requirement Specification	Use Case Description
3.5 Sensor Management Module	
3.5.1 Descriptions	
3.5.2 Functionality & Scenario	3.5.2 Main Success Scenario
3.5.2.1 Initialize function Initialize all the sensors. Scan wireless components, and compare this result with the information stored in this module 1. If the result from scanning is same as the stored information, return success. 2. If the result from scanning is not same as the stored information, return fail.	3.1.3 Extensions 2b1
3.5.2.2 Enable function If User wants to turn on one or several sensors via internet, this module can call each enable function.	3.5.3 Extensions 4a
3.5.2.3 Disable function If User wants to turn on one or several sensors via internet, this module can call each enable function.	3.5.3 Extensions 4a
3.5.2.4 Detect function Detect motions, and send signal. 1. Window detection: send signal to 'Arming module'	

2. Door detection: send signal to 'Arming module' 3. In house motion detection: send signal to 'Alarm module'	
3.5.3 Frequency of use	3.5.4 Frequency of use
3.5.4 Special conditions 1. On the processing of 'initialize function', when a sensor doesn't react : return fail signal. 2. On the processing of 'Enable function', when a sensor is not enabled : send enable sensor fail signal to 'Alarming module' 3. On the processing of 'Disable function', when a sensor is not disabled : ignore interrupts from that sensor, and display what sensor broke down.	3.5.3 Extensions 3a
3.5.5.1 HW Features	
3.5.5.2 The maximum number of sensors The maximum number of sensors which managed by 'Sensor management module' is 20.	3.8 Configuration

## 3.6 Access Cameras

### 3.6.1 Revision History

Date	Author	Description of change
08/04/09	Sejoong, Kwon	Started
08/04/15	Kanghee, Won	Modified Supporting Actor, Scenario
08/04/26	Jieun, Lee	Modified Traceability

#### 1. Use Case: Access Camera

#### 2. Description

In the house under the control of SafeHome system, there are many cameras which photograph the living room, each personal room, and so on. These cameras are controlled by the wireless HW box. The cameras have the function about zoom in/zoom out/panning/tilting. And the data transferred between SafeHome system and each component when the user needs to see the scene of camera.

Via Internet, user can see the interior of the house and control camera.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. User accessing via internet

#### 5. Supporting Actors

1. Camera management module
2. Cameras
3. Internet interface

#### 6. Pre-Conditions

1. Initializing of camera must be conducted
2. User needs to get permission via internet

#### 7. Post Conditions

##### Success end condition

1. User can see the view of camera that the user wants to watch.

##### Failure end condition:

1. If there are some cameras that don't work, those components can't be used, but the others can be used
2. If the management system is not working, user can't use all the camera

#### 8. Trigger

1. User via Internet starts the monitoring in system.

### **3.6.2 Main Success Scenario**

1. User accessing via internet request Internet Interface a floor plan of user's home.
2. User accessing via internet can choose a camera, a file or a thumbnail.
3. User accessing via internet can select cameras which user wants to turn on or off.

### **3.6.3 Extensions**

2a. If there is no floor plan or some configuration data, display error messages.

3a. If user choose a camera,

1. User accessing via internet inputs the password of the selected camera.
2. User accessing via can see the view of the selected camera in a new window.
3. User can control the camera (Zoom/Pan/Tilt ) using the control bar in the web page.
4. During the camera view, user can order to record the view into hard disk.

3b.If user choose a file,

1. User accessing via internet inputs the password of the camera which recorded the selected file.
2. User accessing via internet can see the view of the selected file in a new window.

3c. If user choose a thumbnail,

1. User accessing via internet can see the view of all cameras in one new window. – Window is divided into the number of cameras.
2. If user wants to choose one divided view, user accessing via internet should click the view.
3. User accessing via internet inputs the password of the selected camera.
4. User accessing via internet can see the view of the selected camera in a window.
5. User can control the camera (Zoom/Pan/Tilt ) using the control bar in the web page.
6. During the camera view, user can order to record the view into hard disk.

3a.a. If the camera is not working, display error messages.

3b.a. If the file doesn't exist, display error messages.

3c.3.a. If the camera is not working, display error messages.

3a.3.a. If zoom/pan/tilt degree exceed the limited degree, the exceeded degree will be ignored.

3c.5.a. If zoom/pan/tilt degree exceed the limited degree, the exceeded degree will be ignored.

3a.4.a. If the hard disk is full, display error message.

4c.6.a. If the hard disk is full, display error message.

### **3.6.4 Frequency of use**

Many times per day

### **3.6.5 Priority**

Essential, must be implemented.

### 3.6.6 Traceability

Use case description about Access Cameras is closely related to camera management module in Requirement specification.

Requirement Specification	Use Case Description
3.6 Camera Management Module	
3.6.1 Descriptions	
3.6.2 Functionality & Scenario	3.6.2 Main Success Scenario
3.6.2.1 Initialize function Initialize all the cameras. Scan wireless components, and compare the result with the information stored in this module 1. If the result from scanning is same as the stored information, return success signal. 2. If the result from scanning is not same as the stored information, return fail signal.	3.1.3 Extensions 2b2
3.6.2.2 Password function User needs to input password when user selects each camera. User continues to input password, even if the password entered before was wrong.	3.6.3 Extensions 3a1
3.6.2.3 Enable function Enable all cameras. Each camera has enable() function. When all cameras need to be enabled, 'Camera management module' calls enable() functions in cameras.	3.6.2 Main Success Scenario 3
3.6.2.4 Disable function Disable all cameras. Each camera has enable() function. When all cameras need to be enabled, 'Camera management module' calls enable() functions in cameras.	3.6.2 Main Success Scenario 3
3.6.2.5 Data_management function	
3.6.2.5.1 Display function Display pictures what camera photographs.	3.6.2 Main Success Scenario 2
3.6.2.5.1.1 Normal mode It's a common mode. User can see only one camera.	3.6.3 Extensions 3a
3.6.2.5.1.2 Thumbnail mode User can see several camera views at a time. So user can see all the interior of user's house. The windows is divided into some camera view windows, and all the sub-windows have each own camera view. User can select one camera view in this mode. If user choose one view among this sub-windows, then viewing mode is switched to normal mode.	3.6.3 Extensions 3c
3.6.2.5.2 Record function Record pictures from camera. 'Camera management module' stores movie of constant time.	3.6.3 Extensions 3a4, 3c6

3.6.2.5.3 Zoom_in function Camera zooms in current picture.	3.6.3 Extensions 3a3, 3c5
3.6.2.5.4 Zoom_out function Camera zooms out current picture.	3.6.3 Extensions 3a3, 3c5
3.6.2.5.5 Pan function Camera moves horizontally.	3.6.3 Extensions 3a3, 3c5
3.6.2.6 Reconfigure function Modify camera list - all cameras are saved in a camera list.	3.8 Configuration
3.6.3 Frequency of use Very high	3.6.4 Frequency of use
3.6.4 Special conditions 1. On the processing of 'initialize function', when a camera doesn't react : return fail signal. 2. On the processing of 'Enable function', when a camera is not enabled : send enable fail signal to control panel. 3. On the processing of 'Disable function', when a camera is not disabled : send disable fail signal to control panel.	3.6.4 Extensions 3a.a 3c.3.a
3.6.5 Constraints	3.8 Configuration
3.6.5.1 HW Features	
3.6.5.2 The maximum number of cameras The maximum number of cameras which managed by 'Camera management module' is 20.	
3.6.5.3 The maximum movie recording time The maximum movie recording time is 1 week.	

## 3.7 Alert

### 3.7.1 Revision History

Date	Author	Description of change
08/04/9	Kanghee, Won	Started
08/04/13	Sejoong, Kwon	Modified Trigger
08/04/26	Jieun, Lee	Modified Traceability

#### 1. Use Case: Alert

#### 2. Description

In the emergency situation, which need to sound a warning, Safehome system warn the urgency of situation using some External Hardware that sounds. Also, when the user goes into the house that security function is activated and input the password, a beep sounds until the password is correct.

This use case is for the sounds and alerts.

#### 3. Abstraction Level: Sub-Function

#### 4. Primary Actor

1. User using panel

#### 5. Supporting Actors

1. Alarming Module
2. Phonecall Module

#### 6. Pre-Conditions

1. Phone lines are connected.
2. Phone call list is not empty.

#### 7. Post Conditions

##### Success end condition

1. Alarm rings
2. System makes calls to the numbers in phone call list.

##### Failure end condition

1. Alarm does not rings
2. System can't make calls to the numbers in phone call list.

#### 8. Trigger

1. User who wants to change state from activated state to deactivated state, inputs wrong password more than 3 times.

2. In activated state, sensors detect motion.
3. User using panel press panic button.

### 3.7.2 Main Success Scenario

1. When trigger occurs, system makes calls to the number in phone call list.
2. When trigger occurs, alerts alarm.

### 3.7.3 Extension

None

### 3.7.4 Frequency of use

Low

### 3.7.5 Priority

Very High, must be implanted.

### 3.7.6 Traceability

Use case description about Alert is closely related to phonecall module and alarming module in Requirement specification.

Requirement Specification	Use Case Description
3.4 Phonecall Module	
3.4.1 Descriptions	
3.4.2 Functionality & Scenario	3.7.2 Main Success Scenario
3.4.2.1 Initialize function	3.1.2 Main Success Scenario2
3.4.2.1.1 Check the phone line. 1. If the phone line is not connected or not answered, return fail 3. Else return success	3.1.3 Extensions 2a3
3.4.2.1.2 Check the phone number list 1. If the list is empty, return fail 2. Else return success	3.1.3 Extensions 2a3
3.4.2.2 Call phone function 1. 'phonecall module' receives a signal from 'password module', 'sensor module' or 'display module(panic button). other modules call this module with information about what kind of emergency is occurred. 2. Call the phone numbers listed in the stored list. Call phone. 3. Send the emergency message	3.7.2 Main Success Scenario 1
3.4.3 Frequency of use Not high. But very significant.	3.7.4 Frequency of use

3.4.4 Special conditions None	
3.4.5 Constraints	3.8 Configuration
3.4.5.1 HW Features	
3.4.5.2 Phone number list can store at most five phone numbers	
3.4.5.3 Only administrator can modify phone number list or emergency signal	

Requirement Specification	Use Case Description
3.8 Alarming module	
3.8.1 Descriptions	
3.8.2 Functionality & Scenario	3.7.2 Main Success Scenarios
3.8.2.1 Receive the signal about what condition is occurred.	
3.8.2.2 When the condition need 1. Beep sound Send a signal to the HW related to beep sound 2. The serious emergency Send a signal to the HW related to alarm sounds	
3.8.3 Frequency of use The Frequency is various on the kind of sounds. The beep sound is used at many times, but the serious emergency is hardly occurredb	3.7.4 Frequency of use
3.8.4 Special conditions None	
3.8.5 Constraints	3.8 Configuration
3.8.5.1 HW Features	

## 3.8 Configuration

### 3.8.1 Revision History

Date	Author	Description of change
08/04/10	Sejoong, Kwon	Started
08/04/13	Kanghee, Won	Modified Pre-condition
08/04/26	Sangyoung, Lee	Modified Main Success Scenario and Extension
08/04/27	Sangyoung, Lee	Modified Description

#### 1. Use Case: Configuration

#### 2. Description

In SafeHome system, user can change settings for the system. Administrator can change following things.

1. Floor Plan – the number and the place of floors, rooms, doors, windows, sensors and cameras
2. Phonecall List – add and remove phonecall list
3. Password – change password for permission, set password inserting time

To change settings, user should login to web interface with correct administrator password.

#### 3. Abstraction Level: Summary

#### 4. Primary Actor

1. Administrator

#### 5. Supporting Actors

1. Setting module

#### 6. Pre-Conditions

1. Actor must have permission as an administrator.
2. All changeable components are in setting module.

#### 7. Post Conditions

##### Success end condition

1. The changed configuration is applied.

##### Failure end condition:

1. User can't see status.

#### 8. Trigger

1. Administrator changes setting of SafeHome system.

### 3.8.2 Main Success Scenario

1. User log in to the web service as administrator.
2. Select menu to change settings.
3. Administrator selects “Floor Plan”.
4. Then administrator can see four choices – “Make new floor plan”, “Modify current floor plan”, “Load saved floor plan”, “Back”.
5. If administrator wants to change the number of floors and rooms, administrator must change them with “Make new floor plan”.
6. Work with any of three menus are done, administrator chooses “Back” to go back to previous menu.
7. Administrator selects “Phonecall List”.
8. Administrator can see the list of phonecall and “Add”, “Remove”, “Back” buttons.
9. Work with any of two menus are done, administrator chooses “Back” to go back to previous menu.
10. Administrator selects “Password”.
11. Administrator can see three choices – “Change Password”, “Password Inserting Time”, “Back”.
12. Work with any of two menus are done, administrator chooses “Back” to go back to previous menu.

### 3.8.3 Extensions

- 1a. In step 1, if user fails to login in step 1, then system will be give user chance to login again.
- 4a. In step 4, if administrator chooses “Make new floor plan”,
  1. Input the number of floors.
  2. Input the number of rooms in each floor.
  3. Then the system shows an initial display which has all floors with rooms.
  4. Administrator can set doors, windows, cameras, sensors with drag & drop.
  5. After finish setting floor plan, administrator can select “Apply” to apply new floor plan or “Save & Exit” to save and then quit or “Exit without saving” to go back.
- 4b. In step 4, if administrator chooses “Modify current floor plan”
  1. The system shows the current floor plan.
  2. Administrator can reset doors, windows, cameras, sensors with drag & drop..
  3. After finish setting floor plan, administrator can select “Apply” to apply new floor plan or “Save & Exit” to save and then quit or “Exit without saving” to go back.
- 4c. In step 4, if administrator chooses “Load saved floor plan”,
  1. The system shows the list of the saved floor plan.
  2. Administrator selects one of them, and can see selected floor plan.
  3. Administrator can reset doors, windows, cameras, sensors with drag & drop.
  4. After finish setting floor plan, administrator can select “Apply” to apply new floor plan or “Save & Exit” to save and then quit or “Exit without saving” to go back.
- 9a. In step 9, if administrator chooses “Add”, administrator can add a phonecall number to the list.
- 9b. In step 9, if administrator chooses “Remove”, administrator can remove a phonecall number from the list.

- 10a. In step 10, if administrator chooses “Change Password,”
1. Administrator inputs new password.
  2. Administrator inputs new password again for conforming.
  3. Then password is changed and administrator will see previous menu.
  4. Administrator can cancel this work in any step with pushing “Cancel” button.
- 10b. In step 10, if administrator chooses “Password Inserting Time”,
1. Display shows current password inserting time.
  2. Administrator can change the time.
  3. Time can be inputted by only numbers, and the unit of time should be in second.
  4. Administrator can finish with pushing “Done” button to apply changed setting or “Cancel” to cancel changing.

### 3.8.4 Frequency of use

Not so frequently.

### 3.8.5 Priority

Essential

### 3.8.6 Traceability

Use case description about Configuration is closely related to setting module in Requirement specification.

Requirement Specification	Use Case Description
3.7 Setting module( for Administrator )	
3.7.1 Descriptions	
3.7.2 Functionality & Scenario This module manages the list of the constant values. The list informs where the value is, what the value is, and so on.	3.8.2 Main Success Scenario
3.7.2.1 Via internet, Administrator sends the new value.	3.8.2 Main Success Scenario 4
3.7.2.2 Receive and modify the value in reference to the stored list	3.8.2 Main Success Scenario 5,7,8,11
3.7.3 Frequency of use Very low.	3.8.4 Frequency of use
3.7.4 Special conditions None	
3.7.5 Constraints There is a list stored about constant values. The list should be created at the time of delivery.	

## 3.9 Log-in

### 3.9.1 Revision History

Date	Author	Description of change
08/04/09	Sangyoung, Lee	Started
08/04/12	Kanghee, Won	Modified Scenario, extension 1d
08/04/23	Sangyoung, Lee	Modified Primary Actor

#### 1. Use Case: Log-in

#### 2. Description

When users connect the web-page via internet, all the users must log in. User can do some jobs without accessing Safehome system. User can see manuals, contact the Safehome Company, use Q&A sections and so on. If the user want to control or access the Safehome system, user also need to the other permission.

#### 3. Abstraction Level: Sub-function

#### 4. Primary Actor

1. User accessing via internet

#### 5. Supporting Actors

1. Internet interface

#### 6. Pre-Conditions

1. The user is the customer of Safehome system.
2. ID and Password was provided to users when they purchased and installed the Safehome system.

#### 7. Post Conditions

##### Success end condition

1. User logs in the web-page

##### Failure end condition

1. User doesn't log in the web-page

#### 8. Trigger

1. User using internet input ID/Passwords.

### 3.9.2 Main Success Scenario

1. User inputs ID and password.

### **3.9.3 Extensions**

1a. The length of passwords must be from 6 to 12

### **3.9.4 Frequency of use**

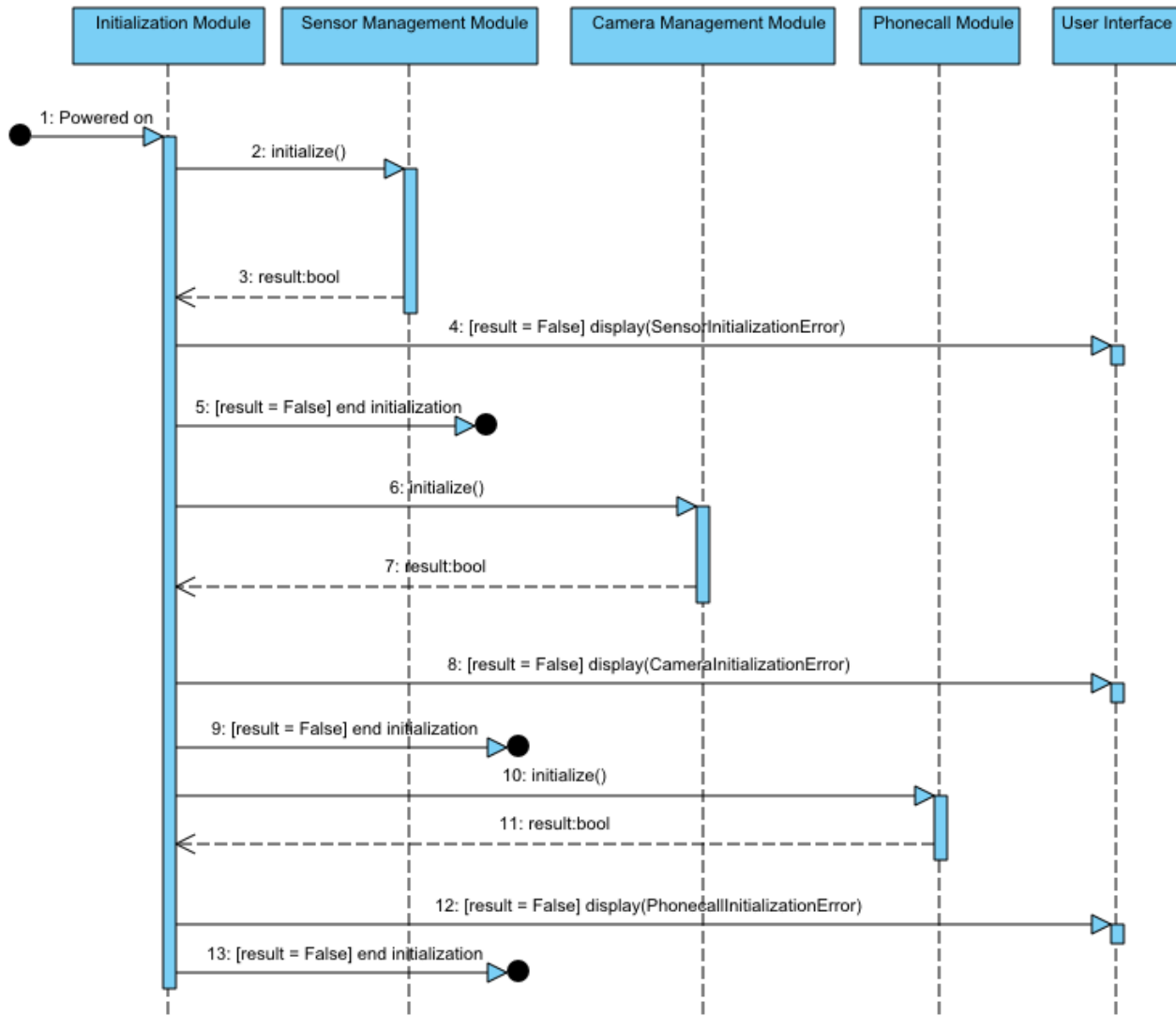
Many times per day

### **3.9.5 Priority**

Essential, must be implemented.

