THE SmartHome

Software Requirements Specification:

Visual Paradigm 15.2

0.5

2/12/2019

Alex Sanchez, Anuj Patel, Onyeka Mordi, Scotty Rodriguez

Prepared for

CSU-CSCI3320—Software Engineering Principles I

Instructor: Junfeng Qu, Ph.D.

Spring 2019

# **Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 2/12/19 | Version 0.5 | Scotty Rodriguez | 1.0 |
| 2/19/19 | Version 0.5 | Anuj Patel | 1.1, 1.2, 1.3, 1.4, 1.5, 2.3, 2.4, 2.5, 3.5, 3.2.1, 3.2.2, 3.2.3, 3.2.4, 3.2.5, 3.2.6, 3.3.1, 3.3.2, 3.3.3, 3.3.4, 3.3.5, 3.3.6, 5.1, 5.2, 5.3, 5.4, 5.5, 5.6 |
| 2/24/19 | Version 0.5 | Alex Sanchez | 3.2.7, 3.3.7,5.7,3.2.13-16, 5.13-16, |
| 2/25/19 | Version 0.5 | Scotty Rodriguez | 1.0, 1.3, 1.4, 2, 2.1, 2.1.1.1, 2.1.1.2, 2.1.1.3, 2.1.2, 2.2, 2.2.1-2.2.8, 3.2, 3.2.8, 3.3.7, 5.8 |
| 2/26/19 |  | Scotty Rodriguez | 3.3, 3.3.17-3.3.19, 5.17, 5.18, 5.19 |
|  |  |  |  |

# **Document Approval**

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | <Your Name> | Lead Software Eng. |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

**REVISION HISTORY II**

**DOCUMENT APPROVAL II**

**1. INTRODUCTION 1**

1.1 Purpose 1

1.2 Scope 1

1.3 Definitions, Acronyms, and Abbreviations 1

1.4 References 1

1.5 Overview 1

**2. GENERAL DESCRIPTION 2**

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Characteristics 2

2.4 General Constraints 2

2.5 Assumptions and Dependencies 2

**3. SPECIFIC REQUIREMENTS 2**

3.1 External Interface Requirements 3

*3.1.1 User Interfaces 3*

*3.1.2 Hardware Interfaces 3*

*3.1.3 Software Interfaces 3*

*3.1.4 Communications Interfaces 3*

3.2 Functional Requirements 3

*3.2.1 <Functional Requirement or Feature #1> 3*

*3.2.2 <Functional Requirement or Feature #2> 3*

3.3 Use Cases 3

*3.3.1 Use Case #1 4*

*3.3.2 Use Case #2 5*

3.4 Classes / Objects 5

*3.4.1 <Class / Object #1> 6*

*3.4.2 <Class / Object #2> 6*

3.5 Non-Functional Requirements 6

*3.5.1 Performance 6*

*3.5.2 Reliability 6*

*3.5.3 Availability 6*

*3.5.4 Security 6*

*3.5.5 Maintainability 6*

*3.5.6 Portability 6*

3.6 Inverse Requirements 6

3.7 Design Constraints 7

3.9 Other Requirements 7

**4. ANALYSIS MODELS 7**

4.1 Sequence Diagrams 7

**5. UI DESIGN 8**

**6. CHANGE MANAGEMENT PROCESS 9**

**7. APPENDICES 9**

A.0 Glossary 9

A.1 Appendix 1 //If any others 9

A.2 Appendix 2 9

# **1.0 Introduction**

*The goal is to have All devices in your home communicating with each other to monitor and control your smart home in a growing industry of Internet of Things (IOT). Some benefits of having a smart home include, energy saving applications, enhancing security, comfort of your home, and maintaining your health better.*

*Examples of energy efficiency will be controlling indoor climates and electricity usage*

*by having your devices switch off appliances, control the temperature, and shut off water.*

*Having the ability to view cameras around your house, turning off the security system, opening the door, and speaking to guest will be a tremendous upgrade from your traditional home security system.*

## **1.1 Purpose**

*The purpose of this ‘Smart Home’ is to control your ‘home’ from a single device with ease. This document will focus on the specifications for THE Smart Home, including the description on user interface, sensors, appliances, and cameras . The focus of this document is on appliances, electronics, lighting, and security. The functional and non-functional requirements are included. The audience is intended for all tech-savvy, smart device users; however, the main audience is Dr. Qu.*

## **1.2 Scope**

*This subsection should:*

*(1) THE Smart Home. It will let the user control the smart devices the user connects to the software for the users control device*

*(2) This product will let the user control smart devices with a touch of a button. It will not allow the user to control devices that isn’t smart or isn’t connected through the software*

*(3) The benefits of this software would be it would be able to control your whole house (the smart devices the user has) within just one application. The reason this would be beneficial is because the response time (once the user submits what they would like to change) would be less than 10 seconds unless their internet connect is bad*

## **1.3 Definitions, Acronyms, and Abbreviations**

1. *IOT - Internet of Things*
2. *GUI - Graphical User Interface*
3. *AC - Air Conditioner*
4. *SRS - Software Requirements Specification*
5. *ADL - Assistance Daily Living*
6. *Cybercriminals - are individuals or teams of people who use technology to commit malicious activities on digital systems or networks with the intention of stealing sensitive company information or personal data, and generating profit.*

## **1.4 References**

*This subsection should:*

*(1) Software Engineering - A Practitioner’s Approach Eighth Edition By: Roger S. Pressman, Bruce R. Maxim*

*(2) McGraw Hill Education*

*(3) Helped us understand the concepts to develop the use cases and concepts*

*(4) Symantec employee. January 2019. 12 Tips to Secure your Smart Home and IOT Devices. retrieved from:* [*https://us.norton.com/internetsecurity-iot-smart-home-security-core.html*](https://us.norton.com/internetsecurity-iot-smart-home-security-core.html)

*(5) Siemens. Synco Living. 2019. Retrieved from:* [*https://www.buildingtechnologies.siemens.com/bt/global/en/buildingautomation-hvac/hvac-products/home-automation-system-synco-living/about-synco-living/Pages/About-synco-living.aspx*](https://www.buildingtechnologies.siemens.com/bt/global/en/buildingautomation-hvac/hvac-products/home-automation-system-synco-living/about-synco-living/Pages/About-synco-living.aspx)

*(6) Joy Wong Daniels, Caroline Conner, Nate Cox, Emilio Passi, Ken Olewiler. A Punchout Perspective. 2019. Retrieved from:* [*https://www.punchcut.com/perspectives/unlocking-the-connected-home-smart-home-research/*](https://www.punchcut.com/perspectives/unlocking-the-connected-home-smart-home-research/)

## **1.5 Overview**

*This subsection should:*

*(1) The rest of this SRS contains the Products functions and constraints, External interfaces, Functional requirements, Use cases, Class/object, Non-functional requirements, Models, UI designs, and updated information.*

*(2) This information is organized in seven main categories: Introduction, General description, Specific requirements, Analysis models, UI designs, Change management process, and an Appendices.*

# **2. General Description**

*The SRS of smart home will be the product and guide of technology being the root of assisting living. We will create use case diagrams and interfaces that will assist in ADL’s and which essential be guidelines and will be useful tool for researchers. We will gain experience on how to implement a smart home during the course of this project. In the system design we will be able to see the different actors and the goals for each actor.*

## **2.1 Product Perspective**

There is some related research for Smart Home ADL products

2.1.1.1 THE RENAISSANCE MILLENNIAL:

Is a Nest Learning Thermostat. When the temperature is high outside, you are able to adjust the AC using the Nest iPhone app when you on the way home, so it’s nice and cool when you get inside the house. It has built in AI that has the ability to adapt to unique behavior patterns through a learning algorithm and is optimistic about the company’s extension to other smart home products.

2.1.1.2 Doorbell Camera Device:

Using your cell phone to monitor who is at your house. When someone rings the doorbell, the owners can either hear it or feel their cell phone vibrating and can open their phone and see who is at the door through a live video feed from a built in camera from the device. There is also a built in speaker to speak to the person at the door.

2.1.1.3 Siemens Synco Living:

Energy efficient home automation system. It can regulate, control, switch on/off and display anything that makes life easier. Can control individual room temperature. Domestic water regulation. Control blinds and lighting all with an user interface.

2.1.2 Concepts of Operations

The concept of the smart home should work as simple as the customer provides an input of a command and the system interprets that input and provide the correct output. Such as “open the window” then the system will open the window. The Smart Home System will automate task through the configuration.