## 4. Sequence Diagram

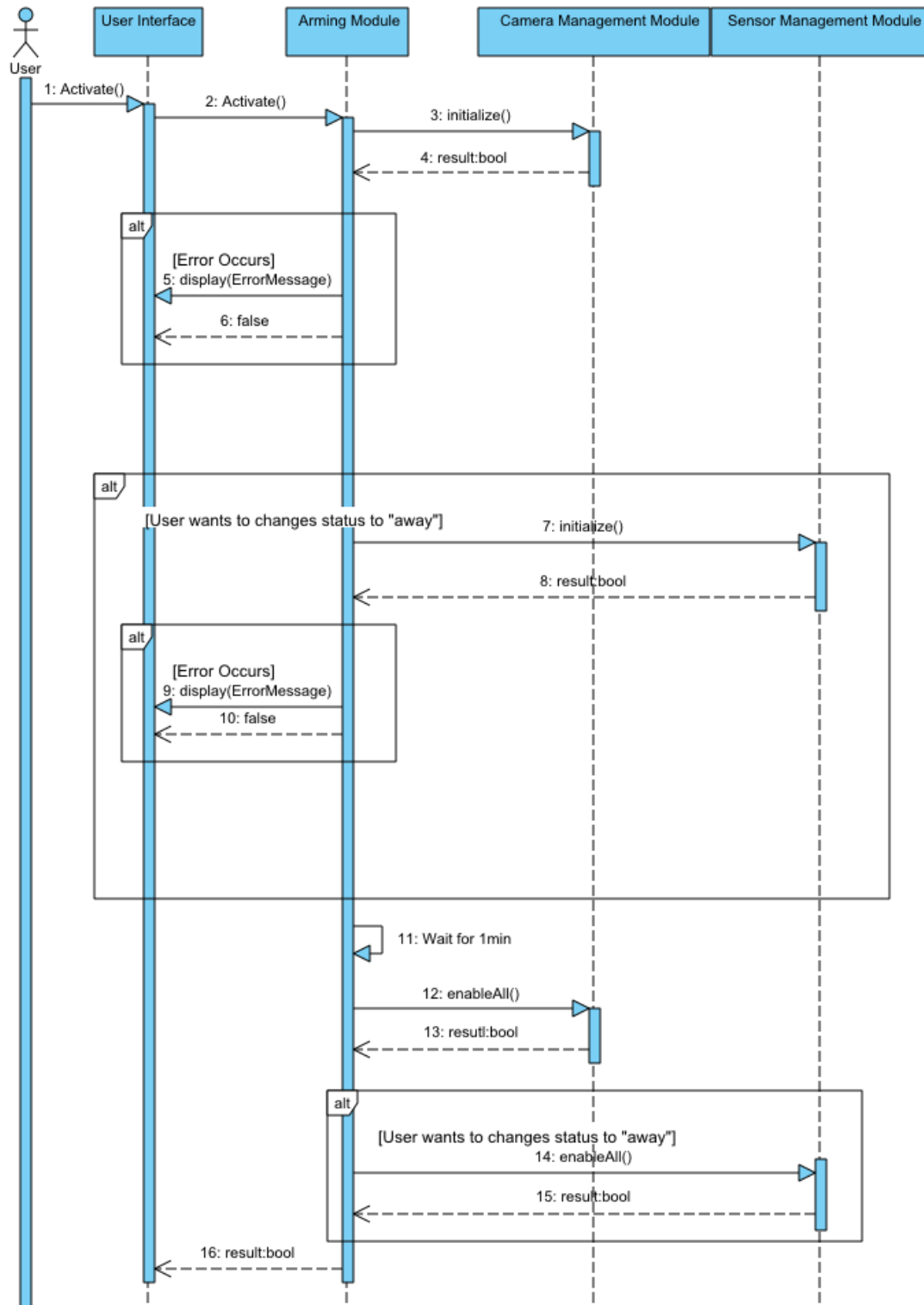
### 4.1 Initialization



1. When user powered on the System, some signal goes to the Initialization Module.
2. Initialization Module requests Sensor Management Module to initialize sensors.
3. Sensor Management Module initializes each sensors and return results to the Initialization Module.
4. If the result from Sensor Management Module is false which means failed to initialize sensors, display to the User Interface that sensor initialization has failed and abort initialization.
5. Described in step 4.
6. Initialization Module requests Camera Management Module to initialize cameras.
7. Camera Management Module initializes each cameras and return results to the Initialization Module.

8. If the result from Camera Management Module is false which means failed to initialize cameras, display to the User Interface that camera initialization has failed and abort initialization.
9. Described in step 8.
10. Initialization Module requests Phonecall Module to initialize.
11. Phonecall Module initializes itself and return result to the Initialization Module.
12. If the result from Phonecall Module is false which means failed to initialize Phonecall Module, display to the User Interface that Phonecall initialization has failed and abort initialization.
13. Described in step 12.

## 4.2 Activation



1. User requests activation on User Interface(Panel or Internet)
2. User Interface requests Arming Module to activate the system.
3. Arming Module requests Camera Management Module to initialize.
4. Camera Management Module returns result to the Arming Module.

Alt. If result is error

5. Arming Module send a message to panel so that panel can display an error.
6. Arming Module returns false to the User Interface, which means failed to activate.

Alt. If User wants to activate to 'away' state

7. Arming Module requests Sensor Management Module to initialize sensors.
8. Sensor Management Module returns result to the Arming Module.

Alt. If result is error

9. Arming Module send a message to panel so that panel can display an error.
10. Arming Module returns false to the User Interface, which means failed to activate.

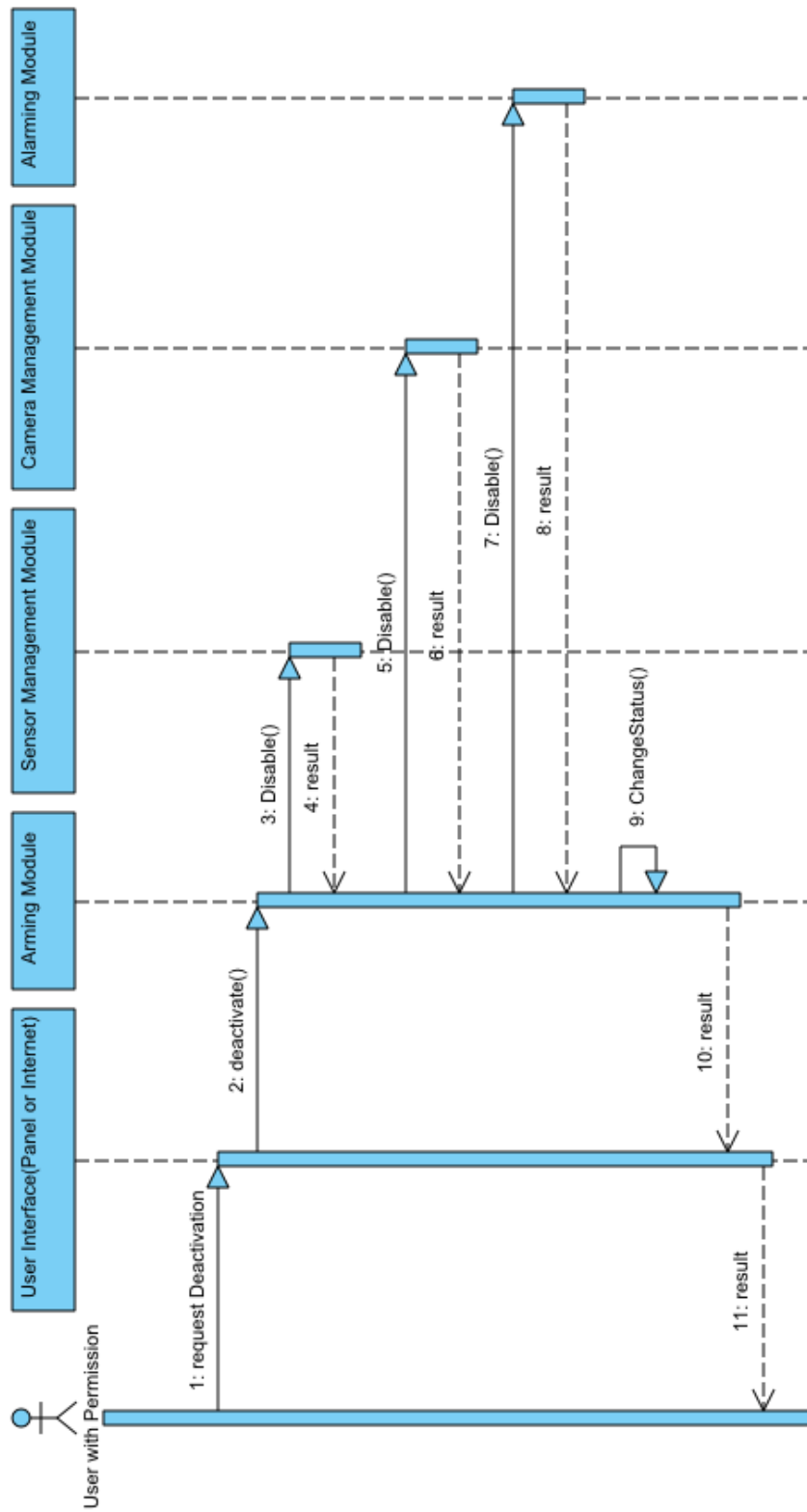
11. After checking is finished, wait for 1 minute before activation.
12. Arming Module requests Camera Management Module to enable all cameras.
13. Camera Management Module enables cameras and returns result.

Alt. If User wants to activate to 'away' state

14. Arming Module requests Sensor Management Module to enable all sensors.
15. Sensor Management Module enables sensors and returns result.

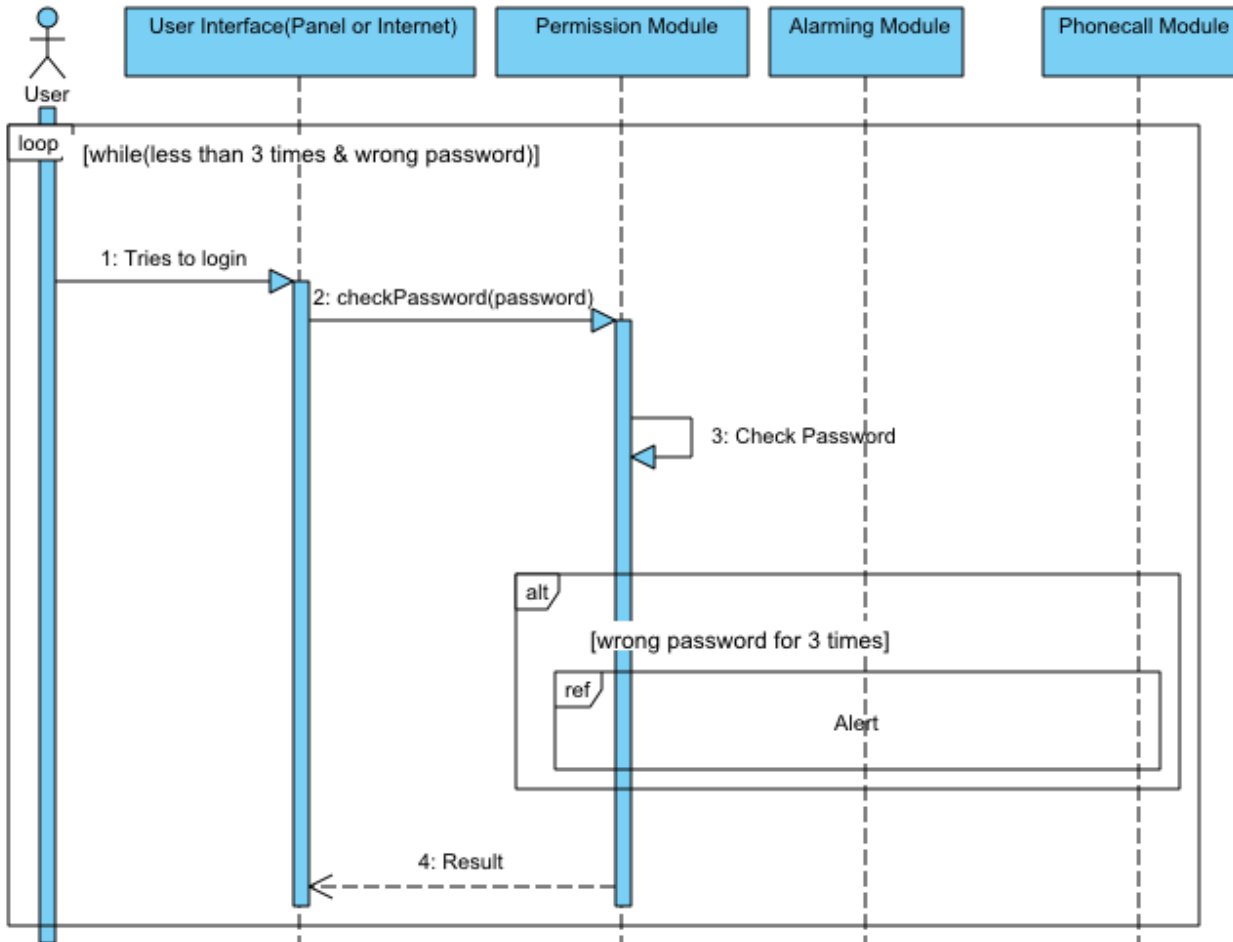
16. Arming Module returns result to User Interface.

### 4.3 Deactivation



1. User with appropriate permission requests Deactivation on User Interface. User Interface may be the Panel or Internet Interface.
2. User Interface requests Arming Module to deactivate the system.
3. Arming Module requests Sensor Management Module to disable all the sensors and gets the result.
4. Described in step 3
5. Arming Module requests Camera Management Module to disable all the cameras and gets the result.
6. Described in step 5
7. Arming Module requests Alarming Module to disable all the alarms and gets the result.
8. Described in step 7
9. Arming Module returns the result to the User Interface.
10. User Interface shows result to the User.

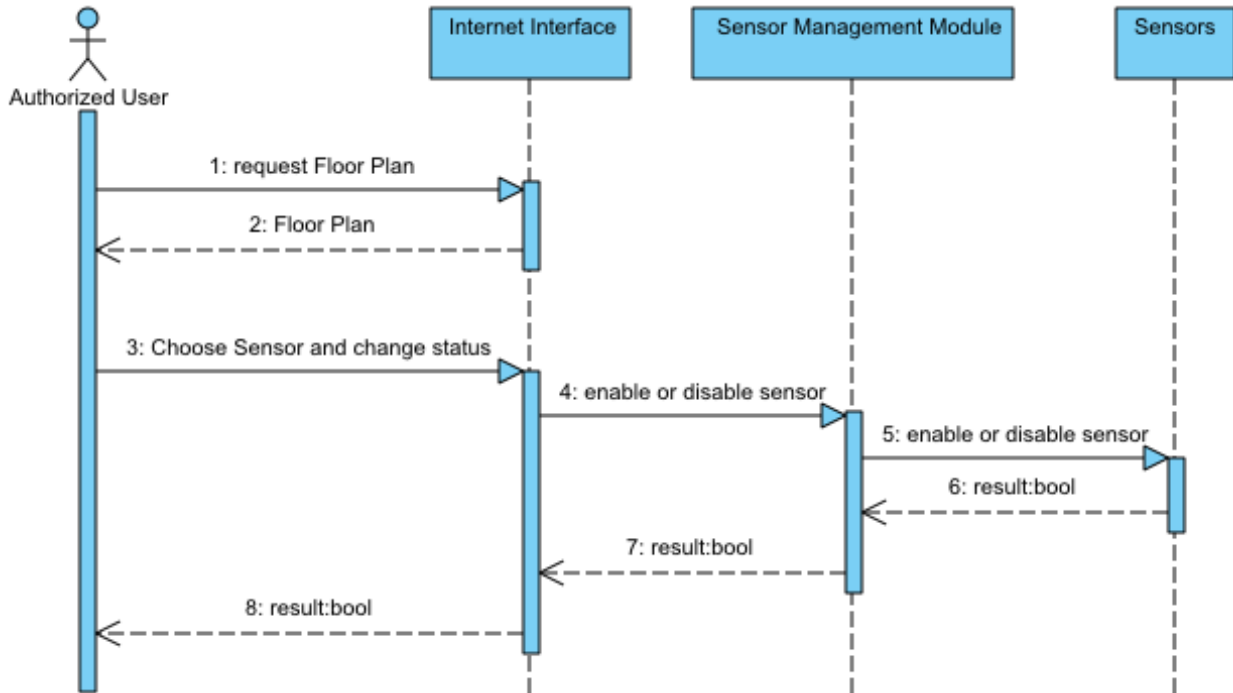
## 4.4 Permission



Loop [while (wrong password & loop has been executed less than 3 times)]

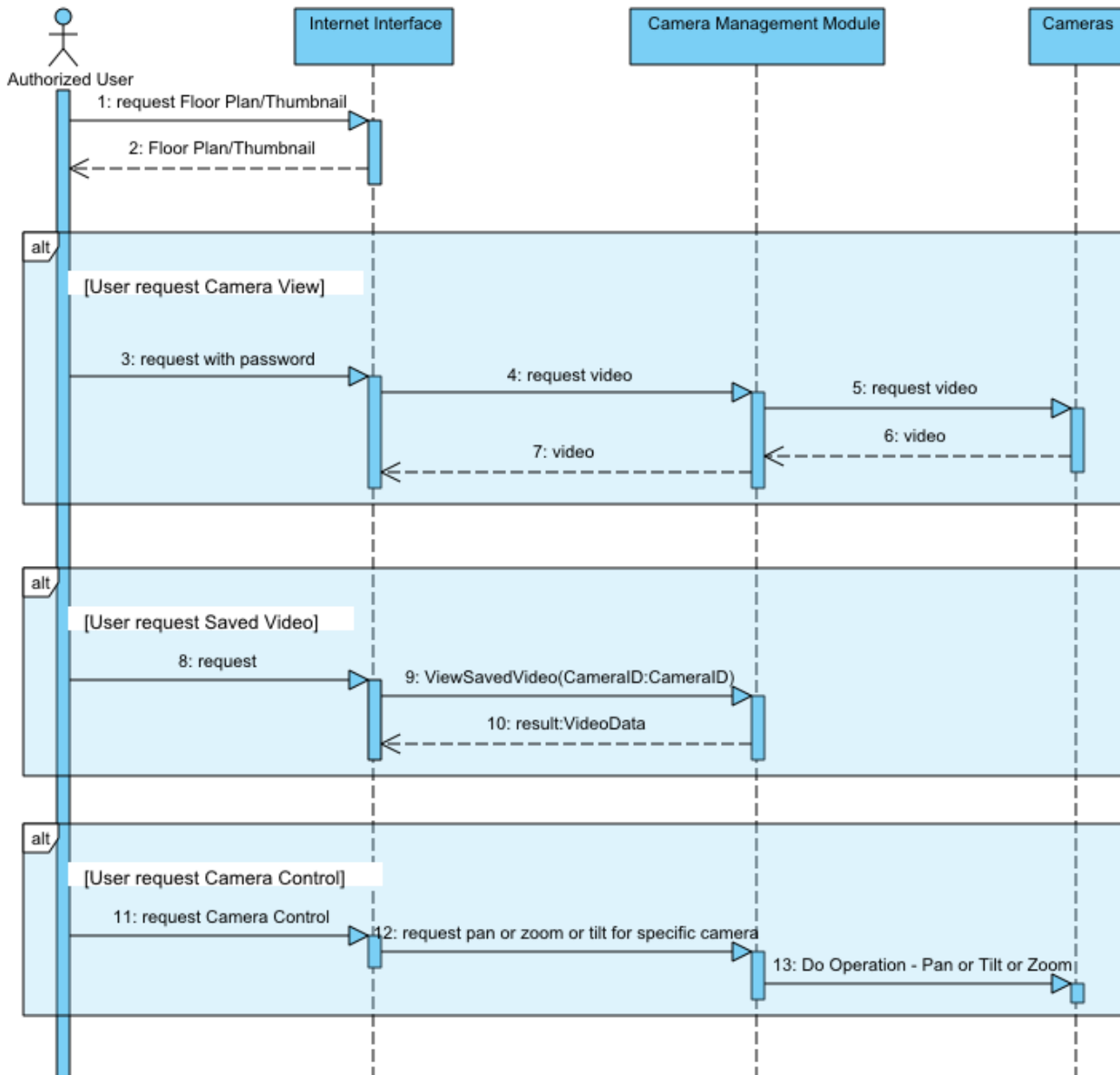
1. User tries to login through User Interface. User Interface may be the Panel or Internet Interface.
2. User Interface sends the password to Permission Module and request to compare with correct password.
3. Permission module checks the password. If this is the third try and the password is still wrong, the alarm will be activated and phone call will be made. See Sequence Diagram: Alert.
4. Permission Module returns result to the User Interface.

## 4.5 Access Sensors



1. User requests floor plan to Internet Interface.
2. Internet Interface shows floor plan to User.
3. User chooses specific sensor and turn on/off the sensor.
4. Internet Interface requests Sensor Management Module to enable/disable the sensor.
5. Sensor Management Module tries to enable/disable sensor.
6. Described in step 5.
7. Return result and show it to User.
8. Described in step 7.

## 4.6 Access Cameras



1. User request floor plan or thumbnail to the Internet Interface.
2. Internet Interface shows floor plan or thumbnail to User.

Alt. User wants to view specific camera

3. User select camera and input password.
4. Internet Interface requests video to Camera Management Module with the password.
5. Camera Management Module get video from the camera.
6. Return video.
7. Described in step 6.

Alt. User wants to see video from a saved file

8. User requests video from a saved file.

9. . Internet Interface requests video to Camera Management Module

10. Camera Management Module returns video to Internet Interface.

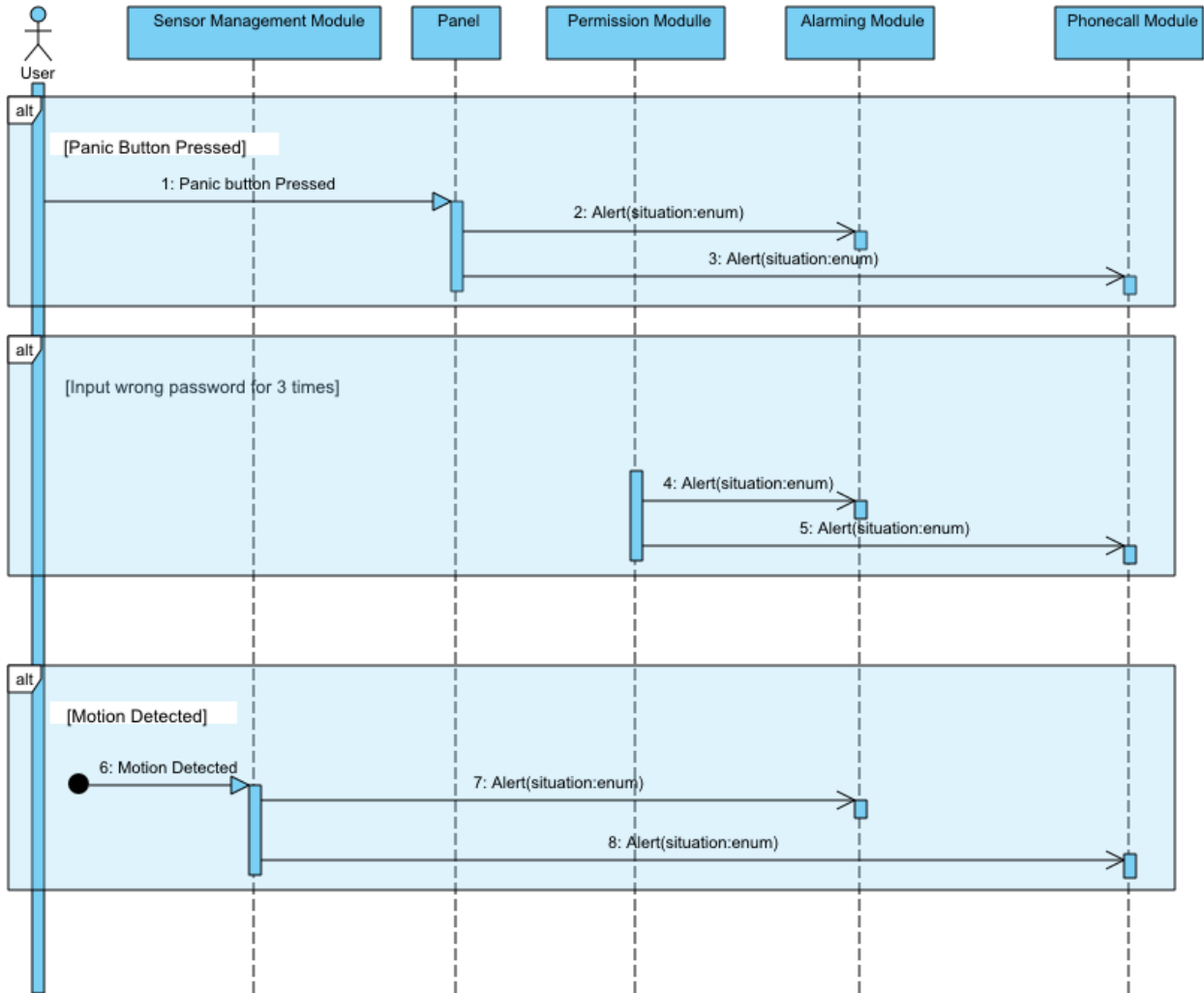
Alt. User wants to control specific camera (control may be zoom/tilt/pan)

11. User controls a camera at Internet Interface.

12. Internet Interface requests Camera Management Module to control the camera.

13. Camera Management Module controls the camera.

## 4.7 Alert



Alt. If Panic button is pressed

1. User press Panic button on the panel.
2. Panel notifies Alarming Module that user pressed Panic Button so that Alarming Module can turn on the alarm.
3. Panel notifies Phonecall Module that user pressed Panic Button so that Phonecall Module can call to the people on the phone number list.

Alt. If user input wrong password for 3 times

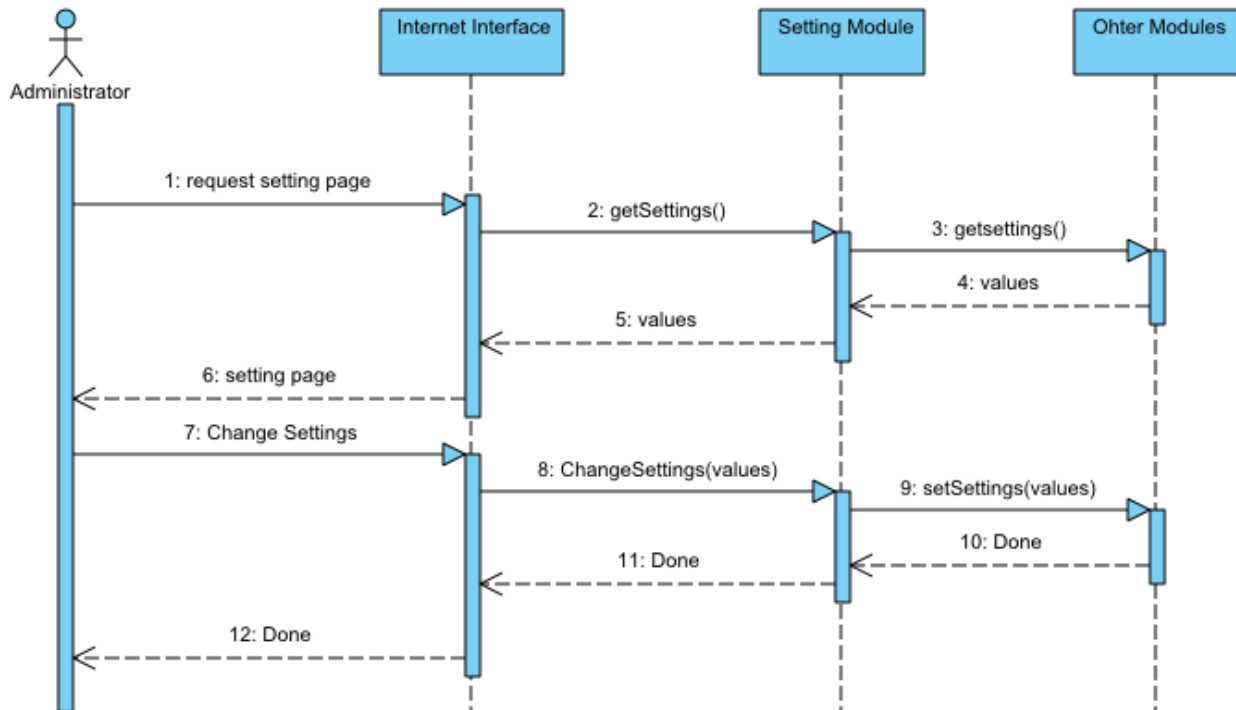
4. Permission Module notifies Alarming Module that user input wrong password for 3 times so that Alarming Module can turn on the alarm.
5. Permission Module notifies Phonecall Module that user input wrong password for 3 times so that Phonecall Module can call to the people on the phone number list.

Alt. If some motion is detected from sensors

6. Sensor notifies Sensor Management Module that some motion has been detected.

7. Sensor Management Module notifies Alarming Module that the motion is detected so that Alarming Module can turn on the alarm.
8. Sensor Management Module notifies Phonecall Module that the motion is detected so that Phonecall Module can call to the people on the phone number list.

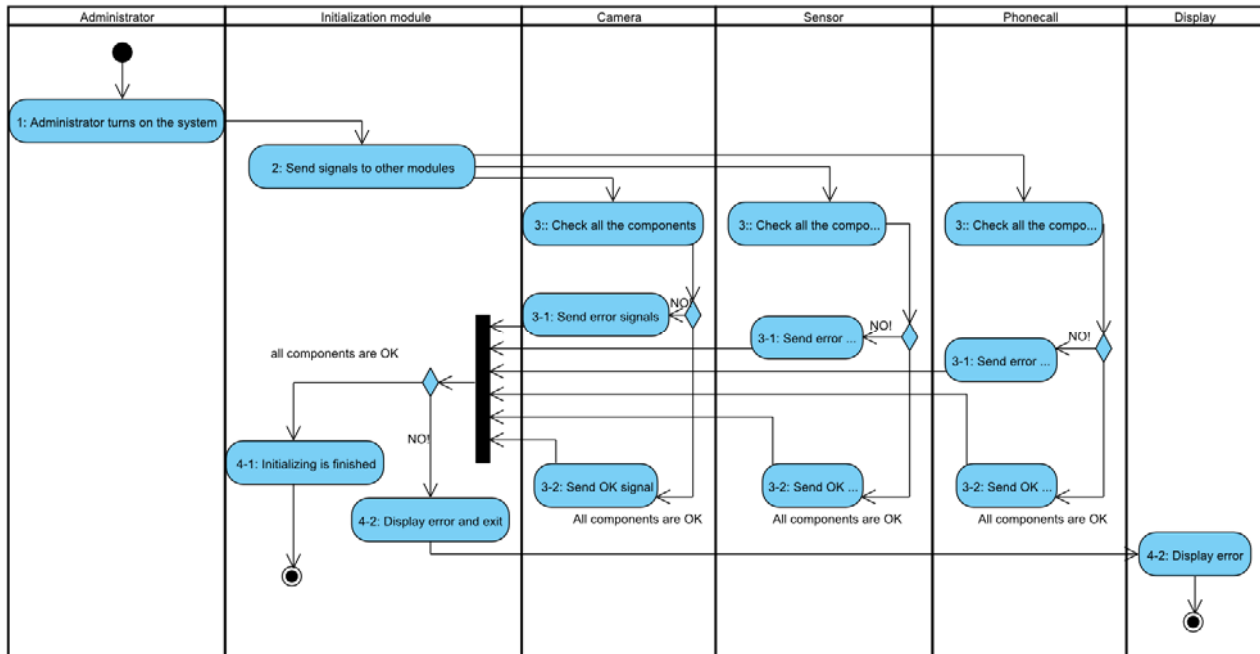
## 4.8 Configuration



1. Administrator request setting page for Internet Interface.
2. Internet Interface request overall setting values to Setting Module.
3. Setting Module request setting values for various modules.
4. Setting Module gathers setting values from various modules.
5. Setting Module returns setting values to Internet Interface.
6. Internet Interface organize setting page from values and show administrator the page.
7. Administrator changes setting values on Internet Interface.
8. Internet Interface passes changed setting values to Setting module.
9. Setting Module change values for each module.
10. After the change, Setting Module gets a message that changing has done.
11. Setting Module notifies Internet Interface that change setting is done.
12. Internet Interface notifies administrator that change setting is done.

## 5. Swimlane Diagram

### 5.1 Initialization



#### Description

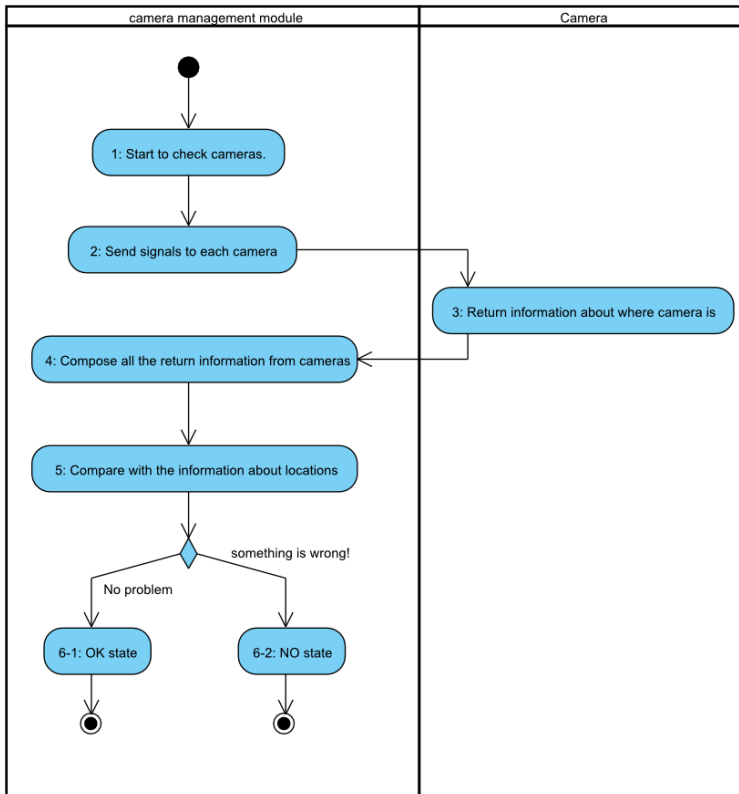
This diagram is for only “Initialization” use-case. The use-case is described in the use-case statement. When the Safehome system starts to power on, system must check the camera management module, sensor management module, and Phonecall Module are fine. The modules have the function about initializing. So, this initializing module just calls the functions in the each module. And receive the result of initializing.

Only administrator can power on the system. When the whole system is powered on, this flow is started.

#### Flow

1. Administrator turns on the system.
2. In the initialization module, send signals each module( Camera, Sensor, Phonecall module)
3. In the each module, check all components that are in the module.
  - 3-1. if there is malfunctioning parts, send error signals and return.
  - 3-2. if all are OK, send OK signals and return
 (There are three sub-diagrams about initialization function in each module.)
4. In the initialization module
  - 4-1. if all modules return OK, initialization successes.
  - 4-2. if not, display errors on display and exit.

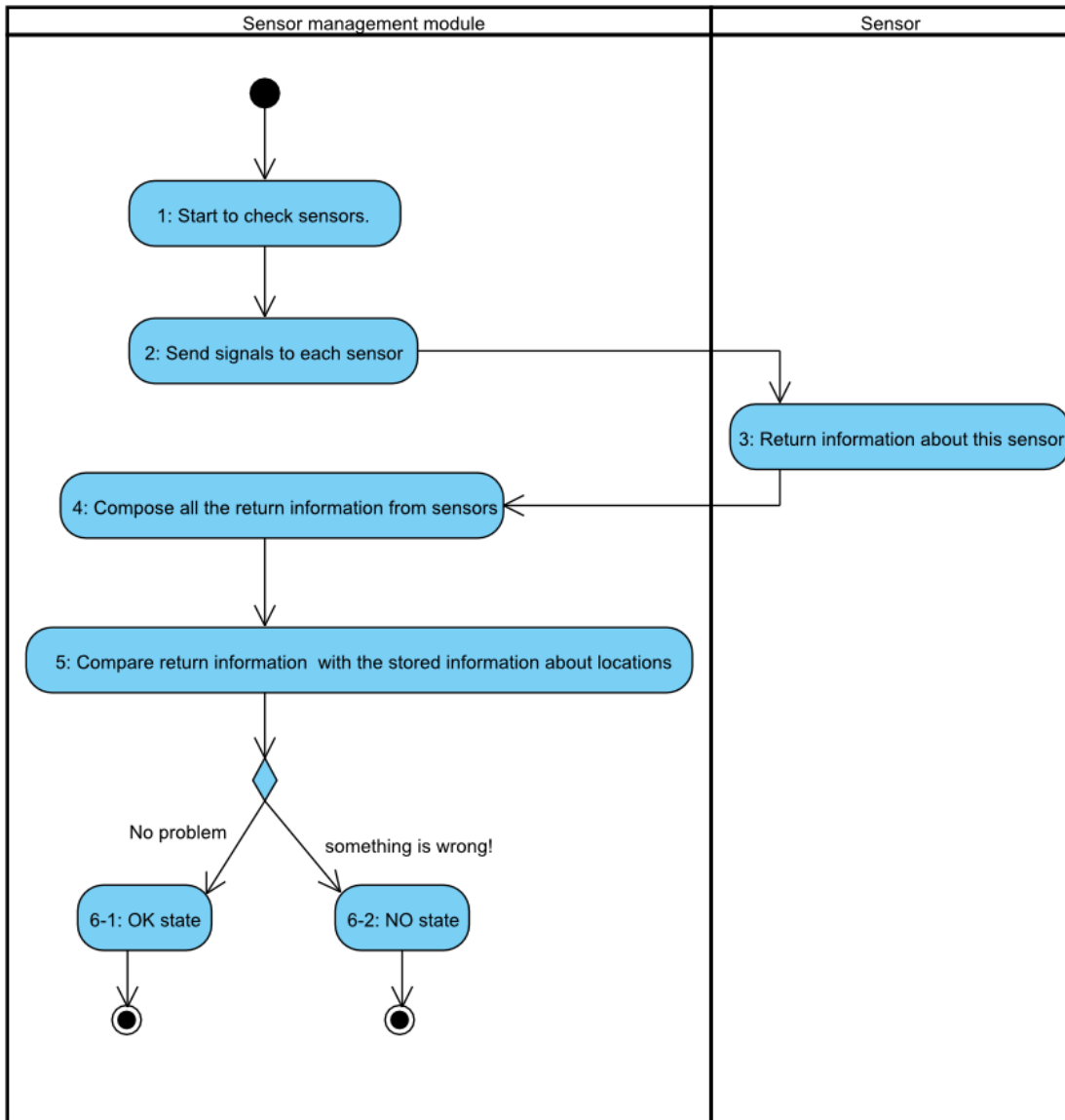
### 5.1.1 Initialization functions in the camera module.