The system will act as the hub or the connection between all of the interconnected devices and the customer. It will have a interface that is easy for the user to select inputs and takes the inputs and send that information to the device requested.

On the development side it should be a source of research of different ideas to learn and develop a functional Smart Home System. We should be able to successfully complete this project if we focus it on the customer needs, and following through on the plans.

## **2.2 Product Functions**

2.2.1 Logging into the interface:

This functional requirement is intended to allow the customer to login into their smart home securely.

2.2.2 Option Interface:

customer to choose between if they would like to leave feedback on the system or to select a Smart Devices to control

2.2.3 Feedback

The ability to leave feedback of the system

2.2.4 Smart Devices selection

Select smart device that you wish to configure/control

2.2.5 Temperature Control

Controls the temperature of the room you would like to control

2.2.6 Sensor Selection

intended to allow the system to automatically check all the sensors (fire, gas, etc.)

2.2.7 Lighting Control

allow the customer to control their lights within their whole house.

2.2.8 Security Control

Customer would be able to function the security system through traditional interfacing. Will be able to select which security device it would like to control

## **2.3 User Characteristics**

* *User/market demographics*
* *User environment* 
  + *Developers environment*
    - *workstations*
  + *closed labs*
    - *closed server*
  + *Smart Home*
    - *general rooms they would like to control*
      * *individual devices within the room*
* *User Needs/Requirements*
  + *Network*
  + *devices*

## **2.4 General Constraints**

* *budget* 
  + *equipment, servers, and appliance that must be purchased*
* *Expertise knowledge with technology and research to improve software*
* *time*
  + *initial requirements*
    - *equipment, servers, and appliance*
  + *framework development*

## **2.5 Assumptions and Dependencies**

* + *we have interface board that we have control of*
  + *users control device is connected to our central server*
  + *wireless connection are throughout the house*
  + *sensors are installed*
    - *for precautions and warnings*
  + *we have GUI application and database to manage and store information*
  + *data we have is sequential to operate drivers*
  + *system runs all the time*
  + *software is dependent on the hardware and OS*

# **3. Specific Requirements**

*This will be the largest and most important section of the SRS. The customer requirements will be embodied within Section 2, but this section will give the D-requirements that are used to guide the project’s software design, implementation, and testing.*

*Each requirement in this section should be:*

* *Correct*
* *Traceable (both forward and backward to prior/future artifacts)*
* *Unambiguous*
* *Verifiable (i.e., testable)*
* *Prioritized (with respect to importance and/or stability)*
* *Complete*
* *Consistent*
* *Uniquely identifiable (usually via numbering like 3.4.5.6)*

*Attention should be paid to the carefully organize the requirements presented in this section so that they may easily accessed and understood. Furthermore, this SRS is not the software design document, therefore one should avoid the tendency to over-constrain (and therefore design) the software project within this SRS.*

## **3.1 External Interface Requirements**

### **3.1.1 User Interfaces**

### **3.1.2 Hardware Interfaces**

### **3.1.3 Software Interfaces**

### **3.1.4 Communications Interfaces**

## **3.2 Functional Requirements**

*This section describes specific features of the software project. If desired, some requirements may be specified in the use-case format and listed in the Use Cases Section.*

*Actor goal table*

|  |  |
| --- | --- |
| *Actor* | *Goal* |
| *Customer* | *Login* |
|  | *leave Feedback* |
|  | *add/delete/update SmartHome* |
|  | *choose Smart Device* |

|  |  |
| --- | --- |
| *Actor* | *Goal* |
| *Owner of IOT system* | *Update system* |
|  | *Control Camera functionality* |
|  | *Name router* |
|  | *choose Smart Device* |
|  | *Make Strong Encryption of Wifi* |
|  | *Disable features* |
|  | *Change Default username and password* |

### **3.2.1** Login Interface

3.2.1.1 (Introduction)This functional requirement is intended to allow the customer to login into their smart home securely

3.2.1.2 (Inputs) UserName, Passwords, and same internet connection

3.2.1.3 (Processing)Checks database for userName and password. Then checks if same the wifi connection is associated with the account within the database.

3.2.1.4 (Outputs)Access to the feedback board and Smart Devices of the devices you have registered to control

3.2.1.5 (Error Handling)wrong userName, password, or wifi connection. Redirected to login

### 3.2.2 Options Interface

3.2.2.1 (Introduction)This functional requirement is intended to allow the customer to choose between if they would like to leave feedback on the system or to select a Smart Devices to control

3.2.2.2 (Inputs) options for ‘Feedback’ or ‘Smart Devices’

3.2.2.3 (Processing) If users chooses ‘Feedback’ then the system is redirected to the Feedback page. If the user choose ‘Smart Devices’ then the system is redirected to the Smart Devices page.

3.2.2.4 (Outputs) Either the system is redirected to the Feedback page or the Smart Devices page based on the user’s prior selection

3.2.2.5 (Error Handling)System crashes and restarted and is redirected to the Login page

### 3.2.3 Feedback Board

3.2.3.1 (Introduction)This functional requirement is intended to allow the customer to leave their feedback on the software to improve the software or to report a problem/bug in the software

3.2.3.2 (Inputs)options for ‘Improvement of software’ or ‘Report a problem/bug’ and their statement

3.2.3.3 (Processing)Inputs their statement into database under either of two classifications so the developers can issue the customer’s feedback promptly.

3.2.3.4 (Outputs) A window that states ‘Thank you for your feedback!’ and after closing the window, the customer is redirected to the page to access the feedback board and Smart Devices

3.2.3.5 (Error Handling)no selection or text area is left empty. Redirected to feedback board

### 3.2.4 Smart Devices Interface

3.2.4.1 (Introduction)This functional requirement is intended to allow the customer to choose which smart device they would like to control. They will also be able to add, delete, or update a smart device.

3.2.4.2 (Inputs)options for all the Smart Devices they have registered and a Add/Delete/Update button to add/delete/update a smart device. A back button will also be included in this interface.

3.2.4.3 (Processing)System will process what interface it will be directed to based on what Smart Device or option they choose.

3.2.4.4 (Outputs) The desired interface in which the user would like to go to based on what Smart Device they would like to control or if they would like to add/delete/update a smart device

3.2.4.5 (Error Handling)System crashes or Smart Device is no longer in the database, redirected to the Smart Device Interface

### 3.2.5 Air Conditioner (AC) Interface

3.2.5.1 (Introduction)This functional requirement is intended to allow the customer to control their AC within their whole house.

3.2.5.2 (Inputs)options for all the AC would be heat/fan/off buttons, a numerical input for the temperature and a back button

3.2.5.3 (Processing)System will process the users input automatically as the user changes an option in the AC interface and change the AC’s settings

3.2.5.4 (Outputs) The desired outputs would be the automatic changes the user makes and it redirected to the Smart Devices interface once user clicks on the Back button

3.2.5.5 (Error Handling)System crashes, AC is no longer in the database (redirected to the Smart Devices Interface), or User attempts to set the temperature out of bounds (Error window is shown)

### 3.2.6 Sensors Interface

3.2.6.1 (Introduction)This functional requirement is intended to allow the system to automatically check all the sensors (fire, gas, etc.) and send the user a real-time warning if a leak or such that causes a dangerous environment

3.2.6.2 (Inputs) options for this interface would be to add/delete/edit a sensor and a back button

3.2.6.3 (Processing)System will process all the sensor level in the background and if a level is too high, the system will send an emergency warning to the user

3.2.6.4 (Outputs) The desired outputs would be to allow the system update in the background to to send the warnings even when the system isn’t logged in on a device

3.2.6.5 (Error Handling)System crashes or Sensor is bad (Sensor level would turn of which will send the user and error message)

### 3.2.7 Lights Interface

3.2.7.1 (Introduction)This functional requirement is intended to allow the customer to control their lights within their whole house.

3.2.7.2 (Inputs)options for all the lights would be on/off buttons, adding/deleting, color and brightness input, and a back button.

3.2.7.3 (Processing)System will process the users input automatically as the user changes an option in the light interface and changes the light settings.