#### Flow

1. Starting camera checking if the initialization function in the camera module.
2. Send signal to each camera via the wireless box.
3. In each camera, return the state of own.
4. Compose all the returned information.
5. Compare with the stored information.
6. If there is no malfunctioning part, no non-responsible part and the returned information is same as the stored information,
  - 6-1. if then, return OK.
  - 6-2. if not, return fail.

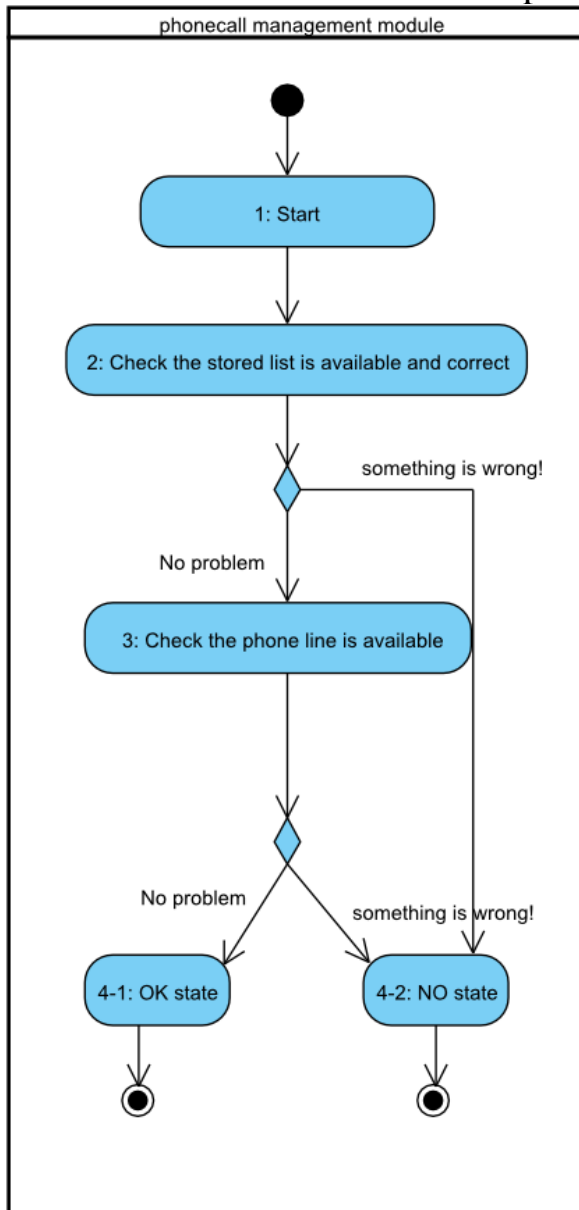
### 5.1.2 Initialization function in the sensor module.



#### Flow

1. Starting sensor checking if the initialization function in the sensor module.
2. Send signal to each sensor via the wireless box.
3. In each sensor, return the state of own.
4. Compose all the returned information
5. Compare with the stored location information
6. If there is no malfunctioning part, no non-responsible part and the information is same as stored information,
  - 6-1. if then, return OK.
  - 6-2. if not, return fail.

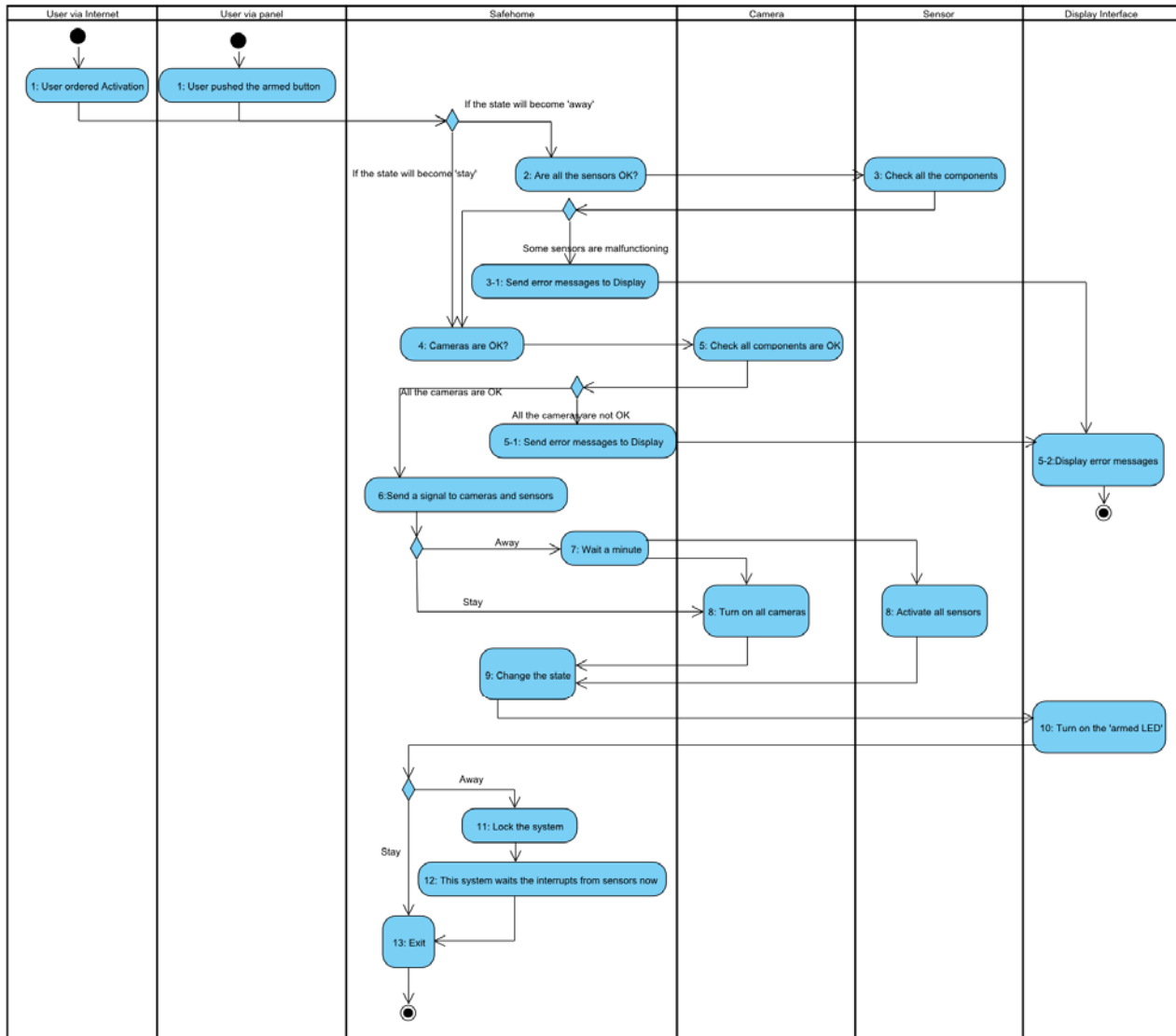
### 5.1.3 Initialization function in the phonecall module.



#### Flow

1. Starting sensor checking if the initialization function in the sensor module.
2. Check the stored list
3. Check the phone line is available.
  - 4-1. if the list and the phone line are OK, return OK.
  - 4-2. if not, return fail.

## 5.2 Activation



### Description

This diagram is for only “Activation” use-case. The use-case is described in the use-case statement. This diagram has relations with Safehome controller, Camera management, Sensor management and display. When user wants to activate the system, this diagram starts.

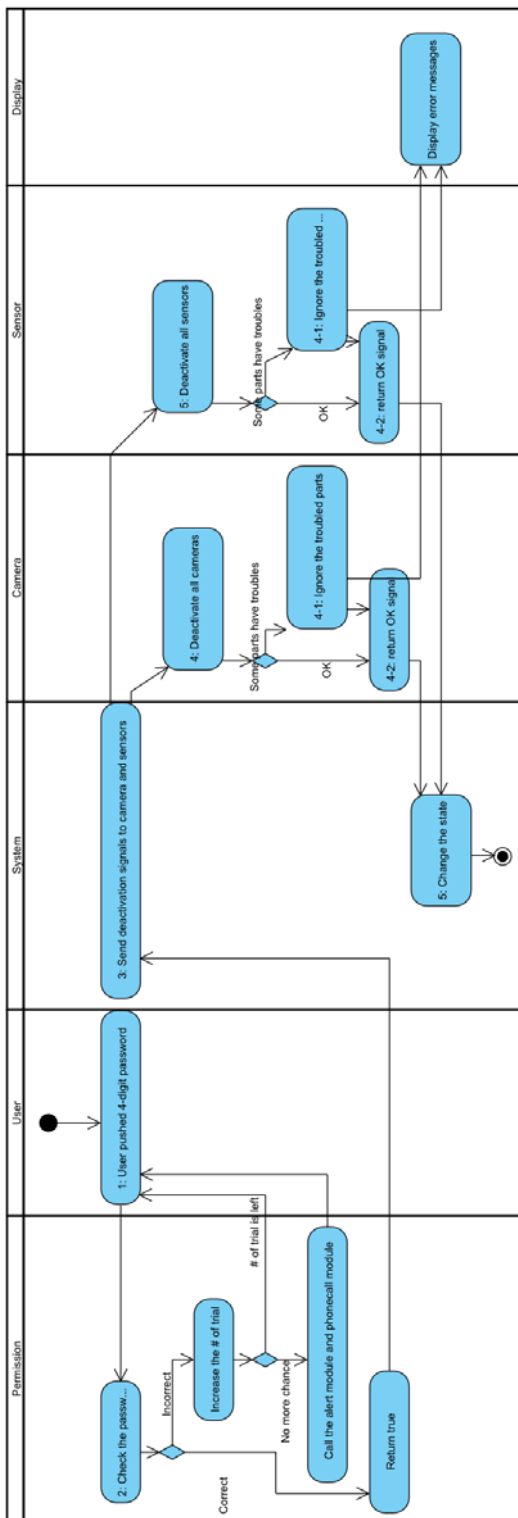
There are two states, “stay” and “away.” When user orders activation, there are two activation buttons and user chooses one button. Then, the flow is different with the state.

### Flow

1. User via panel or internet pushed the armed button.  
If the state will become “stay”
2. Send signal to sensor management module.
3. In the sensor management module, checking the component and return.  
3-1.If the return value is not “OK,” send error messages and go to step 5-2.

4. Send signal to camera management module.
5. In the camera management module, checking the component and return.
  - 5-1. if the return value is not “OK,” send error messages and go to step 5-2.
  - 5-2. in the interface, display the error.
6. Send an activation signal to camera management and sensor management (sensor management is in only “away” state).
7. If the state is “Away,” wait a minute. Because when the user pushes the button, user exits in home and user is detected by the sensors.
8. Turn on the components.
9. Change the state of system.
10. Reflect the changing of the system status. So the “armed LED” in the panel or internet is on.
11. If the state is “away,” lock the system to prevent all controls that intruder can order.
12. If the state is “away,” System waits the interrupts from sensor management until this system is deactivated.
13. The whole process is ended.

## 5.3 Deactivation



### Description

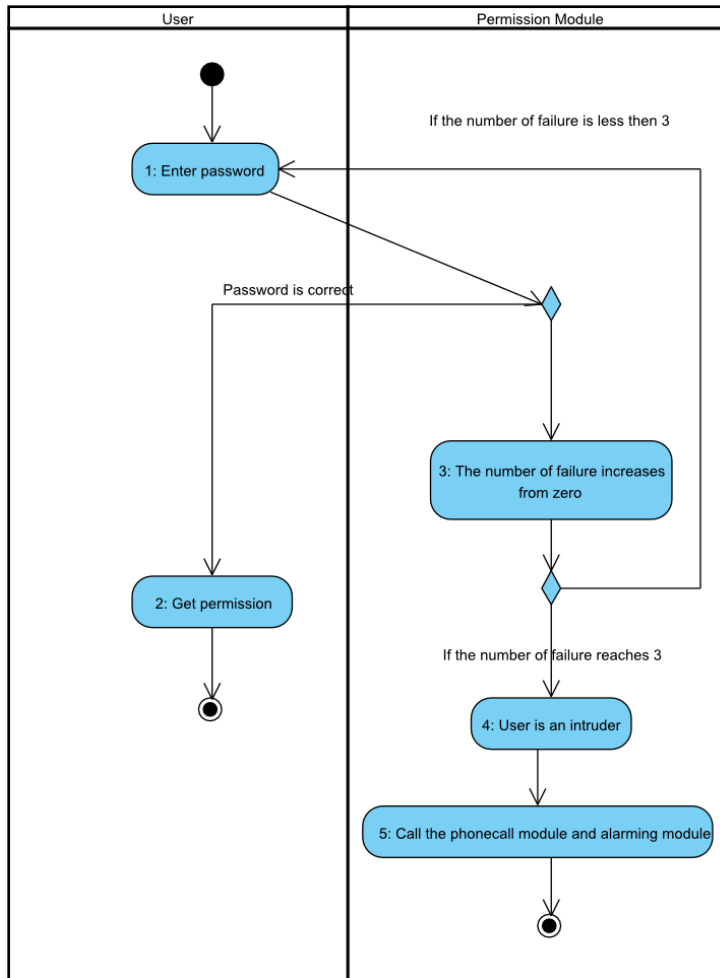
This diagram is for “Deactivation” use-case. The use-case is described in the use-case statement. This diagram has relations with Safehome controller, Camera management, Sensor management, Permission module and display. When user wants to activate the system, this diagram starts.

To deactivate the system, user must input the password correctly. The process is performed with permission module. This activity diagram describes the permission module, too.

### Flow

1. User pushed 4-digit password (In the use-case statement, the process is describe in detail).
2. Permission module checks the password (Reference the permission activity diagram)
3. If user gets the permission, send “stop” signals to cameras and sensors.
4. In each module, try to turn off the parts.
- 4-1. if nothing is troubled, return OK signal.
- 4-2. if some parts has troubled, ignore the troubled parts and display error. And go next.
5. Changing the status of system.

## Permission



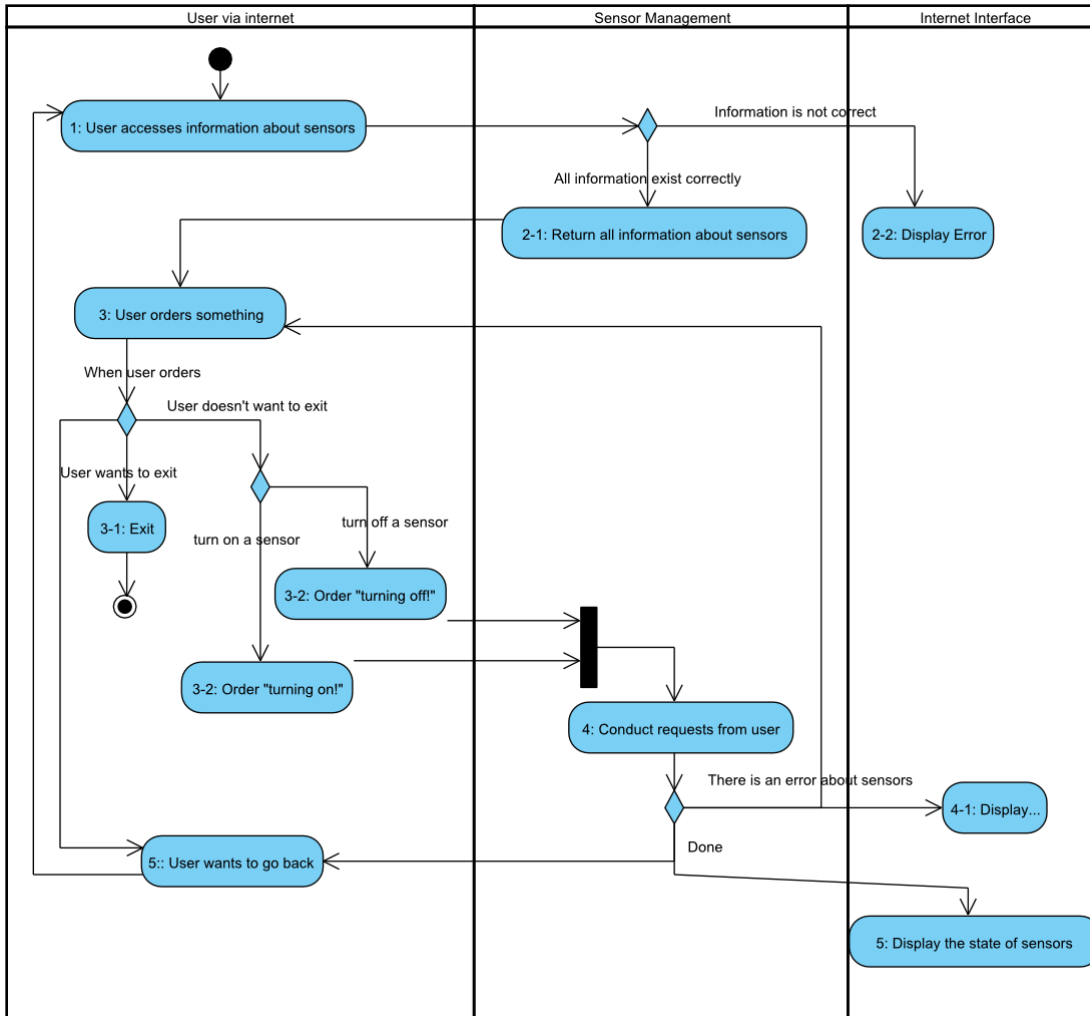
### Description

This diagram is for “Permission” use-case. The use-case is described in the use-case statement. This diagram has no direct relations with other parts. When user pushed the password, and the password is inputted into the permission module, the flow starts.

### Flow

1. User enter password.
2. Check the password in the permission module. If the password is correct, user gets permission. And exit.
3. If the password is not correct, the number of failure increases. Go back to step 1. If user try to get permission over 3 times. Go next step.
4. The system user is regarded as an intruder.
5. Call the phonecall module and alarming module( in the use-case, this modules are in the “Alert” use-case )

## 5.5 Access Sensors



### Description

This diagram is for only “Access Sensors” use-case. The use-case is described in the use-case statement. User can modify the state of sensors via internet. User can turn on or off each sensor. So this use-case has relations with Sensor management module and Internet interface. Also, main actor is user. Sensor management module has a lot of functions. And most of them are for controlling all the sensors at a time. But there are the functions that control one sensor in a time.

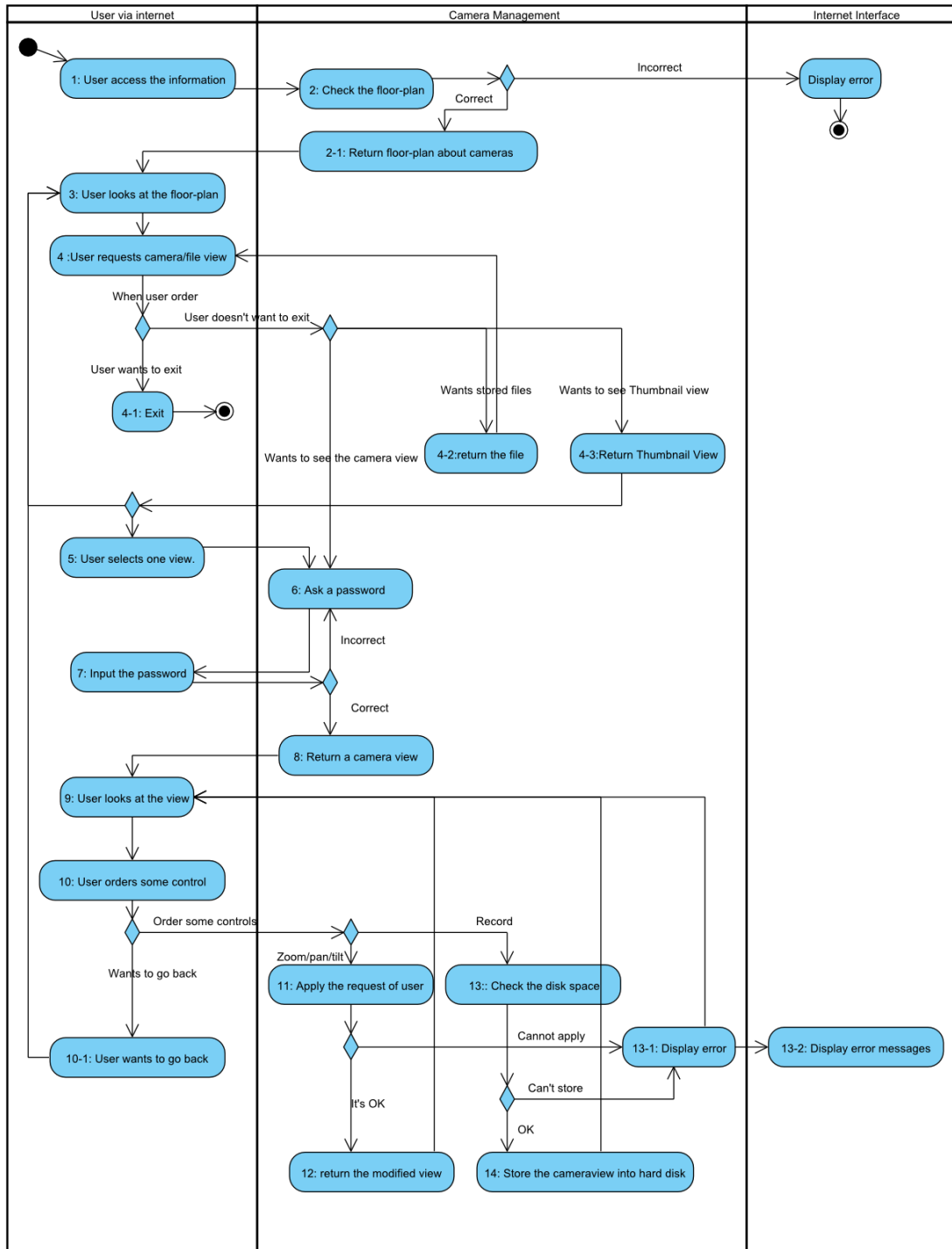
The actor is “User via Internet.” The actor is already logged-in and has permission for control. When actor requires the information about sensors, this flow starts.

### Flow

1. User accessed information about sensors. Then, the request is sent to Sensor management module.
2. Sensor management module checks the information

- 2-1. When it is correct, return that.
- 2-2. When it is not correct, display error on the internet interface.
- 3. User sees the information and order that user wants.
  - 3-1. When user wants to exit, then end.
  - 3-2. When user order “turn off” or “turn on,” then this request with what sensor will be changed is sent to Sensor management module.
- 4. Sensor management module receives the request and conducts the order of user.
  - 4-1. if there are errors in the modifying process, display errors on the internet interface. And go to step 3.
- 5. Display the information of changing and reflect the changing. And go back to step 3.

## 5.6 Access Cameras



## Description

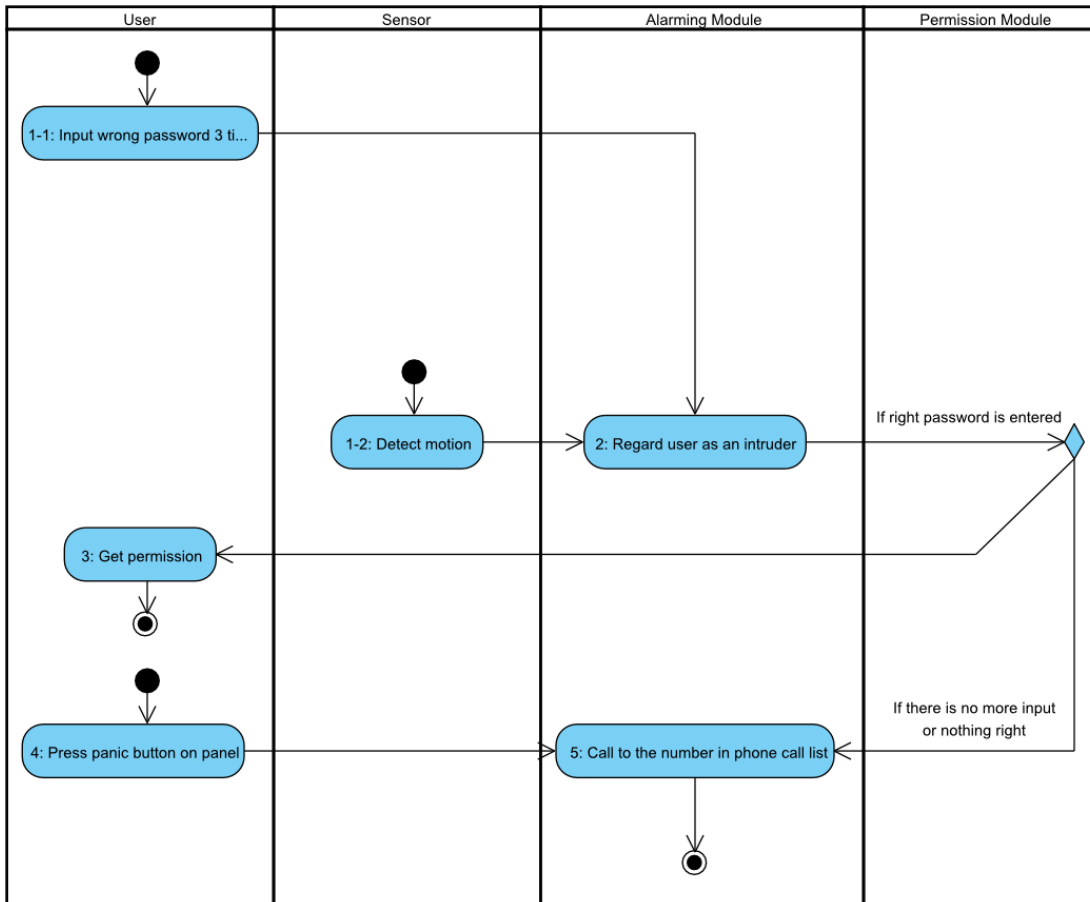
This diagram is for only “Access Cameras” use-case. The use-case is described in the use-case statement. User can access and control cameras via internet. There are many functions in camera management, panning, zooming, storing and so on. This diagram describes the activity of the accessing cameras via internet. This diagram has relations with the camera management module and an internet interface.

The actor is “User via Internet.” The actor is already logged-in and has permission for control. When actor requires the information about cameras, this flow starts.

## Flow

1. User accessed floor-plan about cameras. Then, the request is sent to Camera management module.
2. Camera management module checks the floor-plan
  - 2-1. When it is correct, return that.
  - 2-2. When it is not correct, display error on the internet interface.
3. User sees the floor-plan.
4. User orders something. The ‘something’ is
  - 4-1. “Exit”: This job is ended.
  - 4-2. one camera view: Go to step 6.
  - 4-3. Stored files view: Return the recorded file view and back to step 4.
  - 4-4. Thumbnail View: Return the thumbnail view and go next step (Here is sub-activity diagram for thumbnail view.)
5. In the thumbnail view, user selects one view.
6. Camera management asks password for each camera.
7. User input passwords.
8. If it is correct, return camera view.  
If it is not correct, go back to step 6.
9. User looks at the view.
10. User orders some control  
If the control is
  - 10-1: “Go back”: Go to step 3.
  - 10-2: “Zoom,” “Pan” or “Tilt”: Go to step 11.
  - 10-3: “Record”: Go to step 13
11. Apply the request of user.  
When the management module cannot apply, go to step 13-1.
12. Return the modified view.
13. Check the disk space.  
If the hard disk space is not enough,
  - 13-1. Display errors and go back to step 9.
  - 13-2. display errors messages on the internet interface.
14. Store the view into hard disk space.

## 5.7 Alert



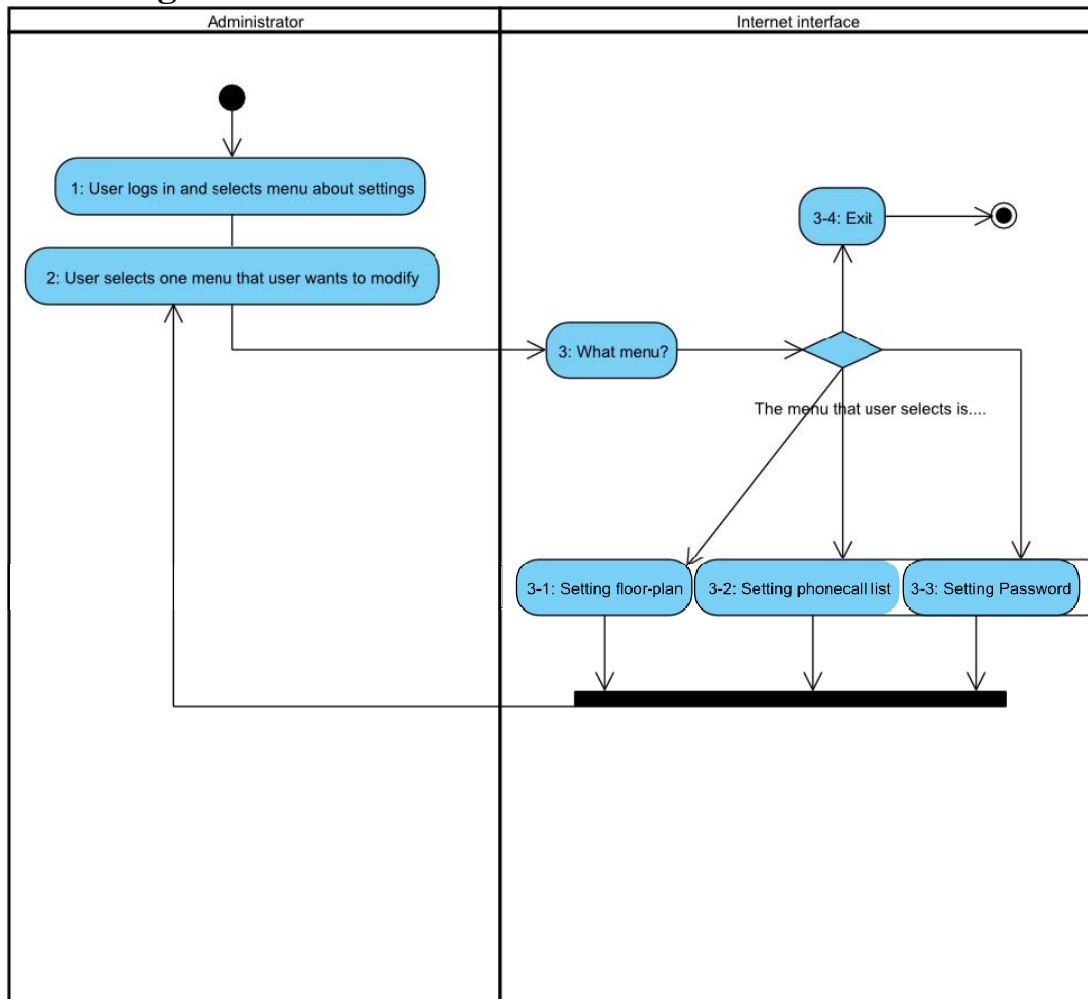
### Description

This diagram is for only “Alert” use-case. The use-case is described in the use-case statement. There are three possible paths that the Alert event is occurred. The initial nodes are 3.

### Flow

1. Initial node
  - 1-1. Input wrong password 3 times from permission module. (Initial node)
  - 1-2. Detect motion from Safehome system.
2. Regard user as an intruder.
3. If the intruder pushed right password, then he or she is not intruder. Then he or she is regarded as user, and get permission. And exit.
  - If not, go to step 5.
4. Initial node: User presses the panic button on the panel.
5. Call the phone number in the phone list of the phonecall module.

## 5.8 Configuration



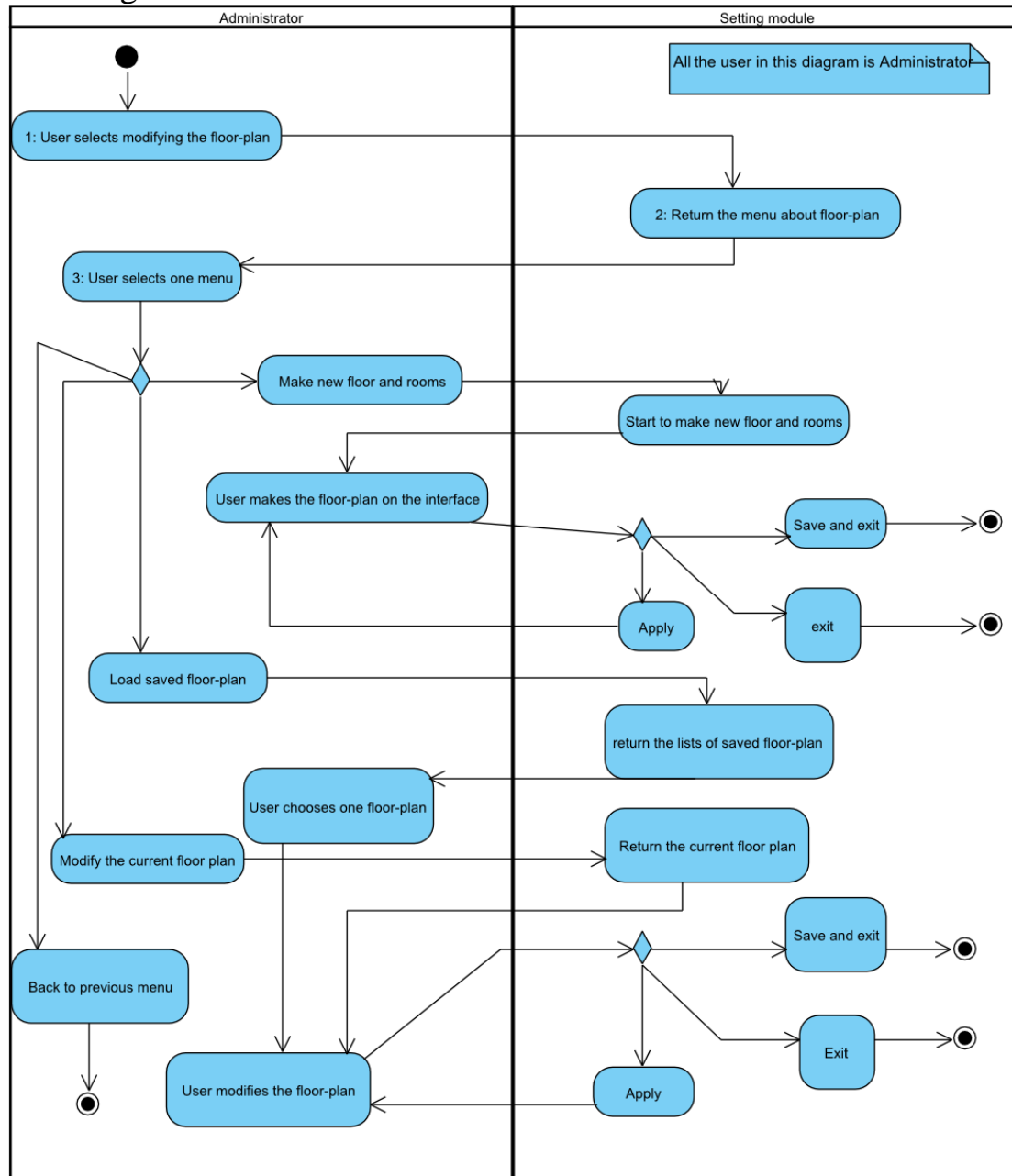
### Description

In the internet interface, Administrator can modify some information about Safehome, floor-plans, passwords, phonecall lists, and so on. This diagram is based on the use-case statement about configuration. And there are sub-diagrams because each modifying function is too complex to describe one diagram.

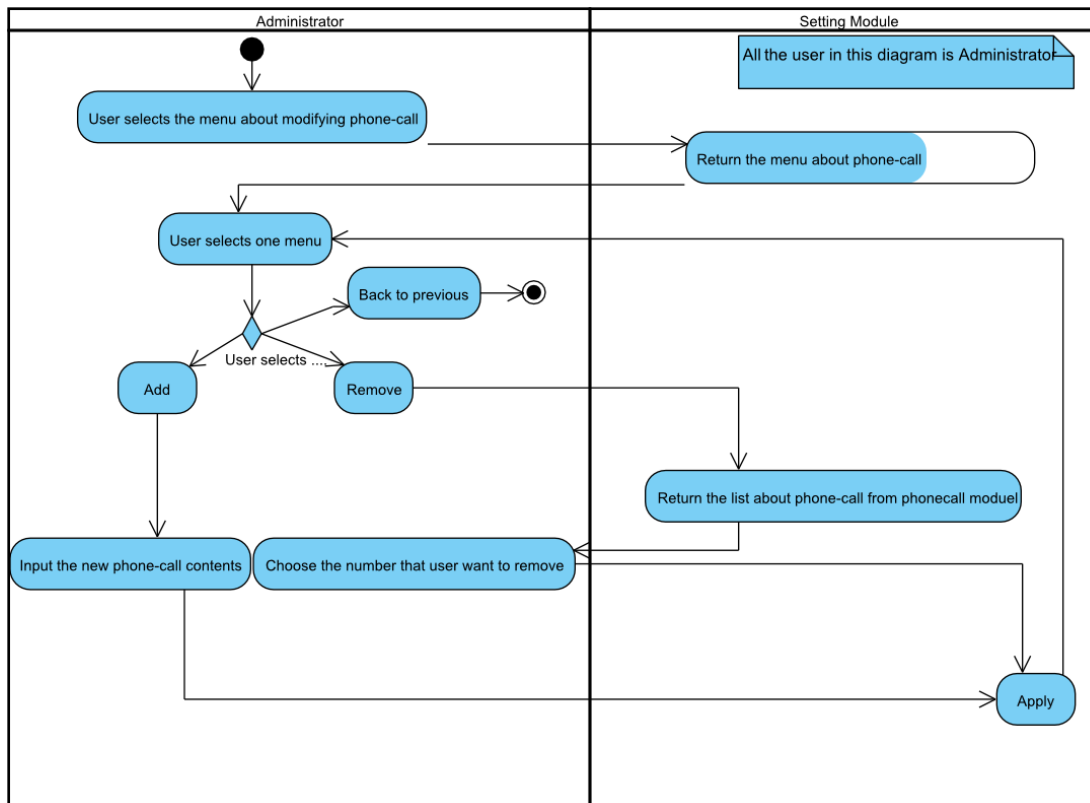
### Flow

1. User logs in and selects one menu
  2. User selects one menu that user wants to notify
  3. The module check the order
    - 3-1. Setting floor-plan -> SettingFloorPlan Activity diagram
    - 3-2. Setting phonecall list -> SettingPhonecallList Activity diagram
    - 3-3. Setting password -> SettingPassword Activity diagram
- Return to step 2