3.2.7.4 (Outputs) The desired outputs would be the automatic changes the user makes and it redirected to the Smart Devices interface once user clicks on the Back button

3.2.7.5 (Error Handling) System crashes, light is no longer in the database (redirected to the Smart Devices Interface), or User attempts to set invalid option (Error window is shown)

**3.2.8 Security**

3.2.8.1 (Introduction) Customer would be able to function the security system through traditional interfacing. Will be able to select which security device it would like to control

3.2.8.2 (Inputs) Customer input includes updating the system, controlling camera, name the router, add timer, turn sensor on and off, choose smart device, disable features, change password.

3.2.8.3 (Processing) The security system will take the input of the customer and make the change to the IOT device selected.

3.2.8.4 (Output) The output of the this will include the system will update, will turn off/on features, control camera and update password for selected device.

3.2.8.5 (Error Handling) If no device is selected before choosing an action (error will show and state “Please select device to modify”. If password that you trying to update does not meet requirements (Error will print out password requirement)

### **3.2.**9 Smart Refrigerator

3.2.9.1 (Introduction) The functional requirements would allow the customer to set the temperature of different compartment of the refrigerator and check to see when filtration unit is due for change if need be.

3.2.9.2 (Inputs) Customer would be able to increase and decrease temperature of compartments and have display to show feedback.

3.2.9.3 (Processing) System would process temperature change in the background and make changes to the temperature of each compartment.

3.2.9.4 (Outputs) New temperature levels would be displayed for each compartment and so will the filtration unit level

3.2.9.5 (Error Handling) System would beep if display a warning when filtration level are low and would need replacement.

**3.2.10 Smart Oven-cooker**

3.2.10.1 (Introduction) The intended functional requirements is to allow the customer turn on/off, the cooker set the level how high or low the cooker burns, turn on/off the oven, also increase or decrease the temperature of the oven and set a timer.

3.2.10.2 (Input) Customer input would be on/off either the cooker or the oven, set level for the cooker, increase or decrease the oven temperature.

3.2.10.3 (Processing) System would do do most of the processing in the background and make changes to temperature oven and level of cooker.

3.2.10.4 (Outputs) Display would have cooker status on/off, cooker level, oven status on/off, oven temperature, oven light indicator and countdown clock

3.2.10.5 (Error Handling) System would beep if clock runs out of set time and turn of either cooker or oven.

**3.2.11 Smart Dishwasher**

3.2.11.1 (Introduction) The functional requirements intended would allow customer to set a daily autowash which will wash dishes at a particular time daily.

3.2.11.2 (Inputs) Inputs from customer would be o/off, regular wash, or auto wash and also a time set for daily auto wash.

3.2.11.3 (Processing) System would carry out all the processing in the background and update changes if need be.

3.2.11.4 (Outputs) Dishwasher status would be displayed if on/off, on a regular wash, set for auto wash or daily auto wash.

3.2.11.5 (Error Handling) System beeps when wash is done and turns off, in case of overflow dishwashers turns off.

**3.2.12 Smart Microwave**

3.2.12.1 (Introduction) The functional requirements intended would allow customer to warm ,

cook and defrost food.

3.2.12.2 (Inputs) Input would be made by the customer on/off , start/stop, options, set time to warm , cook or defrost the food

3.2.12.3 (Processing) System will process the users input automatically as the user changes the time they want.

3.2.12.4 (Output) Display should show food is ready when the time set by customer is done counting down.

3.2.12.5 (Error Handling) display would show close door if microwave door is open, display would also ask to set time if not done before pressing start.

### 3.2.13 Schedule Interface

3.2.13.1 (Introduction) This functional requirement is intended to allow the customer to set schedules for any device within the home.

3.2.13.2 (Inputs)options for schedule would be time, day, repeat option, on/off, and a back button.

3.2.13.3 (Processing) System will process the users input automatically saving it into database to trigger schedule when needed.

3.2.13.4 (Outputs) The desired outputs would be the automatic database update the user makes and it redirected to the Smart Devices interface once user clicks on the Back button

3.2.13.5 (Error Handling) System crashes, device is no longer in the database (redirected to the Smart Devices Interface), or User attempts to set invalid schedule (Error window is shown)

### 3.2.14 Smart Tv Interface

3.2.14.1 (Introduction) This functional requirement is intended to allow the customer to control smart tvs.

3.2.14.2 (Inputs)options for the tv would be on/off, volume, channel, ok button, direction keys and a back button.

3.2.14.3 (Processing) System will process the users input automatically as user makes changes

3.2.14.4 (Outputs) The desired outputs would be the automatic changes the user makes and redirected to the Smart Devices interface once user clicks on the Back button

3.2.14.5 (Error Handling) System crashes, device is no longer in the database (redirected to the Smart Devices Interface), or User attempts to do an invalid action (Error window is shown)

### 3.2.15 Smart Speaker Interface

3.2.15.1 (Introduction) This functional requirement is intended to allow the customer to control smart speaker options.

3.2.15.2 (Inputs)options for the speaker would be on/off, volume,mute, bass, treble, input, play/pause, next/previous and a back button.

3.2.15.3 (Processing) System will process the users input automatically as user makes changes

3.2.15.4 (Outputs) The desired outputs would be the automatic changes the user makes and redirected to the Smart Devices interface once user clicks on the Back button

3.2.15.5 (Error Handling) System crashes, device is no longer in the database (redirected to the Smart Devices Interface), volume limit, or User attempts to choose non existing device.

### 3.2.16 Smart Plug Interface

3.2.16.1 (Introduction) This functional requirement is intended to allow the customer to control any non smart device that is plugged into a smart plug.

3.2.16.2 (Inputs)options for the smart plug would be on/off, and back button.

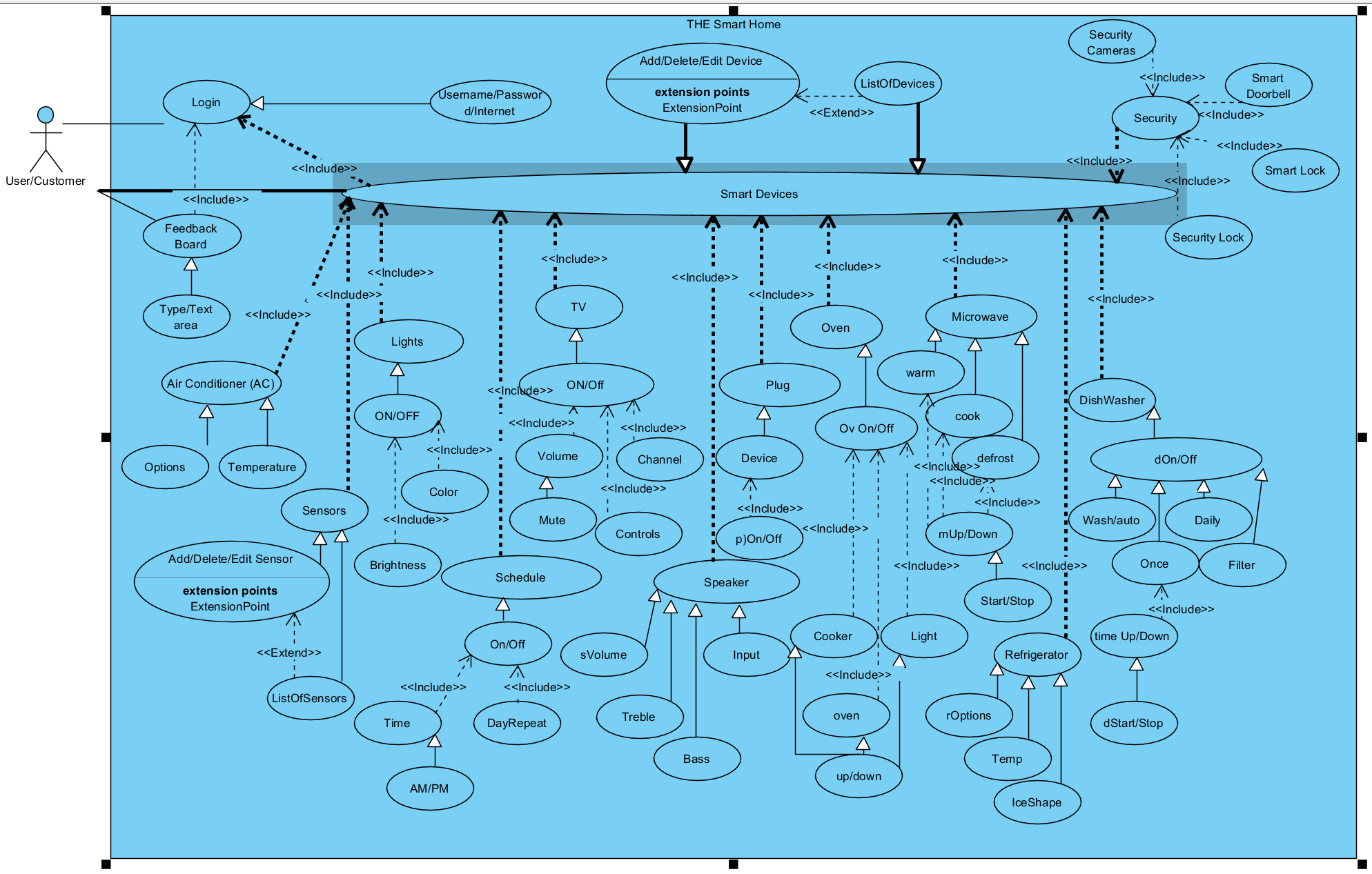
3.2.16.3 (Processing) System will process the users input automatically as user makes changes

3.2.16.4 (Outputs) The desired outputs would be the automatic changes the user makes and redirected to the Smart Devices interface once user clicks on the Back button

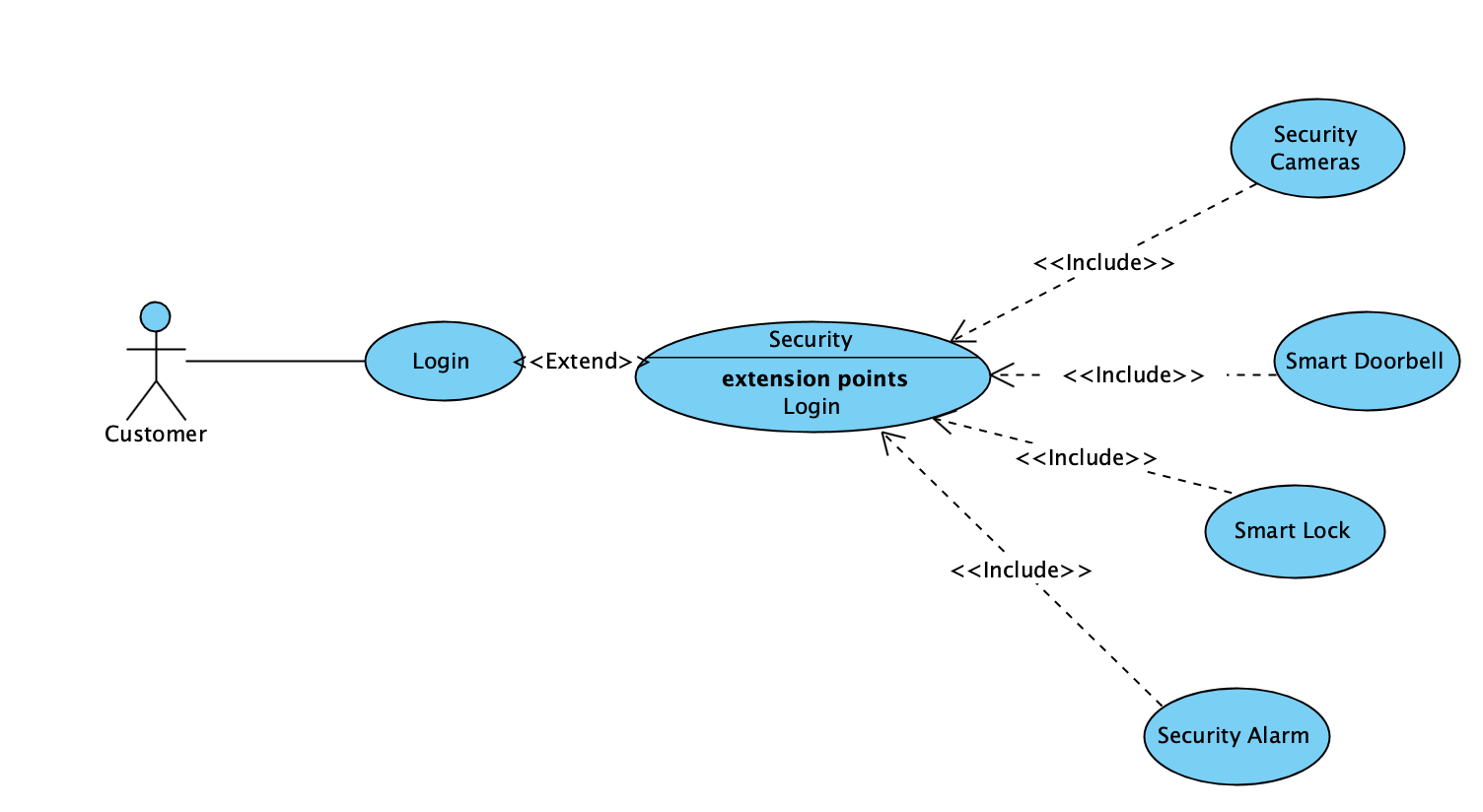
3.2.16.5 (Error Handling) System crashes (redirected to the Smart Devices Interface), device is no longer plug in (error message will be displayed).

**3.3 Use Cases**

**Use case**



**Use Case for 3.3.8 and 3.3.17-3.3.19**

****

**3.3.1 Use Case** - Login Interface

Allows User to login into their smart home account via the same wifi network the smart home is connected too

|  |  |
| --- | --- |
| Use Case Name: User Login | ID: UC-001 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  User will be able to use this page to login in into their smart home account. | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:   1. User starts the system 2. login page is shown 3. user enters in username and password 4. login in page disappears and is redirected to the Options page | |
| Alternate/Exceptional Flows:  user types wrong info and login page is displayed again | |
| Pre-Conditions:  User must exist in the database | |
| Post-Conditions:  User is logged in and is redirected to the Options page | |

### 

### 3.3.2 Use Case - Options Interface

Allows User to choose their next page whether if they would like to leave some type of feedback or to view the Smart Devices they can control

|  |  |
| --- | --- |
| Use Case Name: Options | ID: UC-002 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose to go to the Feedback page or the Smart Devices page | |
| Trigger: User successfully logs-in | |
| Normal Flow of Events:   1. User chooses ‘Feedback’ option    1. system is redirected to the Feedback page   or   1. User chooses ‘Smart Devices’ option    1. System is redirected to the Smart Devices page | |
| Alternate/Exceptional Flows:  System crashed and restarts. When the system reboots, it’ll show the Login interface | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice | |

### **3.3.**3 **Use Case** - Feedback Board

User will be able to provide feedback on the system with this page. The feedback page will have three options and a textbox. This is to provide Improvements or to report a bug.

|  |  |
| --- | --- |
| Use Case Name: Feedback Board | ID: UC-003 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  User will be able to provide feedback on the system with this page | |
| Trigger: User selects ‘Feedback’ option on previous page | |
| Normal Flow of Events:   1. user chooses between the two options: ‘Improvement of software’ or ‘Report a problem/bug’ 2. user enters their thoughts into the textboxbox under the two options 3. user clicks ‘Done’ 4. Feedback page is redirected to the Option page | |
| Alternate/Exceptional Flows:   1. user doesn't select one of the two options or doesn’t enter text in the textbox before they click ‘Done’ and the Feedback page is shown again 2. User clicks the back button to be redirected to the Options page | |
| Pre-Conditions: User chooses the Feedback option on the previous Option page | |
| Post-Conditions: User is redirected to the Option Page | |

### 3.3.4 Use Case - Smart Devices

User will be able to choose which Smart Device they will like to control or they have an option to add/delete/update a Smart Device

|  |  |
| --- | --- |
| Use Case Name: Smart Devices | ID: UC-004 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose which Smart Device they will like to control or they have an option to add/delete/update a Smart Device. With what they choose, the system will automatically redirect the User to their desire Smart Device interface | |
| Trigger: User selects ‘Smart Devices’ option on previous page | |
| Normal Flow of Events:   1. User chooses which Smart Device they would like to control    1. System is redirected to the desired interface for the Smart Device   or   1. User chooses ‘Add/Delete/Update Smart Device’    1. A window pops up so they user can do one of the following functions: Add Smart Device, Delete Smart Device, or Update Smart Device   or   1. User selects ‘Back’    1. the system redirects User to the Options page | |
| Alternate/Exceptional Flows:   1. User selects the ‘Back’ option and the user is redirected to the Options page 2. The Smart Device they choose doesn’t exist in the database | |
| Pre-Conditions: User chooses the ‘Smart Devices’ option on the previous Option page | |
| Post-Conditions: User is redirected to the desired interface for the Smart Device they would like to control or redirected to a window so they can add/delete/update a Smart Device. | |