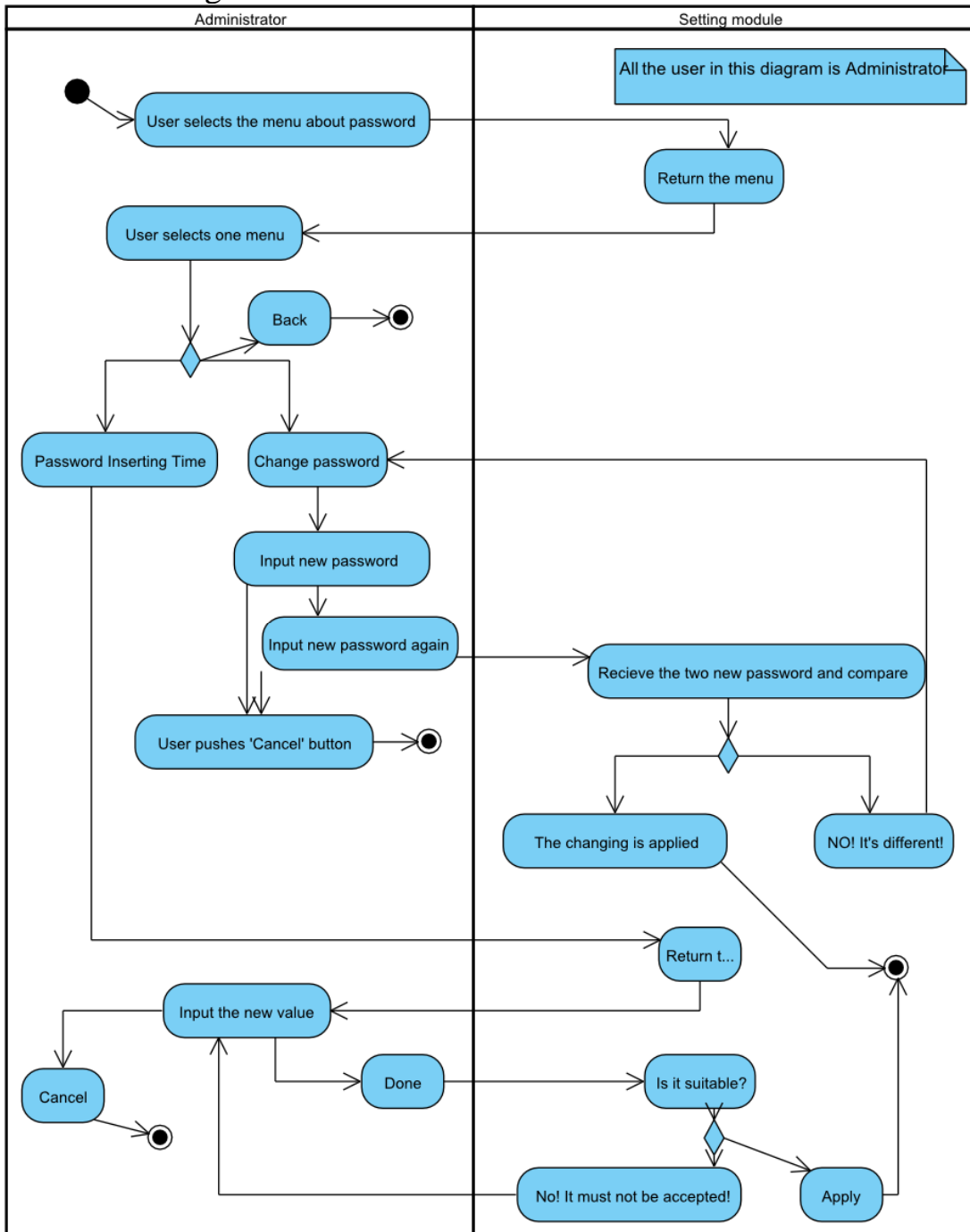
### 5.8.1 Setting Floor Plan



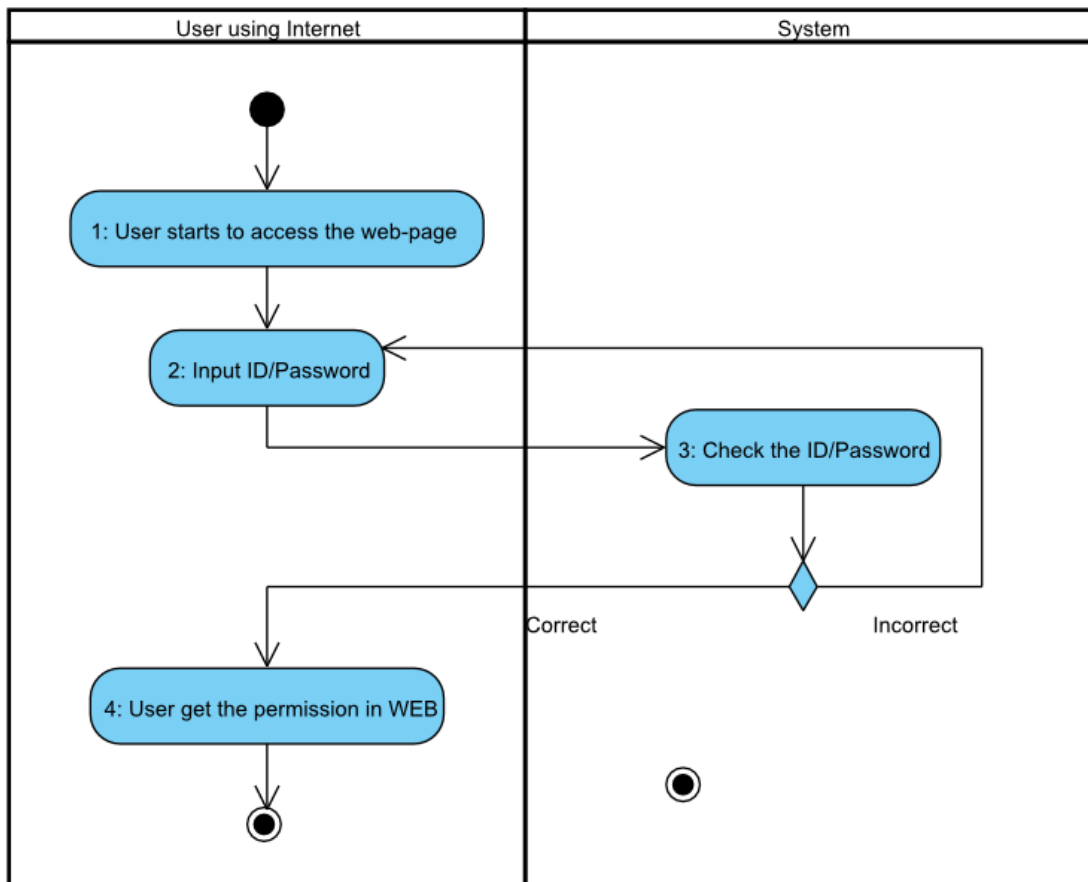
## 5.8.2 Setting Phone-call List



### 5.8.3 Setting Password



## 5.9 Log-in



### Description

This diagram is for only “Log-in” use-case. The use-case is described in the use-case statement. This diagram is for only the user via internet. When the user accesses the web, before user get the permission of system, user can do some jobs without accessing Safehome system. User can see manual, contact the Safehome Company, uses Q&A sections and so on.

### Flow

1. User starts to access the web-page.
2. User input ID and password.
3. In the web, check the ID/password.  
If it is not correct, go back to step 2(there is no trial limitation.)
4. User gets the permission in WEB.

# Appendix A: Doors Links

\*\* This is a document that represents Doors Links. There are ‘o’ marks and ‘x’ marks in the behind tables. If there is ‘o’ in (x,y), the mark means “Linked from x to y.” If there is ‘x’ in (x,y), the mark means “Linked from y to x.”

## A.1 ‘System modules’ -> ‘System Modules’

### A.1.1 Initalization module

	3.1	3.1.1	3.1.2	3.1.2.1	3.1.2.2	3.1.2.3		3.1	3.1.1	3.1.2	3.1.2.1	3.1.2.2	3.1.2.3
3 System Modules							3.4 Phonecall Module						
3.1 Initialization module							3.4.2 Functionality & Scenario						
3.1.1 Descriptions							3.4.2.1 Initialize function						
3.1.2 Functionality & Scenario							3.4.2.2 Call phone function						
3.1.2.1 call the initialize function							3.5 Sensor Management Module						
3.1.2.2 call the initialize function							3.5.2 Functionality & Scenario						
3.1.2.3 call the initialize function							3.5.2.1 Initialize function						
3.2 Arming Module							3.5.2.3 Disable function						
3.2.2 Functionality & Scenario							3.5.2.4 Detect function						
3.2.2.1 Activate function							3.6 Camera Management Module						
3.2.2.1.1 Send signal							3.6.1 Descriptions						
3.2.2.1.5 If sensors send information							3.6.2 Functionality & Scenario						
3.2.2.2 Deactivate function							3.6.2.1 Initialize function						
3.2.2.2.1 To stop all sensor							3.6.2.4 Disable function						
3.2.2.2.2 To turn off all camera							3.6.2.5 Data_management function						
3.2.2.2.3 If the alarm is opened							3.7 Setting module( for Admin)						
3.2.4 Special conditions							3.7.2 Functionality & Scenario						
3.2.4.1 When the one of function							3.8 Alarming module						
3.2.4.2 When the response is							3.8.2 Functionality & Scenario						
3.2.4.3 fail function( with message)							3.8.2.1 Receive the signal and						
3.3 Permission Module													
3.3.2 Functionality & Scenario													
3.3.2.1 When display sends													
3.3.2.4 If the password is not													
3.3.2.5 Control panel display													
3.3.2.7 Call the function in t													

### A.1.2 Arming Module

	3.2	3.2.2	3.2.2.1	3.2.2.1.1	3.2.2.1.5	3.2.2.2	3.2.2.2.1	3.2.2.2.2	3.2.2.2.3	3.2.4	3.2.4.1	3.2.4.2	3.2.4.3		3.2	3.2.2	3.2.2.1	3.2.2.1.1	3.2.2.1.5	3.2.2.2	3.2.2.2.1	3.2.2.2.2	3.2.2.2.3	3.2.4	3.2.4.1	3.2.4.2	3.2.4.3
3 System Modules														3.4 Phonecall Module													
3.1 Initialization module														3.4.2 Functionality & Scenario													
3.1.1 Descriptions														3.4.2.1 Initialize function													
3.1.2 Functionality & Scenario														3.4.2.2 Call phone function													
3.1.2.1 call the initialize function														3.5 Sensor Management Module													
3.1.2.2 call the initialize function														3.5.2 Functionality & Scenario													
3.1.2.3 call the initialize function														3.5.2.1 Initialize function													
3.2 Arming Module														3.5.2.3 Disable function													
3.2.2 Functionality & Scenario														3.5.2.4 Detect function													
3.2.2.1 Activate function														3.6 Camera Management Module													
3.2.2.1.1 Send signal														3.6.1 Descriptions													
3.2.2.1.5 If sensors send information														3.6.2 Functionality & Scenario													
3.2.2.2 Deactivate function														3.6.2.1 Initialize function													
3.2.2.2.1 To stop all sensor														3.6.2.4 Disable function													
3.2.2.2.2 To turn off all camera														3.6.2.5 Data_management function													
3.2.2.2.3 If the alarm is opened														3.7 Setting module( for Admin)													
3.2.4 Special conditions														3.7.2 Functionality & Scenario													
3.2.4.1 When the one of function														3.8 Alarming module													
3.2.4.2 When the response is														3.8.2 Functionality & Scenario													
3.2.4.3 fail function( with message)														3.8.2.1 Receive the signal and													
3.3 Permission Module																											
3.3.2 Functionality & Scenario																											
3.3.2.1 When display sends																											
3.3.2.4 If the password is not																											
3.3.2.5 Control panel display																											
3.3.2.7 Call the function in t																											

## A.1.3 Permission Module

	3.3	3.3.2	3.3.2.1	3.3.2.4	3.3.2.5	3.3.2.7		3.3	3.3.2	3.3.2.1	3.3.2.4	3.3.2.5	3.3.2.7
3 System Modules							3.4 Phonecall Module						
3.1 Initialization module							3.4.2 Functionality & Scenario					o	
3.1.1 Descriptions							3.4.2.1 Initialize function						
3.1.2 Functionality & Scenario							3.4.2.2 Call phone function						
3.1.2.1 call the initialize function							3.5 Sensor Management Module						
3.1.2.2 call the initialize function							3.5.2 Functionality & Scenario						
3.1.2.3 call the initialize function							3.5.2.1 Initialize function						
3.2 Arming Module							3.5.2.3 Disable function						
3.2.2 Functionality & Scenario							3.5.2.4 Detect function						
3.2.2.1 Activate function							3.6 Camera Management Module						
3.2.2.1.1 Send signal							3.6.1 Descriptions						
3.2.2.1.5 If sensors send information							3.6.2 Functionality & Scenario						
3.2.2.2 Deactivate function							3.6.2.1 Initialize function						
3.2.2.2.1 To stop all sensor							3.6.2.4 Disable function						
3.2.2.2.2 To turn off all camera							3.6.2.5 Data_management function						
3.2.2.2.3 If the alarm is opened							3.7 Setting module( for Admin)						
3.2.4 Special conditions							3.7.2 Functionality & Scenario						
3.2.4.1 When the one of function							3.8 Alarming module						
3.2.4.2 When the response is							3.8.2 Functionality & Scenario					o	
3.2.4.3 fail function( with module)							3.8.2.1 Receive the signal and						
3.3 Permission Module													
3.3.2 Functionality & Scenario													
3.3.2.1 When display sends													
3.3.2.4 If the password is not													
3.3.2.5 Control panel display													
3.3.2.7 Call the function in the													

## A.1.4 Phonecall Module

	3.4	3.4.2	3.4.2.1	3.4.2.2		3.4	3.4.2	3.4.2.1	3.4.2.2
3 System Modules					3.4 Phonecall Module				
3.1 Initialization module					3.4.2 Functionality & Scenario				
3.1.1 Descriptions					3.4.2.1 Initialize function				
3.1.2 Functionality & Scenario					3.4.2.2 Call phone function				
3.1.2.1 call the initialize function					3.5 Sensor Management Module				
3.1.2.2 call the initialize function					3.5.2 Functionality & Scenario				
3.1.2.3 call the initialize function			x		3.5.2.1 Initialize function				
3.2 Arming Module					3.5.2.3 Disable function				
3.2.2 Functionality & Scenario					3.5.2.4 Detect function				
3.2.2.1 Activate function					3.6 Camera Management Module				
3.2.2.1.1 Send signal					3.6.1 Descriptions				
3.2.2.1.5 If sensors send information				x	3.6.2 Functionality & Scenario				
3.2.2.2 Deactivate function					3.6.2.1 Initialize function				
3.2.2.2.1 To stop all sensor					3.6.2.4 Disable function				
3.2.2.2.2 To turn off all camera					3.6.2.5 Data_management function				
3.2.2.2.3 If the alarm is opened					3.7 Setting module( for Admin)				
3.2.4 Special conditions					3.7.2 Functionality & Scenario				
3.2.4.1 When the one of function					3.8 Alarming module				
3.2.4.2 When the response is					3.8.2 Functionality & Scenario				
3.2.4.3 fail function( with module)					3.8.2.1 Receive the signal and				
3.3 Permission Module									
3.3.2 Functionality & Scenario									
3.3.2.1 When display sends									
3.3.2.4 If the password is not									
3.3.2.5 Control panel display									
3.3.2.7 Call the function in the			x						

## A.1.5 Sensor Management Module

	3.5	3.5.2	3.5.2.1	3.5.2.3	3.5.2.4		3.5	3.5.2	3.5.2.1	3.5.2.3	3.5.2.4
3 System Modules						3.4 Phonecall Module					
3.1 Initialization module						3.4.2 Functionality & Scenario					
3.1.1 Descriptions						3.4.2.1 Initialize function					
3.1.2 Functionality & Scenario						3.4.2.2 Call phone function					
3.1.2.1 call the initialize function			x			3.5 Sensor Management Module					
3.1.2.2 call the initialize function						3.5.2 Functionality & Scenario					
3.1.2.3 call the initialize function						3.5.2.1 Initialize function					
3.2 Arming Module						3.5.2.3 Disable function					
3.2.2 Functionality & Scenario						3.5.2.4 Detect function					
3.2.2.1 Activate function						3.6 Camera Management Module					
3.2.2.1.1 Send signal						3.6.1 Descriptions					
3.2.2.1.5 If sensors send information			x			3.6.2 Functionality & Scenario					
3.2.2.2 Deactivate function						3.6.2.1 Initialize function					
3.2.2.2.1 To stop all sensor				x		3.6.2.4 Disable function					
3.2.2.2.2 To turn off all camera						3.6.2.5 Data_management function					
3.2.2.2.3 If the alarm is open						3.7 Setting module( for Admin)					
3.2.4 Special conditions						3.7.2 Functionality & Scenario					
3.2.4.1 When the one of function						3.8 Alarming module					
3.2.4.2 When the response is						3.8.2 Functionality & Scenario					o
3.2.4.3 fail function( with module)						3.8.2.1 Receive the signal and					
3.3 Permission Module											
3.3.2 Functionality & Scenario											
3.3.2.1 When display sends											
3.3.2.4 If the password is not											
3.3.2.5 Control panel display											
3.3.2.7 Call the function in the											

## A.1.6 Camera Management Module

	3.6	3.6.1	3.6.2	3.6.2.1	3.6.2.4	3.6.2.5		3.6	3.6.1	3.6.2	3.6.2.1	3.6.2.4	3.6.2.5
3 System Modules							3.4 Phonecall Module						
3.1 Initialization module							3.4.2 Functionality & Scenario						
3.1.1 Descriptions							3.4.2.1 Initialize function						
3.1.2 Functionality & Scenario							3.4.2.2 Call phone function						
3.1.2.1 call the initialize function							3.5 Sensor Management Module						
3.1.2.2 call the initialize function				x			3.5.2 Functionality & Scenario						
3.1.2.3 call the initialize function							3.5.2.1 Initialize function						
3.2 Arming Module							3.5.2.3 Disable function						
3.2.2 Functionality & Scenario							3.5.2.4 Detect function						
3.2.2.1 Activate function							3.6 Camera Management Module						
3.2.2.1.1 Send signal							3.6.1 Descriptions						
3.2.2.1.5 If sensors send information							3.6.2 Functionality & Scenario						
3.2.2.2 Deactivate function							3.6.2.1 Initialize function						
3.2.2.2.1 To stop all sensor							3.6.2.4 Disable function						
3.2.2.2.2 To turn off all camera					x		3.6.2.5 Data_management function						
3.2.2.2.3 If the alarm is open							3.7 Setting module( for Admin)						
3.2.4 Special conditions							3.7.2 Functionality & Scenario						
3.2.4.1 When the one of function							3.8 Alarming module						
3.2.4.2 When the response is							3.8.2 Functionality & Scenario						
3.2.4.3 fail function( with module)							3.8.2.1 Receive the signal and						
3.3 Permission Module													
3.3.2 Functionality & Scenario													
3.3.2.1 When display sends													
3.3.2.4 If the password is not													
3.3.2.5 Control panel display													
3.3.2.7 Call the function in the													

## A.1.7 Setting Module & Alarming Module

	3.7	3.7.2	3.8	3.8.2	3.8.2.1		3.7	3.7.2	3.8	3.8.2	3.8.2.1
<b>3 System Modules</b>											
<b>3.1 Initialization module</b>											
<b>3.1.1 Descriptions</b>											
<b>3.1.2 Functionality &amp; Scenario</b>											
3.1.2.1 call the initialize function											
3.1.2.2 call the initialize function											
3.1.2.3 call the initialize function											
<b>3.2 Arming Module</b>											
<b>3.2.2 Functionality &amp; Scenario</b>											
<b>3.2.2.1 Activate function</b>											
3.2.2.1.1 Send signal											
3.2.2.1.5 If sensors send information					x						
<b>3.2.2.2 Deactivate function</b>											
3.2.2.2.1 To stop all sensor											
3.2.2.2.2 To turn off all camera											
3.2.2.2.3 If the alarm is opened					x						
<b>3.2.4 Special conditions</b>											
3.2.4.1 When the one of function					x						
3.2.4.2 When the response is											
3.2.4.3 fail function( with message											
<b>3.3 Permission Module</b>											
<b>3.3.2 Functionality &amp; Scenario</b>											
3.3.2.1 When display sends											
3.3.2.4 If the password is not											
3.3.2.5 Control panel display											
3.3.2.7 Call the function in the					x						
<b>3.4 Phonecall Module</b>											
<b>3.4.2 Functionality &amp; Scenario</b>											
3.4.2.1 Initialize function											
3.4.2.2 Call phone function											
<b>3.5 Sensor Management Module</b>											
<b>3.5.2 Functionality &amp; Scenario</b>											
3.5.2.1 Initialize function											
3.5.2.3 Disable function											
3.5.2.4 Detect function											
<b>3.6 Camera Management Module</b>											
<b>3.6.1 Descriptions</b>											
<b>3.6.2 Functionality &amp; Scenario</b>											
3.6.2.1 Initialize function											
3.6.2.4 Disable function									x		
3.6.2.5 Data management function											
<b>3.7 Setting module( for Administrator)</b>											
<b>3.7.2 Functionality &amp; Scenario</b>											
<b>3.8 Alarming module</b>											
<b>3.8.2 Functionality &amp; Scenario</b>											
3.8.2.1 Receive the signal and											