### 3.3.5 Use Case - Air Conditioner (AC)

The customer will have the ability to control their AC unit within their whole house

|  |  |
| --- | --- |
| Use Case Name: AC | ID: UC-005 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  User will be able to control their AC unit with the options of heat/fan/off and a numerical input to change the temperature, itself. The system will automatically change the AC when the user changes a setting within the AC interface | |
| Trigger: User selects ‘Air Conditioner (AC)” option on previous page | |
| Normal Flow of Events:   1. user changes desired option within the interface 2. system automatically changes the AC unit’s settings | |
| Alternate/Exceptional Flows:   1. User enter a numeric value for the temperature which is out of bounds    1. system send error message | |
| Pre-Conditions: User chooses the ‘Air Conditioner (AC)’ option on the previous Smart Devices page | |
| Post-Conditions: User is redirected to the to the Smart Devices page once user clicks the Back option on the AC interface | |

### 3.3.6 Use Case - Sensors

Allows the system to automatically check all the sensors (fire, gas, etc.) and send the user a real-time warning if a leak or such that causes a dangerous environment in the background. The user can also add/delete/edit a sensor

|  |  |
| --- | --- |
| Use Case Name: Sensors | ID: UC-006 |
| Version: 0.5 | Author: Anuj Patel |
| Primary Actor: customer | |
| Brief Description:  The system will run in the background and if a sensor’s level is too high/low the system will automatically send the user a warning on all registered devices. The user’s options in this user face would be to add/delete/edit a sensor | |
| Trigger: User selects ‘Sensors’ option on previous page | |
| Normal Flow of Events:   1. User clicks on add/delete/edit option 2. user does desired thing with the chosen option above 3. user clicks back | |
| Alternate/Exceptional Flows:   1. Sensor is damaged    1. system sends a notification to replace | |
| Pre-Conditions: User chooses the ‘Sensors’ option on the previous Smart Devices page | |
| Post-Conditions: User is redirected to the Smart Devices page once the Back button is clicked on | |

### 3.3.7 Use Case - Lights

Allows the user to interact with lights option of turning on/off, changing color, and brightness

|  |  |
| --- | --- |
| Use Case Name: Lights | ID: UC-007 |
| Version: 0.5 | Author: Alex Sanchez |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose to go to the Feedback page or the Smart Devices page | |
| Trigger: User selects lights system | |
| Normal Flow of Events:   1. user changes desired option within the interface 2. customer changes color or brightness or turns device on or off 3. system automatically changes the lights’ settings | |
| Alternate/Exceptional Flows:  System crashed and restarts. When the system reboots, it’ll show the Login interface  Device is removed after choosing it system will throw an error. power goes out system will alert customer | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice. Device is updated | |

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 

### 3.3.8 Use Case - Camera

Use case descriptions here

|  |  |
| --- | --- |
| Use Case Name: Security | ID: UC-008 |
| Version: 0.0.0 | Author: Scotty Rodriguez |
| Primary Actor: customer | |
| Brief Description: The security interface will let you select which security devices you will like to add/configure. Such as surveillance camera, motion detectors, and speaker built into camera.  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:  1.Customer clicks on security device they want to view/configure  2.Option to select device, add new device, delete device  3.Can rotate camera, view live camera, change password or back to menu  1.2 Gets an notification of motion detected on camera on mobile device  1.3 Open notification can view live feed of camera that picked up motion and also get snapshot of the motion picked up  4 Exit | |
| Alternate/Exceptional Flows:  One of the security cameras go offline or stops working   1. The system will send me a notification that system is offline and it will automatically will try to reset that camera | |
| Pre-Conditions: All cameras must be connected to the same network | |
| Post-Conditions: Customer will have the ability to view the live feed and or captured photos of people caught on the cameras | |

### 3.3.9 Use Case - Smart Refrigerator

The customer would have the ability to to set temperature for the different compartments of the refrigerator from his device.

|  |  |
| --- | --- |
| Use Case Name: Smart Refrigerator | ID: UC-009 |
| Version: 0.0.0 | Author: Onyeka Mordi |
| Primary Actor: customer | |
| Brief Description: the customer can set the temperature of different compartment of the refrigerator and check to see when filtration unit is due for change if need be. This will all be available on the display and can be accessed remotely  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:   1. customer clicks on fridge options for the different compartment show 2. customer select a compartment 3. temperature regulator pop up for client to set temperature 4. after temperature is set client display goes back to the option screen 5. customer click exit to head to home screen. | |
| Alternate/Exceptional Flows:  filtration system is due for change fridge beeps and has a red light indicator   1. customer click on the indicator 2. message shows changer filter 3. customer exit 4. if filter is not changed with in one hour loop runs again 5. if filter is changed filtration indicator turns green | |
| Pre-Conditions: fridge must be connected to the internet and must be add to the smart home devices. | |
| Post-Conditions: customer can view and make adjustments to the refrigerator from anywhere in the house | |

**3.3.10 Smart Oven-cooker**

|  |  |
| --- | --- |
| Use Case Name: | ID: UC-010 |
| Version: 0.0.0 | Author: Onyeka Mordi |
| Primary Actor: customer | |
| Brief Description: Tallow the customer turn on/off, the cooker set the level how high or low the cooker burns, turn on/off the oven, also increase or decrease the temperature of the oven and set a timer.  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:   1. customer clicks on/off on the cooker / oven 2. asked to pick level or set temperature 3. clicks start to begin | |
| Alternate/Exceptional Flows:   1. customer pick wrong option and wishes to go back 2. double clicks on stop that take system back to beginning | |
| Pre-Conditions: Display would have cooker status on/off, cooker level, oven status on/off, oven temperature, oven light indicator and countdown clock | |
| Post-Conditions: System beeps when wash is done and turns off, in case of overflow dishwashers turns off. | |

**3.3.11 Smart Dishwasher**

|  |  |
| --- | --- |
| Use Case Name: | ID: UC-011 |
| Version: 0.0.0 | Author: Onyeka Mordi |
| Primary Actor: customer | |
| Brief Description: The functional requirements intended would allow customer to set a daily autowash which will wash dishes at a particular time daily.  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:   1. customer clicks on/off 2. has options to wash now, autowash 3. set if use cold/ hot water option 4. when finished dishwasher turns off | |
| Alternate/Exceptional Flows:   1. customer selected wash now but wants to set it to autowash later 2. clicks on stop washer stop 3. clicks onn stop again and display goes back to home option | |
| Pre-Conditions: Dishwasher status would be displayed if on/off, on a regular wash, set for auto wash or daily auto wash. | |
| Post-Conditions: System beeps when wash is done and turns off, in case of overflow dishwashers turns off. | |

**3.3.12 Smart Microwave**

|  |  |
| --- | --- |
| Use Case Name: | ID: UC-012 |
| Version: 0.0.0 | Author: Onyeka Mordi |
| Primary Actor: customer | |
| Brief Description: The functional requirements intended would allow customer to warm ,  cook and defrost food.  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:   1. customer pick options on display 2. customer set the time 3. customer clicks starts 4. when timer is done system stop. | |
| Alternate/Exceptional Flows:   1. customer can stop current process 2. clicks on stop to pause system process 3. double clicks on | |
| Pre-Conditions: Display should show food is ready when the time set by customer is done counting down. | |
| Post-Conditions: display would show close door if microwave door is open, display would also ask to set time if not done before pressing start. | |

### 3.3.13 Use Case - Schedule

Allows the user to set a schedule whether to turn on or off on a certain day or days.

|  |  |
| --- | --- |
| Use Case Name: Schedule | ID: UC-0013 |
| Version: 0.5 | Author: Alex Sanchez |
| Primary Actor: customer | |
| Brief Description:  User will be able to set a schedule for desired device | |
| Trigger: User selects scheduling system | |
| Normal Flow of Events:   1. user sets time and frequency,and whether to turn on or off within interface 2. user confirms action 3. system updates settings and schedule to its database and obeys the command | |
| Alternate/Exceptional Flows:System crashed and restarts. When the system reboots, it’ll show the Login interface Device is removed after choosing it system will throw an error. | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice. Device is updated | |