## A.2 ‘System modules’ -> ‘External Interface Requirement’

	4	4.1	4.1.1	4.1.1.2	4.1.1.3	4.1.1.4	4.1.1.6	4.1.2	4.1.2.4	4.1.2.5		4	4.1	4.1.1	4.1.1.2	4.1.1.3	4.1.1.4	4.1.1.6	4.1.2	4.1.2.4	4.1.2.5
<b>3 System Modules</b>																					
<b>3.1 Initialization module</b>																					
<b>3.1.1 Descriptions</b>																					
<b>3.1.2 Functionality &amp; Scenario</b>																					
3.1.2.1 call the initialize function																					
3.1.2.2 call the initialize function																					
3.1.2.3 call the initialize function																					
<b>3.2 Arming Module</b>																					
<b>3.2.2 Functionality &amp; Scenario</b>																					
<b>3.2.2.1 Activate function</b>																					
3.2.2.1.1 Send signal																					
3.2.2.1.5 If sensors send information																					
<b>3.2.2.2 Deactivate function</b>																					
3.2.2.2.1 To stop all sensor																					
3.2.2.2.2 To turn off all camera																					
3.2.2.2.3 If the alarm is opened																					
<b>3.2.4 Special conditions</b>																					
3.2.4.1 When the one of function																					
3.2.4.2 When the response is																					
3.2.4.3 fail function( with message																					
<b>3.3 Permission Module</b>																					
<b>3.3.2 Functionality &amp; Scenario</b>																					
3.3.2.1 When display sends																					
3.3.2.4 If the password is not																					
3.3.2.5 Control panel display																					
3.3.2.7 Call the function in the																					
<b>3.4 Phonecall Module</b>																					
<b>3.4.2 Functionality &amp; Scenario</b>																					
3.4.2.1 Initialize function																					
3.4.2.2 Call phone function																					
<b>3.5 Sensor Management Module</b>																					
<b>3.5.2 Functionality &amp; Scenario</b>																					
3.5.2.1 Initialize function																					
3.5.2.3 Disable function																					
3.5.2.4 Detect function																					
<b>3.6 Camera Management Module</b>																					
<b>3.6.1 Descriptions</b>																					
<b>3.6.2 Functionality &amp; Scenario</b>																					
3.6.2.1 Initialize function																					
3.6.2.4 Disable function																					
3.6.2.5 Data management function																					
<b>3.7 Setting module( for Administrator)</b>																					
<b>3.7.2 Functionality &amp; Scenario</b>																					
<b>3.8 Alarming module</b>																					
<b>3.8.2 Functionality &amp; Scenario</b>																					
3.8.2.1 Receive the signal and																					

### A.3 'External Interface Requirement' -> 'System modules'

	3.1	3.1.1	3.1.2	3.1.2.1	3.1.2.2	3.1.2.3	3.2	3.2.2	3.2.2.1	3.2.2.1.1	3.2.2.1.5	3.2.2.2	3.2.2.2.1	3.2.2.2.2	3.2.2.2.3	3.2.4	3.2.4.1	3.2.4.2	3.2.4.3	3.3	3.3.2	3.3.2.1	3.3.2.4	3.3.2.5	3.3.2.7	3.4	3.4.2	3.4.2.1	3.4.2.2	3.5	3.5.2	3.5.2.1	3.5.2.3	3.5.2.4	3.6	3.6.1	3.6.2	3.6.2.1	3.6.2.4	3.6.2.5	3.7	3.7.2	3.8	3.8.2	3.8.2.1		
4. External Interface Requirements																																															
4.1 User Interfaces																																															
4.1.1 Panel																																															
4.1.1.2 Password Input																																															
4.1.1.3 Display																																															
4.1.1.4 Activation																																															
4.1.1.6 Panic button																																															
4.1.2 GUI																																															
4.1.2.4 Video Image																																															
4.1.2.5 Configuration																																															

### A.4 'External Interface Requirement' -> 'External Interface Requirement'

	4	4.1	4.1.1	4.1.1.2	4.1.1.3	4.1.1.4	4.1.1.6	4.1.2	4.1.2.4	4.1.2.5
<b>4. External Interface Requirements</b>										
<b>4.1 User Interfaces</b>										
<b>4.1.1 Panel</b>										
<b>4.1.1.2 Password Input</b>										
<b>4.1.1.3 Display</b>										
<b>4.1.1.4 Activation</b>										
<b>4.1.1.6 Panic button</b>										
<b>4.1.2 GUI</b>										
<b>4.1.2.4 Video Image</b>										
<b>4.1.2.5 Configuration</b>									x	

# SafeHome Team 4

## Requirement specification

Version 1.2

Printed by Team4

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Purpose	1
1.2	Intended Audience and Reading Suggestions	1
1.3	Project Scope	1
1.4	Contents and simple descriptions	1
<b>2</b>	<b>Overall Descriptions about product</b>	<b>3</b>
2.1	User Classes and Characteristics	3
2.2	Operating Environment	3
2.3	Design and Implementation Constraints	3
2.4	Overall Functionality Introduction	3
2.4.1	Internal System Modules	3
2.4.1.1	Initialization module	3
2.4.1.1.1	Functionality	3
2.4.1.2	Arming Module	3
2.4.1.2.1	Functionality	3
2.4.1.3	Permission Module	4
2.4.1.3.1	Functionality	4
2.4.1.4	Phonecall Module	4
2.4.1.4.1	Functionality	4
2.4.1.5	Sensor Management	4
2.4.1.5.1	Functionality	4
2.4.1.6	Camera Management	4
2.4.1.6.1	Functionality	4
2.4.1.7	Setting Module	4
2.4.1.7.1	Functionality	4

2.4.1.8	Alarming Module	4
2.4.1.8.1	Functionality	5
2.5	Assumptions and Dependencies	5
<b>3</b>	<b>System Modules</b>	<b>6</b>
3.1	Initialization module	6
3.1.1	Descriptions	6
3.1.2	Functionality & Scenario	6
3.1.2.1	call the initialize function in the sensor management module	6
3.1.2.2	call the initialize function in the camera management module	6
3.1.2.3	call the initialize function in the phonecall module	6
3.1.3	Frequency of use	6
3.1.4	Special conditions	6
3.1.5	Constraints	6
3.2	Arming Module	6
3.2.1	Descriptions	7
3.2.2	Functionality & Scenario	7
3.2.2.1	Activate function	7
3.2.2.1.1	Send signal	7
3.2.2.1.2	Receive the response signals	7
3.2.2.1.3	Wait interrupts of sensors.	7
3.2.2.1.4	Lock the panel(If the state is "stay", skip)	7
3.2.2.1.5	If sensors send interrupt signals,	7
3.2.2.2	Deactivate function	8
3.2.2.2.1	To stop all sensors, send a signal to the sensor management module.	8
3.2.2.2.2	To turn off all cameras, send a signal to the camera management module	8
3.2.2.2.3	If the alarm is operating, send a signal to the "Alarming module"	8

3.2.3	Frequency of use	8
3.2.4	Special conditions	8
3.2.4.1	When the one of functions related to this alarm module don't send the response messages in constant2 time,	8
3.2.4.2	When the response message represent that some components are not working or not ready	8
3.2.4.3	fail function( with message )	8
3.2.4.3.1	To notify the trouble, send the message to display module.	8
3.2.4.3.2	Wait the respond from control panel.	9
3.2.5	Constraints	9
3.2.5.1	HW Features	9
3.2.5.2	Constant value	9
3.3	Permission Module	9
3.3.1	Descriptions	9
3.3.2	Functionality & Scenario	9
3.3.2.1	When user using panel or accessing via inputs 4-digit password, the password is passed to Permission module.	9
3.3.2.2	Compare the 4-digit with the true password.	9
3.3.2.3	If the password is true,	9
3.3.2.4	If the password which is inputted via panel is not true, user has 2 more chance before alarming.	9
3.3.2.5	If the password which is inputted via internet is not true, user have infinite chance to retry.	9
3.3.2.6	Control panel displays number of left chances to insert password.	10
3.3.2.7	But if the password still is not true, the system considers the user as an intruder.	10
3.3.2.8	Call the function in the 3.8 and Call the function in the 3.4	10
3.3.3	Frequency of use	10
3.3.4	Special conditions	10

3.3.4.1	When the control panel management module don't send the response a signal in constant time,	10
3.3.5	Constraints	10
3.3.5.1	HW Features	10
3.3.5.2	All passwords are made up of 4 digits	10
3.4	Phonecall Module	10
3.4.1	Descriptions	10
3.4.2	Functionality & Scenario	10
3.4.2.1	Initialize function	10
3.4.2.1.1	Check the phone line.	10
3.4.2.1.2	Check the phone number list	11
3.4.2.2	Call phone function	11
3.4.3	Frequency of use	11
3.4.4	Special conditions	11
3.4.5	Constraints	11
3.4.5.1	HW Features	11
3.4.5.2	Phone number list can store at most five phone numbers.	11
3.4.5.3	Only administrator can modify phone number list or emergency signal.	11
3.5	Sensor Management Module	11
3.5.1	Descriptions	11
3.5.2	Functionality & Scenario	11
3.5.2.1	Initialize function	12
3.5.2.2	Enable function	12
3.5.2.3	Disable function	12
3.5.2.4	Detect function	12
3.5.2.5	Reconfigure function	12
3.5.3	Frequency of use	12
3.5.4	Special conditions	13

3.5.5	Constraints	13
3.5.5.1	HW Features	13
3.5.5.2	The maximum number of sensors	13
3.6	Camera Management Module	13
3.6.1	Descriptions	13
3.6.2	Functionality & Scenario	13
3.6.2.1	Initialize function	13
3.6.2.2	Password function	14
3.6.2.3	Enable function	14
3.6.2.4	Disable function	14
3.6.2.5	Data_management function	14
3.6.2.5.1	Display function	14
3.6.2.5.1.1	Normal mode	14
3.6.2.5.1.2	Thumbnail mode	14
3.6.2.5.2	Record function	14
3.6.2.5.3	Zoom_in function	14
3.6.2.5.4	Zoom_out function	15
3.6.2.5.5	Pan function	15
3.6.2.6	Reconfigure function	15
3.6.3	Frequency of use	15
3.6.4	Special conditions	15
3.6.5	Constraints	15
3.6.5.1	HW Features	15
3.6.5.2	The maximum number of cameras	15
3.6.5.3	The maximum movie recording time	15
3.7	Setting module( for Administrator )	15
3.7.1	Descriptions	16
3.7.2	Functionality & Scenario	16
3.7.2.1	Via internet, Administrator sends the new value.	16

3.7.2.2	Receive and modify the value in reference to the stored list	16
3.7.3	Frequency of use	16
3.7.4	Special conditions	16
3.7.5	Constraints	16
3.8	Alarming module	16
3.8.1	Descriptions	16
3.8.2	Functionality & Scenario	17
3.8.2.1	Receive the signal about what condition is occurred.	17
3.8.2.2	When the condition need	17
3.8.3	Frequency of use	17
3.8.4	Special conditions	17
3.8.5	Constraints	17
3.8.5.1	HW Features	17
<b>4</b>	<b>External Interface Requirements</b>	<b>18</b>
4.1	User Interfaces	18
4.1.1	Panel	18
4.1.1.1	Not Ready	18
4.1.1.2	Password Input	18
4.1.1.3	Display	18
4.1.1.4	Activation	18
4.1.1.5	Deactivation	18
4.1.1.6	Panic button	18
4.1.2	GUI	18
4.1.2.1	Access	19
4.1.2.2	Checking System status	19
4.1.2.3	Floor Plan	19
4.1.2.4	Video Image	19
4.1.2.5	Configuration	19

4.2	Hardware Interfaces	19
4.2.1	Hardware types	19
4.2.2	Connection	19
4.3	Software Interfaces	19
4.3.1	Development Environment	19
4.4	Communications Interfaces	19
4.4.1	Safehome Network	20
<b>5</b>	<b>Other Nonfunctional Requirements</b>	<b>21</b>
5.1	Safety Requirements	21
5.1.1	Loss	21
5.1.2	Damage	21
5.1.2.1	Panel	21
5.1.2.2	Camera	21
5.1.2.3	Sensor	21
5.1.3	Harm	21
5.2	Security Requirements	21
5.2.1	Qualification for accessing system	21
5.2.2	Password display management	22
5.3	Software Quality Attributes	22
5.3.1	Availability	22
5.3.2	Correctness	22
5.3.3	Flexibility	22
5.3.4	Interoperability	22
5.3.5	Maintainability	22
5.3.6	Portability	23
5.3.7	Reliability	23
5.3.8	Reusability	23
5.3.9	Robustness	23

5.3.10	Testability	23
5.3.11	Usability	23

ID	
1	<b>1 Introduction</b>
3	<b>1.1 Purpose</b>  <p>This document is Software Requirement Specification. This Document is one of the development documents of Safehome system which is made in early stage, using Wireless 802.11 Box. This paper covers Safehome Software which will be developed by Team4, in 2008. This requirement basically aims to be a stepping stone to go on to the next software development stage, and aims to state specifically, and obviously about customer's needs.</p>
5	<b>1.2 Intended Audience and Reading Suggestions</b>  <p>For hardware Team, '2. Overall description' and '4. External Interface Requirement' are more useful for their development than other specific functionality in '3. System Modules'. If you who read this paper are customers, we would like for you to focus on the context of 'Overall descriptions' and the description in 'System Features'. The detailed functionality may not be needed.</p> <p>At the last part, "5. Other nonfunctional requirement", we mention the nonfunctionality, which is for the safety of users, security and the focusing attributes. If you read the last part, you can be aware of what attribute is important in this project.</p>
6	<b>1.3 Project Scope</b>  <p>SafeHome system is composed of Control Panel, System, several components connected to wireless hardware box. The purpose of this project is to implement software which maintains and controls all the components.</p>
289	<b>1.4 Contents and simple descriptions</b>  <p>1. Introduction</p> <p>2. Overall Descriptions</p> <p>: There are overall descriptions about this project and this paper.</p> <p>So there are simple descriptions about functionality.</p> <p>And this part defines the user classes and Operating Environment.</p> <p>3. System Modules.</p>

ID	
289	<p>The program unit in the system is called "Module" in this project. This part specifies the functionality, scenario, constraints, exceptions and so on.</p> <p>4. External Interface Requirements</p> <p>There are some interfaces. a Panel, Internet and so on.</p> <p>This part specifies the simple description about HW.</p> <p>This part is not complete, but this paper is for Software development. So we don't need to think about Hardware deeply.</p> <p>5. Other Nonfunctional Requirements</p> <p>We consider nonfunctional requirement like safety and security in this part.</p> <p>And some quality attributes we are focusing on are shown to make software worthy.</p>

ID	
8	<b>2 Overall Descriptions about product</b>
17	<b>2.1 User Classes and Characteristics</b>  User Classes: Safehome's user is mainly house owner, and members of owner's family. Characteristic: Safehome's users want to keep their home safe.
18	<b>2.2 Operating Environment</b>  Safehome users are basically given hardware box, so there is no need to give specific operating environment to control panel. But, administrator can access Safehome system via internet. In this case, the Operating Environment is Window OS.
19	<b>2.3 Design and Implementation Constraints</b>  This project is based on the HW wireless box. The only reason that our company develops this product is making money using wireless Box. So we emphasize and make full use of this HW. So we, software team, must think the HW box in SW design step.
16	<b>2.4 Overall Functionality Introduction</b>
230	<b>2.4.1 Internal System Modules</b>
232	<b>2.4.1.1 Initialization module</b>  When the system is powered on, check whether all the components are fine or not.
253	<b>2.4.1.1.1 Functionality</b>  Initialize function.
233	<b>2.4.1.2 Arming Module</b>  This module is for switching the state between activating and deactivating arming.
254	<b>2.4.1.2.1 Functionality</b>  Activate, Deactivate, Fail

ID	
234	<b>2.4.1.3 Permission Module</b>  this module checks the password is correct.
255	<b>2.4.1.3.1 Functionality</b>  Check password
236	<b>2.4.1.4 Phonecall Module</b>  In a state of emergency, this module inform of the emergency.
257	<b>2.4.1.4.1 Functionality</b>  Initialize function, Call phone function
237	<b>2.4.1.5 Sensor Management</b>  Manage the sensors connected to the Safehome system.
258	<b>2.4.1.5.1 Functionality</b>  Initialize function, enable function, disable function, detect function, reconfigure
238	<b>2.4.1.6 Camera Management</b>  Manage the cameras connected to the Safehome system.
259	<b>2.4.1.6.1 Functionality</b>  Initialize function, enable function, disable function, data_management function, reconfigure function.
239	<b>2.4.1.7 Setting Module</b>  For only administrator, provide modifying the constant values.
260	<b>2.4.1.7.1 Functionality</b>  Modify function
240	<b>2.4.1.8 Alarming Module</b>  This module is for the sounds and alerts

ID	
261	<b>2.4.1.8.1 Functionality</b>  Sound function
21	<b>2.5 Assumptions and Dependencies</b>  "Module" is the program unit that conduct some functions that be grouped by similarity.

ID	
9	<b>3 System Modules</b>
242	<b>3.1 Initialization module</b>
243	<b>3.1.1 Descriptions</b>  <p>In safehome system, there are so many components and module. When the system is powered on, it had better check whether all the components are fine or not. If the one of component is failed to initialize, then system notify the user or the administrator to fix it. And it never turns on until the problem is fixed. Only administrator can try to turn on or off the system.</p>
244	<b>3.1.2 Functionality &amp; Scenario</b>
245	<b>3.1.2.1</b> call the initialize function in the sensor management module  <p>If the return signal is fail signal, display what sensor is failed. And the system cancels starting.</p>
251	<b>3.1.2.2</b> call the initialize function in the camera management module  <p>If the return signal is fail signal, display what camera is failed. And the system cancels starting.</p>
252	<b>3.1.2.3</b> call the initialize function in the phonecall module  <p>If the return signal is fail signal, display "phonecall management module is failed". And the system cancels starting.</p>
248	<b>3.1.3 Frequency of use</b>  <p>Very low</p>
249	<b>3.1.4 Special conditions</b>  <p>None</p>
250	<b>3.1.5 Constraints</b>  <p>None</p>
87	<b>3.2 Arming Module</b>

ID	
88	<b>3.2.1 Descriptions</b>  <p>Safehome system has two types of sensors; First-type of sensors sticks to doors, and windows. And second-type of sensors is motion detector. If all sensors are functioning, the system is in activated state. And if not, the system is deactivated. And these states need function which alters states.</p> <p>This module is for switching the state between activating and deactivating arming.</p>
89	<b>3.2.2 Functionality &amp; Scenario</b>  <p>There is two state of activation. One is "stay", which only the camera works. The other is "away" which all the components work.</p>
273	<b>3.2.2.1 Activate function</b>  <p>: Switch the state to the other state that sensors are working for detect the dangerous factor.</p>
274	<b>3.2.2.1.1 Send signal</b>  <p>If the state that user wants is "stay", conduct only #2.  If the previous state is "stay", don't need to conduct #2.</p> <ol style="list-style-type: none"> <li>1. To enable all sensors, send a signal to the sensor management module.</li> <li>2. To turn on all cameras, send a signal to the camera management module.</li> </ol>
277	<b>3.2.2.1.2 Receive the response signals</b>
278	<b>3.2.2.1.3 Wait interrupts of sensors.</b>  <p>If the state that user wants is "stay", there is no need to wait.  Wait 1 minute because the user can't go out right after the mode is started.</p>
310	<b>3.2.2.1.4 Lock the panel(If the state is "stay", skip)</b>
279	<b>3.2.2.1.5 If sensors send interrupt signals,</b>  <p>Detected motions can be from Safehome users who want to deactivate, system should not send signal to alarm on the spot.</p> <ol style="list-style-type: none"> <li>1. Classify signals <ol style="list-style-type: none"> <li>1-1. When door sensors send signal, Wait for deactivate in constant1 time.</li> </ol> </li> </ol>

ID	
279	<p>1-2. After waiting time, follow next step.</p> <p>2. Send signal to the 'phoncall module' and 'alarming module' on the spot</p>
163	<p><b>3.2.2.2 Deactivate function</b></p> <p>: Switch to the state that sensors are not working and turn off all the sensors or "stay" state</p>
165	<b>3.2.2.2.1</b> To stop all sensors, send a signal to the sensor management module.
166	<b>3.2.2.2.2</b> To turn off all cameras, send a signal to the camera management module
281	<b>3.2.2.2.3</b> If the alarm is operating, send a signal to the "Alarming module"
91	<p><b>3.2.3 Frequency of use</b></p> <p>Very high</p>
92	<b>3.2.4 Special conditions</b>
167	<p><b>3.2.4.1</b> When the one of functions related to this alarm module don't send the response messages in constant2 time,</p> <p>1. Resend the signal three times.</p> <p>2. If the module doesn't respond, call the 'fail function' with the message, what kind of module</p>
168	<p><b>3.2.4.2</b> When the response message represent that some components are not working or not ready</p> <p>1. To be aware of the components that are not working, send a signal again</p> <p>2. Receive a message about what kind of components is not working.</p> <p>3. Display 'not ready'</p>
169	<p><b>3.2.4.3 fail function( with message )</b></p> <p>: Notify the user that the some parts of system are not working. And ask the user whether he or she turns on or off the alarm module.</p>
170	<b>3.2.4.3.1</b> To notify the trouble, send the message to display module.