### 3.3.14 Use Case - Tv

Allows the user to interact with tv option of turning on/off, volume control, changing channel, and maneuvering through tv

|  |  |
| --- | --- |
| Use Case Name: Tv | ID: UC-0014 |
| Version: 0.5 | Author: Alex Sanchez |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose to interact with tv turning it on or off changing channel or volume and buttons to go through tv settings | |
| Trigger: User selects tv system | |
| Normal Flow of Events:   1. user changes desired option within the interface   2. customer changes color or brightness or turns device on or off  3. system automatically changes the tv’s settings | |
| Alternate/Exceptional Flows:  System crashed and restarts. When the system reboots, it’ll show the Login interface  Device is removed after choosing it system will throw an error. Volume too high system will give a volume warning. | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice. Device is updated | |

### 3.3.15 Use Case - Speaker

Allows the user to interact with speaker option of turning on/off, controlling volume, bass, treble, and input device.

|  |  |
| --- | --- |
| Use Case Name: Speaker | ID: UC-0015 |
| Version: 0.5 | Author: Alex Sanchez |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose to go to interact with the speaker | |
| Trigger: User selects speaker system | |
| Normal Flow of Events:   1. user changes desired option within the interface   2. customer changes desired options  3. system automatically changes the speaker’s settings | |
| Alternate/Exceptional Flows:  System crashed and restarts. When the system reboots, it’ll show the Login interface  Device is removed after choosing it system will throw an error. Volume too high it will give a warning | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice. Device is updated | |

### 3.3.16 Use Case - Plug

Allows the user to turn on or off device that is connected to the smart plug.

|  |  |
| --- | --- |
| Use Case Name: Plug | ID: UC-0016 |
| Version: 0.5 | Author: Alex Sanchez |
| Primary Actor: customer | |
| Brief Description:  User will be able to choose to turn on or off any device thats connected to the plug | |
| Trigger: User selects plug system | |
| Normal Flow of Events:  1. user changes desired option within the interface  2. customer turns device on or off  3. system automatically changes the plug’s settings | |
| Alternate/Exceptional Flows:  System crashed and restarts. When the system reboots, it’ll show the Login interface  Device is removed after choosing it system will throw an error. | |
| Pre-Conditions:  User successfully logs-in | |
| Post-Conditions:  User is redirected to the desired page in the system based on their choice. Device is updated | |

### 3.3.17 Use Case - Smart Doorbell

Use case descriptions here

|  |  |
| --- | --- |
| Use Case Name: Smart Doorbell | ID: UC-017 |
| Version: 0.0.0 | Author: Scotty Rodriguez |
| Primary Actor: customer | |
| Brief Description: The doorbell will have a built in motion sensor, speaker and camera that will send an alert to your mobile device with a picture of captured object when motion is detected. The configuration will let you select if you would like to configure, view live feed, or use speaker  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:  1.Camera picks up motion  2.Sends alert to mobile device  3.Customer can see live feed, see photo, or talk on speaker  4 Contact police or Exit  1.1 User interface for configuration  1.2 Can update mobile device listed, the sensitivity of motion, and if there should be a delay in captured photo  1.3 Exit | |
| Alternate/Exceptional Flows:  Smart doorbell goes offline or stops working   1. The system will send me a notification that system is offline and it will automatically will try to reset the smart doorbell | |
| Pre-Conditions: Smart Doorbell needs to be connected to the network | |
| Post-Conditions: Customer will have the ability to view the live feed and or captured photos of people caught on the cameras | |

### 3.3.18 Use Case - Smart Lock

Use case descriptions here

|  |  |
| --- | --- |
| Use Case Name: Smart Lock | ID: UC-018 |
| Version: 0.0.0 | Author: Scotty Rodriguez |
| Primary Actor: customer | |
| Brief Description: With the Smart Lock you will be able to lock and unlock your home with your interconnected devices from anywhere you have internet connection  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:  1.Go to device interface  2.Select door  3.Unlock or lock door  4 Exit  1.1 User interface for configuration  1.2 Can add or remove smart locks add enter system settings  1.3 Exit | |
| Alternate/Exceptional Flows:  Smart lock goes offline or stops working   1. The system will send me a notification that which smart lock is offline and it will automatically will try to reset the smart doorbell | |
| Pre-Conditions: Smart Lock needs to be connected to the same network | |
| Post-Conditions: Customer can lock and unlock doors in their smart home | |

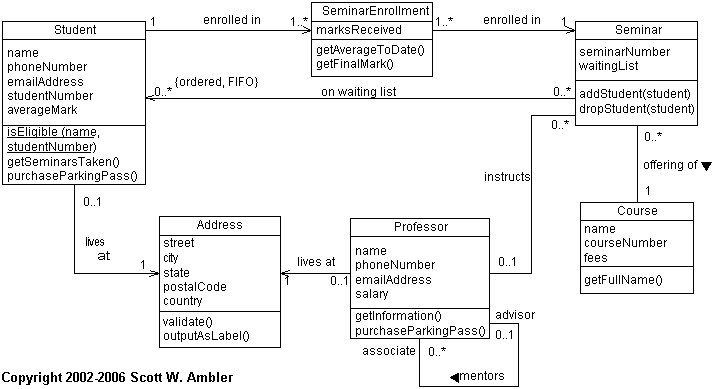
### 3.3.19 Use Case - Alarm System

Use case descriptions here

|  |  |
| --- | --- |
| Use Case Name: Security Alarm System | ID: UC-019 |
| Version: 0.0.0 | Author: Scotty Rodriguez |
| Primary Actor: customer | |
| Brief Description: The security interface will let you enter configuration to set a password, network and button configurations. This will have motion detectors and will have the ability to turn on alarms and motion detection from a connected device from a offsite location  User …… | |
| Trigger: User starts to use the the system | |
| Normal Flow of Events:  1.Customer enters pin for Security Alarm  2.Set security alarm to stay, away, turn motion off/on, close and open windows using any interconnect device  3.Exit  1.1 Enter password to configuration  1.2 Change pin, add/remove mobile device, or configure the network  1.3 Exit | |
| Alternate/Exceptional Flows:  One of the security alarm disconnects from mobile device or network   1. The system will send me a notification that system is offline and it will automatically will try to reset that Alarm | |
| Pre-Conditions: The mobile device must be allowed to control alarm system | |
| Post-Conditions: Customer will have the ability to set alarm, set motion, and open/close windows while they are or are not home | |

## **3.4 Classes / Objects**

Class diagrams and Object diagrams here.



### **3.4.1 <Class / Object #1>**

3.4.1.1 Attributes

3.4.1.2 Functions

<Reference to functional requirements and/or use cases>

### **3.4.2 <Class / Object #2>**

…

## **3.5 Non-Functional Requirements**

### **3.5.1 Performance**

*Since the system is interactive, there should be minimal delays. As far as measurable terms:*

*1) connecting to server (based on distance between servers) 30 seconds max*

*2) opening the database 0-5 seconds*

*3) pop up windows (save settings or the current session) 5 seconds max*

*4) opening, computing, editing, and posting 95% of transactions 5 seconds max*

### **3.5.2 Reliability**

*Have a feedback board to receive feedback and if any bad feedback, solve the problem to keep the system operable and any sensitive information secure.*

### **3.5.3 Availability**

*The availability of the software truly is based on internet connectivity. For example, if the internet signal is disrupted while information is being transferred, the information can be verified by sending it again.*

### **3.5.4 Security**

*Since security is a major issue, a login page will be required to prevent hacking and also only registered smart devices’ serial numbers (associated with the account) will be allowed in order to control device within the smart home. Another prevention from the software to be hacked will be, the device that is used to be the controller must be connection to the same wifi connection.*

### **3.5.5 Maintainability**

*With the active feedback board, the software can always fix a known bug in which a customer reports. This board can also help the software be improved by suggestions. Other then these two things, the software will be maintained daily after the first initial launch of the software.*

### **3.5.6 Portability**

The portability is based on the smart device connectivity and the wifi connection range.

## **3.6 Inverse Requirements**

*State any \*useful\* inverse requirements.*

## **3.7 Design Constraints**

*Specify design constraints imposed by other standards, company policies, hardware limitation, etc. that will impact this software project.*

## **3.9 Other Requirements**

*Catchall section for any additional requirements.*

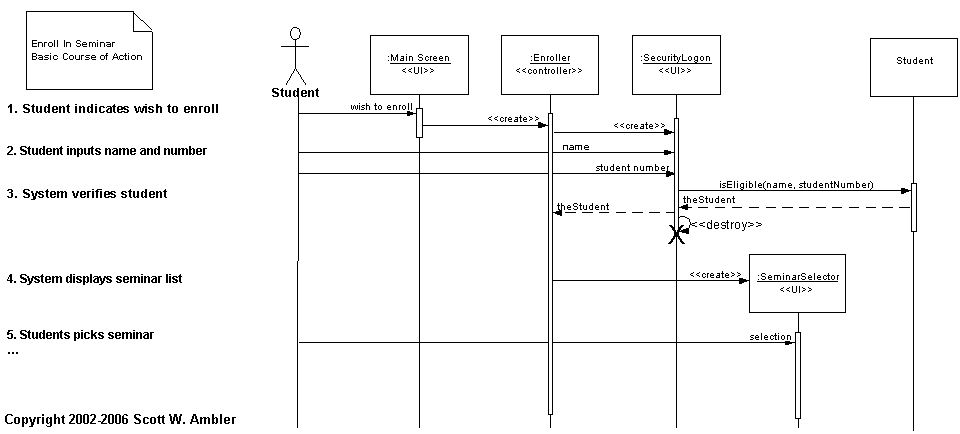
# **4. Analysis Models**

*List all analysis models used in developing specific requirements previously given in this SRS. Each model should include an introduction and a narrative description. Furthermore, each model should be traceable the SRS’s requirements.*

## **4.1 Sequence Diagrams**

Draw a sequence diagram for each use case discovered so far.

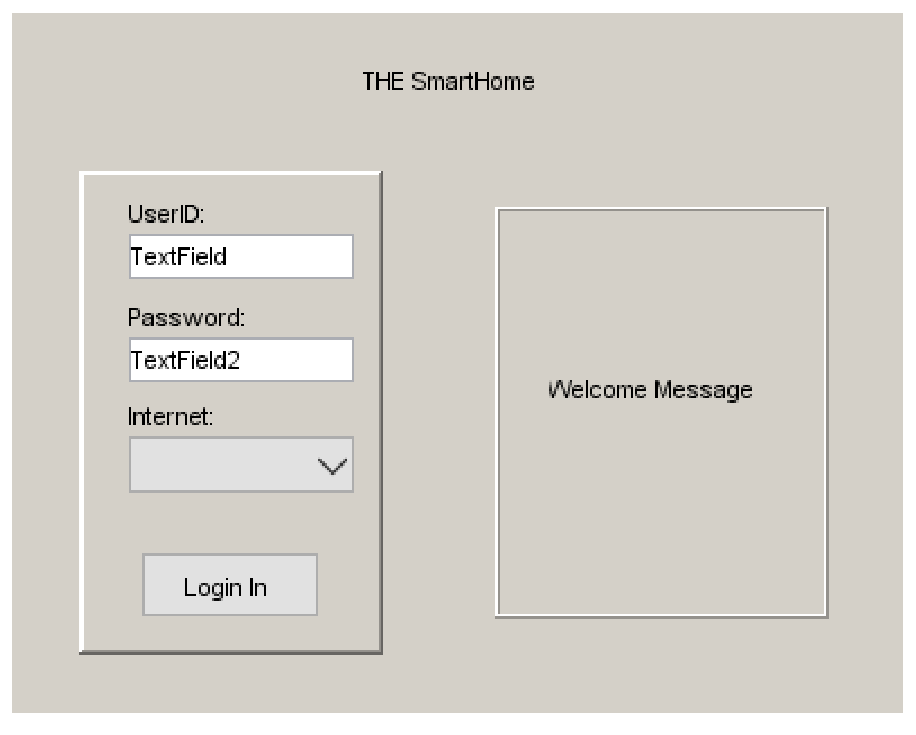
Example:



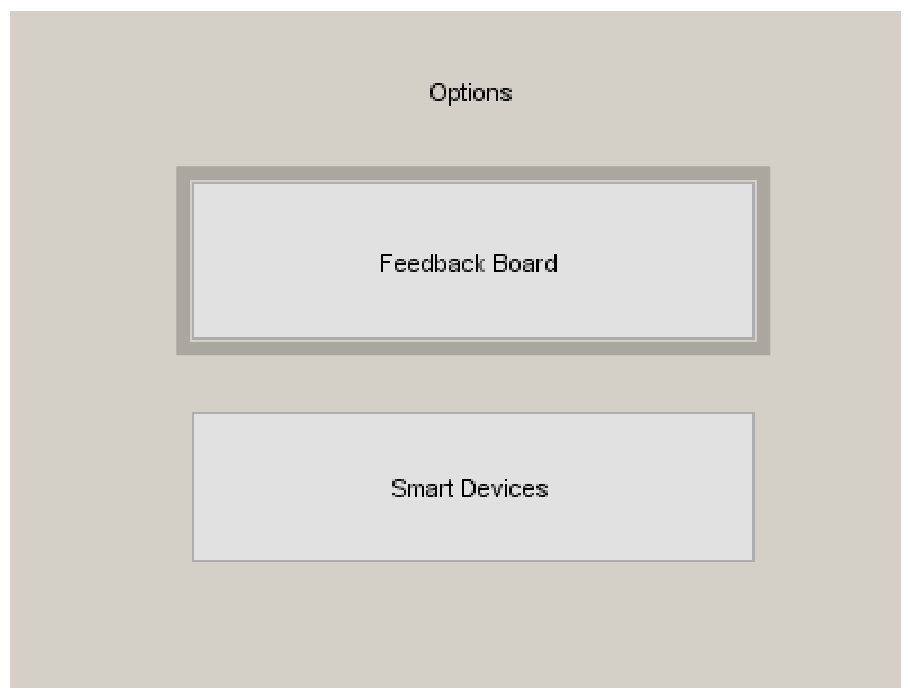
# **5. UI Design**

#### *For this section, demo the pseudo user interface and navigational paths of each use case*.

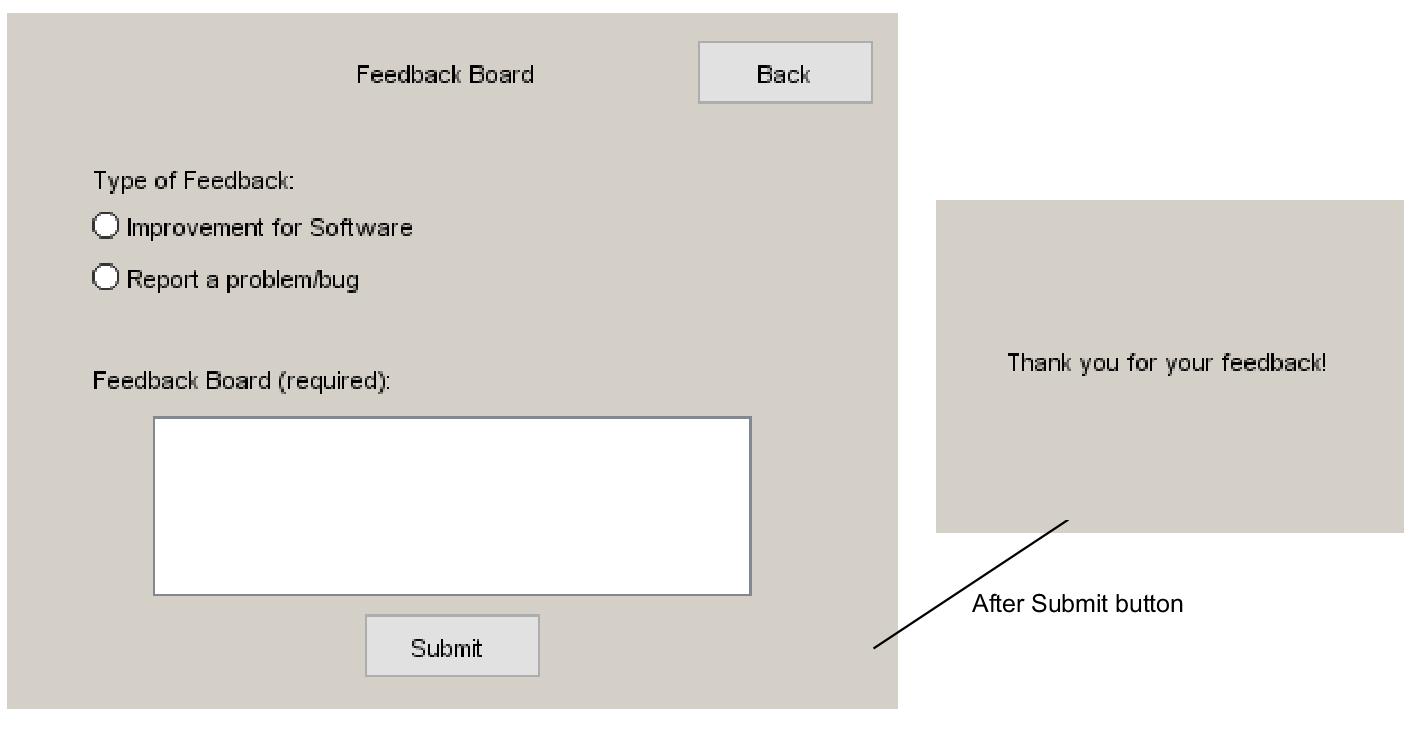
#### 5.1 Use case -- User Login Mock Screen