ID	
171	<b>3.2.4.3.2</b> Wait the respond from control panel.  2-1. If the respond is "yes", continue the alarm activation function. 2-2. If the respond is "no", call deactivate function.
93	<b>3.2.5 Constraints</b>
290	<b>3.2.5.1</b> HW Features  display, control panel.
172	<b>3.2.5.2</b> Constant value  The waiting time for response signals from the modules related to alarm module is 0.1 second.
13	<b>3.3 Permission Module</b>
30	<b>3.3.1 Descriptions</b>  When user wants to log in the Safehome control panel, he or she must insert 4-digit password. When he or she inputs the password, this module checks the password is correct. And return result to Arming Module.
37	<b>3.3.2 Functionality &amp; Scenario</b>
181	<b>3.3.2.1</b> When user using panel or accessing via inputs 4-digit password, the password is passed to Permission module.
182	<b>3.3.2.2</b> Compare the 4-digit with the true password.
185	<b>3.3.2.3</b> If the password is true,  return result to Arming module.
188	<b>3.3.2.4</b> If the password which is inputted via panel is not true, user has 2 more chance before alarming.
317	<b>3.3.2.5</b> If the password which is inputted via internet is not true, user have infinite chance to retry.

ID	
189	<b>3.3.2.6</b> Control panel displays number of left chances to insert password.
197	<b>3.3.2.7</b> But if the password still is not true, the system considers the user as an intruder.
198	<b>3.3.2.8</b> Call the function in the 3.8 and Call the function in the 3.4
33	<b>3.3.3 Frequency of use</b>  Very high
35	<b>3.3.4 Special conditions</b>
307	<b>3.3.4.1</b> When the control panel management module don't send the response a signal in constant time,  1. Resend the signal 3 times. 2. If the module doesn't respond, call the 'fail function'
32	<b>3.3.5 Constraints</b>
293	<b>3.3.5.1</b> HW Features  Control panel.
200	<b>3.3.5.2</b> All passwords are made up of 4 digits
45	<b>3.4 Phonecall Module</b>
46	<b>3.4.1 Descriptions</b>  In a state of emergency, system should inform to user or a police station. When those dangerous situation comes up, Safehome automatically calls to 119, Safehome development company, and user.
47	<b>3.4.2 Functionality &amp; Scenario</b>
265	<b>3.4.2.1</b> Initialize function
267	<b>3.4.2.1.1</b> Check the phone line.  1. If the phone line is not connected or not answered, return fail 2. Else return success

ID	
269	<b>3.4.2.1.2 Check the phone number list</b>  1. If the list is empty, return fail 2. Else return success
263	<b>3.4.2.2 Call phone function</b>  1. 'phonecall module' receives a signal from 'password module', 'sensor module' or 'display module(panic button). other modules call this module with information about what kind of emergency is occurred. 2. Call the phone numbers listed in the stored list. Call phone. 3. Send the emergency message
49	<b>3.4.3 Frequency of use</b>  Not high. But very significant.
50	<b>3.4.4 Special conditions</b>  None
51	<b>3.4.5 Constraints</b>
294	<b>3.4.5.1 HW Features</b>  Phone line, the storage that stored the phone list.
218	<b>3.4.5.2</b> Phone number list can store at most five phone numbers.
219	<b>3.4.5.3</b> Only administrator can modify phone number list or emergency signal.
66	<b>3.5 Sensor Management Module</b>
67	<b>3.5.1 Descriptions</b>  Safehome system has two types of sensors. One detects an opening or closing of window or door, and one detects motions in the house. Safehome system needs to manage sensors.  So here is Sensor management module.
68	<b>3.5.2 Functionality &amp; Scenario</b>

ID	
212	<p><b>3.5.2.1 Initialize function</b></p> <p>Initialize all the sensors.</p> <p>Scan wireless components, and compare this result with the information stored in this module</p> <ol style="list-style-type: none"> <li>1. If the result from scanning is same as the stored information, return success.</li> <li>2. If the result from scanning is not same as the stored information, return fail.</li> </ol>
213	<p><b>3.5.2.2 Enable function</b></p> <p>Enable all sensors or one sensor. Each sensor has enable() function. When all sensors need to be enabled, 'Sensor management module' calls enable() functions in sensors.</p> <p>If User wants to turn on one or several sensors via internet, this module can call each enable function.</p>
214	<p><b>3.5.2.3 Disable function</b></p> <p>Disable all sensors or one sensor. Each sensor has enable() function. When all sensors need to be enabled, 'Sensor management module' calls enable() functions in sensors.</p> <p>If User wants to turn on one or several sensors via internet, this module can call each enable function.</p>
215	<p><b>3.5.2.4 Detect function</b></p> <p>Detect motions, and send signal.</p> <ol style="list-style-type: none"> <li>1. Window detection: send signal to 'Arming module'</li> <li>2. Door detection: send signal to 'Arming module'</li> <li>3. In house motion detection: send signal to 'Alarm module'</li> </ol>
216	<p><b>3.5.2.5 Reconfigure function</b></p> <p>Modify sensor list - all sensors are saved in a sensor list.</p>
70	<p><b>3.5.3 Frequency of use</b></p> <p>Very High</p>

ID	
71	<b>3.5.4 Special conditions</b>  1. On the processing of 'initialize function', when a sensor doesn't react : return fail signal. 2. On the processing of 'Enable function', when a sensor is not enabled : send enable sensor fail signal to 'Alarming module' 3. On the processing of 'Disable function', when a sensor is not disabled : ignore interrupts from that sensor, and display what sensor broke down.
72	<b>3.5.5 Constraints</b>
295	<b>3.5.5.1 HW Features</b>  802.11 wireless HW box, sensors.
297	<b>3.5.5.2 The maximum number of sensors</b>  The maximum number of sensors which managed by 'Sensor management module' is
59	<b>3.6 Camera Management Module</b>
60	<b>3.6.1 Descriptions</b>  In the house under the control of safehome system, there are many cameras which photograph the living room, each personal room, and so on. These cameras are controlled by the wireless HW box. The cameras have the function about zoom in/zoom out/panning/tilting. And the data transferred between safehome system and each component when the user needs to see the scene of camera. This module is a management system that manages all above functionality.
61	<b>3.6.2 Functionality &amp; Scenario</b>
190	<b>3.6.2.1 Initialize function</b>  Initialize all the cameras. Scan wireless components, and compare the result with the information stored in this module 1. If the result from scanning is same as the stored information, return success signal. 2. If the result from scanning is not same as the stored information, return fail signal.

ID	
313	<b>3.6.2.2 Password function</b>  User needs to input password when user selects each camera. User continues to input password, even if the password entered before was wrong.
194	<b>3.6.2.3 Enable function</b>  Enable all cameras. Each camera has enable() function. When all cameras need to be enabled, 'Camera management module' calls enable() functions in cameras.
195	<b>3.6.2.4 Disable function</b>  Disable all cameras. Each camera has enable() function. When all cameras need to be enabled, 'Camera management module' calls enable() functions in cameras.
196	<b>3.6.2.5 Data_management function</b>
202	<b>3.6.2.5.1 Display function</b>  Display pictures what camera photographs.
314	<b>3.6.2.5.1.1 Normal mode</b>  It's a common mode. User can see only one camera.
316	<b>3.6.2.5.1.2 Thumbnail mode</b>  User can see several camera views at a time. So user can see all the interior of user's house. The windows is divided into some camera view windows, and all the sub-windows have each own camera view.  User can select one camera view in this mode. If user choose one view among this sub-windows, then viewing mode is switched to normal mode.
203	<b>3.6.2.5.2 Record function</b>  Record pictures from camera. 'Camera management module' stores movie of constant time.
204	<b>3.6.2.5.3 Zoom_in function</b>  Camera zooms in current picture.

ID	
206	<b>3.6.2.5.4 Zoom_out function</b>  Camera zooms out current picture.
209	<b>3.6.2.5.5 Pan function</b>  Camera moves horizontally.
199	<b>3.6.2.6 Reconfigure function</b>  Modify camera list - all cameras are saved in a camera list.
63	<b>3.6.3 Frequency of use</b>  Very high
64	<b>3.6.4 Special conditions</b>  1. On the processing of 'initialize function', when a camera doesn't react : return fail signal. 2. On the processing of 'Enable function', when a camera is not enabled : send enable fail signal to control panel. 3. On the processing of 'Disable function', when a camera is not disabled : send disable fail signal to control panel.
299	<b>3.6.5 Constraints</b>
300	<b>3.6.5.1 HW Features</b>  802.11 wireless HW box, cameras, storage that store the recoding information.
301	<b>3.6.5.2 The maximum number of cameras</b>  The maximum number of cameras which managed by 'Camera management module' is
302	<b>3.6.5.3 The maximum movie recording time</b>  The maximum movie recording time is 1 week.
52	<b>3.7 Setting module( for Administrator )</b>

ID	
53	<b>3.7.1 Descriptions</b>  <p>In this Safehome system, there are many constant values( ex. 3.2.6, and Constraints ). Somebody needs to set up and modify these values. But if anybody can modify these values, then the system can be confused. So only administrator via only internet can set up and modify values.</p> <p>This module is for such controls.</p>
54	<b>3.7.2 Functionality &amp; Scenario</b>  <p>This module manages the list of the constant values. The list informs where the value is, what the value is, and so on.</p>
183	<b>3.7.2.1</b> Via internet, Administrator sends the new value.
186	<b>3.7.2.2</b> Receive and modify the value in reference to the stored list
56	<b>3.7.3 Frequency of use</b>  <p>Very low.</p>
57	<b>3.7.4 Special conditions</b>  <p>None</p>
58	<b>3.7.5 Constraints</b>  <p>There is a list stored about constant values. The list should be created at the time of delivery.</p>
108	<b>3.8 Alarming module</b>
109	<b>3.8.1 Descriptions</b>  <p>In the emergency situation, which need to sound a warning, Safehome system warn the urgency of situation using some External Hardware that sounds. Also, when the user goes into the house that security function is activated and input the password, a beep sounds until the password is correct.</p> <p>This module is for the sounds and alerts. When other module call this module and send a signal about what condition is occurred, this module makes the related Hardware sound.</p>

ID	
110	<b>3.8.2 Functionality &amp; Scenario</b>
179	<b>3.8.2.1</b> Receive the signal about what condition is occurred.
178	<b>3.8.2.2</b> When the condition need <ul style="list-style-type: none"> <li>1. Beep sound Send a signal to the HW related to beep sound.</li> <li>2. The serious emergency Send a signal to the HW related to alarm sounds.</li> </ul>
112	<b>3.8.3 Frequency of use</b> <p>The Frequency is various on the kind of sounds. The beep sound is used at many times, but the serious emergency is hardly occurred.</p>
113	<b>3.8.4 Special conditions</b> <p>None</p>
114	<b>3.8.5 Constraints</b>
303	<b>3.8.5.1</b> HW Features <p>Beep sound, alarm sound</p>

ID	
10	<b>4 External Interface Requirements</b>
22	<b>4.1 User Interfaces</b>
75	<b>4.1.1 Panel</b>
122	<b>4.1.1.1 Not Ready</b>  When user tries to activate the system, the panel asks system whether system is ready or not. if system is "not ready", panel should show user that system is not ready and should not activate the system. ("not ready" implies that a sensor is open; i.e., that a door or window is open.)
77	<b>4.1.1.2 Password Input</b>  In panel, there are two states. One state is only for password, in other words locked state. The other is non-locked state. User can do everything. User should be able to enter a password to be allowed to use all other functions in activation state. User can input only 4-digit for a password.
78	<b>4.1.1.3 Display</b>  Panel should display the status of a security zone and the status of sensors.
79	<b>4.1.1.4 Activation</b>  User can select "Stay" or "Away" to activate the system. "Stay" activates only perimeter sensors (inside motion detecting sensors are deactivated). "Away" activates all sensors. When activation occurs, a red alarm light can be observed by the homeowner.
123	<b>4.1.1.5 Deactivation</b>  Panel should provide key to deactivate system when user inputs correct password.
282	<b>4.1.1.6 Panic button</b>  On panel, there is a panic button which is used in emergency situation. If panic button is pressed when system is deactivated, system should deal it with Phonecall module.
76	<b>4.1.2 GUI</b>

ID	
127	<b>4.1.2.1 Access</b>  On GUI, user should enter an ID and password to access remotely.
133	<b>4.1.2.2 Checking System status</b>  User should be able to check system status through GUI.
124	<b>4.1.2.3 Floor Plan</b>  GUI should show floor plan and sensor locations to user. Panel should display zones on floor plan which user can change.
130	<b>4.1.2.4 Video Image</b>  User should be able to select video camera for viewing video images, pan or zoom the video camera.
280	<b>4.1.2.5 Configuration</b>  When administrator of system is log on, GUI should provide configuration mode to user. User can use setting module on this mode to change settings for system.
23	<b>4.2 Hardware Interfaces</b>
74	<b>4.2.1 Hardware types</b>  The hardware types used in system are camera, sensor, panel, alarm, and so on
73	<b>4.2.2 Connection</b>  There should be secure connections between system, panel, pc, and other devices.
24	<b>4.3 Software Interfaces</b>
125	<b>4.3.1 Development Environment</b>  We will use Java to implement this project.
25	<b>4.4 Communications Interfaces</b>

ID	
126	<p><b>4.4.1 Safehome Network</b></p> <p>Safehome Network is based on 802.11 wireless network. All connections should be secure so that all people cannot know the detail of status through abnormal route which is not intended by development team.</p>

ID	
11	<b>5 Other Nonfunctional Requirements</b>
27	<b>5.1 Safety Requirements</b>
131	<b>5.1.1 Loss</b>  Panel, cameras, sensors should be placed their own point and be fixed. So, there will be no loss for them.
150	<b>5.1.2 Damage</b>  If any of following components is damaged, user cannot be served fully. We should give users the contact service for repairing, and repair components by visiting.
226	<b>5.1.2.1 Panel</b>  1. Whoever at home can control the panel. 2. Whoever at home can access the system through the panel at first. 3. Panel has possibility to be most frequently damaged. 4. Materials of panel should be concerned for durability of panel.
227	<b>5.1.2.2 Camera</b>  1. Cameras will be placed on high and untouchable position. 2. There is no expecting damages to cameras while using them.
228	<b>5.1.2.3 Sensor</b>  1. Sensors will be placed in invisible points. 2. There is no expecting damages to sensors while using them.
151	<b>5.1.3 Harm</b>  Cameras and sensors will be placed untouchably or invisibly. So, we need to concern only about the shape and materials of panel.  Panel need to be made of anything except metals to prevent an electric shock.
28	<b>5.2 Security Requirements</b>
152	<b>5.2.1 Qualification for accessing system</b>  The only user who is permitted by password can access the system.

ID	
285	<p><b>5.2.2 Password display management</b></p> <p>When user input 4-digit password on panel, the password should not be shown on display.</p> <p>When user input password on GUI, the user can read "*" mark how many numbers are put with the numbers of the mark, so that user know she/he is putting well.</p>
29	<p><b>5.3 Software Quality Attributes</b></p>
144	<p><b>5.3.1 Availability</b></p> <p>The most simple representation for availability is as a ratio of the expected value of the uptime of a system to the aggregate of the expected values of up and down time.</p>
143	<p><b>5.3.2 Correctness</b></p> <p>Correctness of an algorithm is asserted when it is said that the algorithm is correct with respect to a specification. Functional correctness refers to the input-output behavior of the algorithm (i.e., for each input it produces the correct output).</p>
142	<p><b>5.3.3 Flexibility</b></p> <p>Flexibility is the popular term for the ability to easily bend an object or the ability to adapt to different circumstances.</p>
141	<p><b>5.3.4 Interoperability</b></p> <p>Interoperability is a property referring to the ability of diverse systems and organizations to work together (inter-operate).</p>
140	<p><b>5.3.5 Maintainability</b></p> <p>The term maintainability has the following meanings:</p> <ol style="list-style-type: none"> <li>1. A characteristic of design and installation, expressed as the probability that an item will be retained in or restored to a specified condition within a given period of time, when the maintenance is performed in accordance with prescribed procedures and resources.</li> <li>2. The ease with which maintenance of a functional unit can be performed in accordance with prescribed requirements.</li> </ol>

ID	
270	<p><b>5.3.6 Portability</b></p> <p>Portability is the general characteristic of being readily transportable from one location to another, and may specifically refer to:</p> <ol style="list-style-type: none"> <li>1. Portability (software), the portability of a piece of software to multiple platforms</li> <li>2. Porting, the ability of a program to be ported from one system to another in computer science.</li> </ol>
271	<p><b>5.3.7 Reliability</b></p> <p>Reliability (systemic def.) is the ability of a person or system to perform and maintain its functions in routine circumstances, as well as hostile or unexpected circumstances.</p>
272	<p><b>5.3.8 Reusability</b></p> <p>Reusability is the likelihood a segment of source code can be used again to add new functionalities with slight or no modification. Reusable modules and classes reduce implementation time, increase the likelihood that prior testing and use has eliminated bugs and localizes code modifications when a change in implementation is required.</p>
136	<p><b>5.3.9 Robustness</b></p> <p>Robustness is the quality of being able to withstand stresses, pressures, or changes in procedure or circumstance.</p>
135	<p><b>5.3.10 Testability</b></p> <p>Testability, a property applying to an empirical hypothesis, involves two components:</p> <ol style="list-style-type: none"> <li>1. The logical property that is variously described as contingency, defeasibility, or falsifiability, which means that counterexamples to the hypothesis are logically possible.</li> <li>2. The practical feasibility of observing a reproducible series of such counterexamples if they do exist.</li> </ol>
134	<p><b>5.3.11 Usability</b></p> <p>Usability is a term used to denote the ease with which people can employ a particular tool or other human-made object in order to achieve a particular goal. Usability can also refer to the methods of measuring usability and the study of the principles behind an</p>

ID	
134	object's perceived efficiency or elegance.