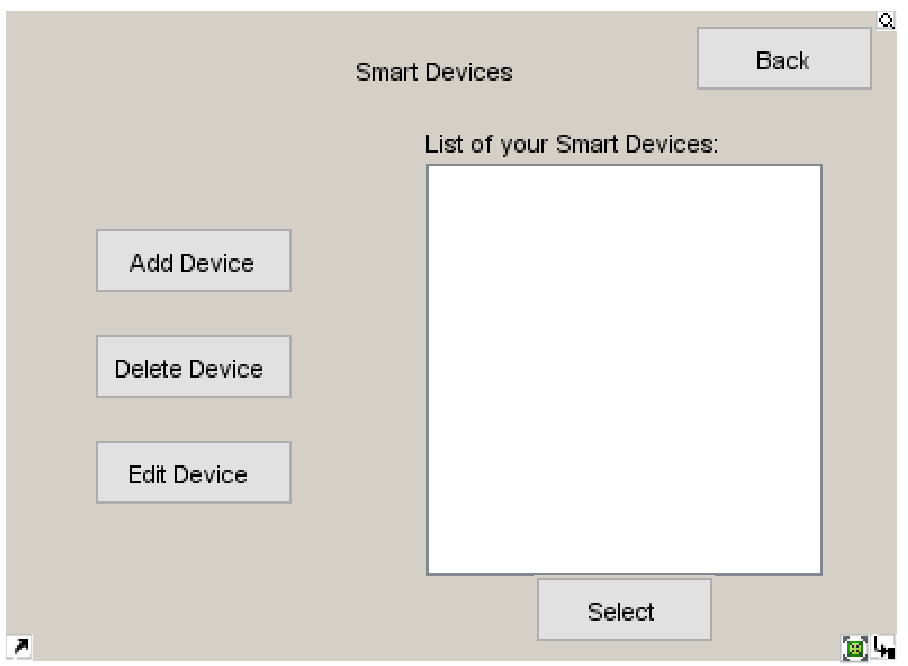
# 5.2 Use case -- Options Mock Screen



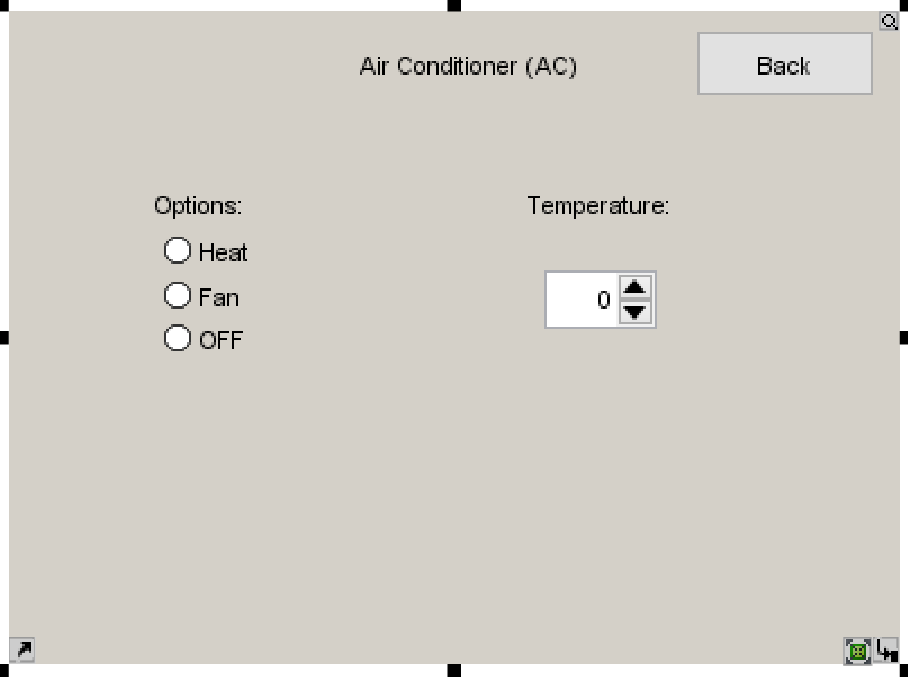
**5.3 Use case -- FeedBack Board Mock Screen**

****

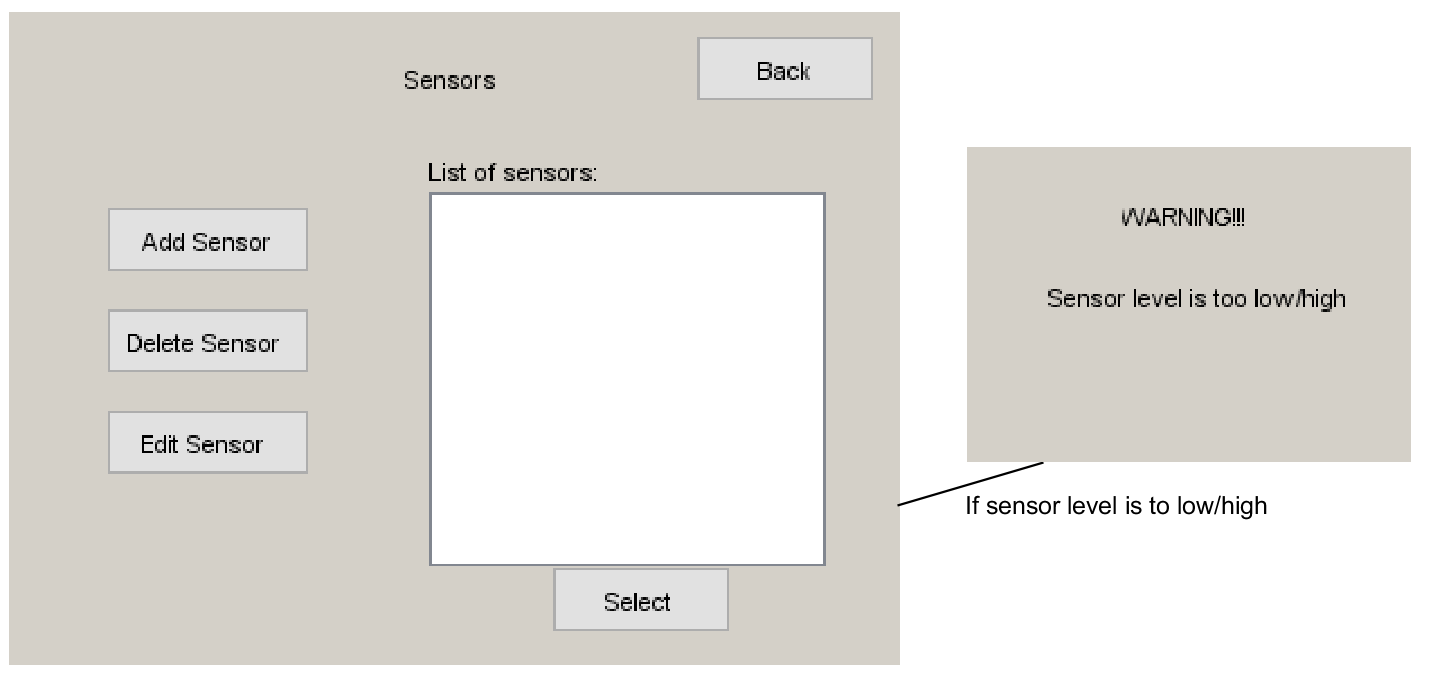
**5.4 Use case -- Smart Devices Mock Screen**

****

**5.5 Use case -- Air Conditioner (AC) Mock Screen**

****

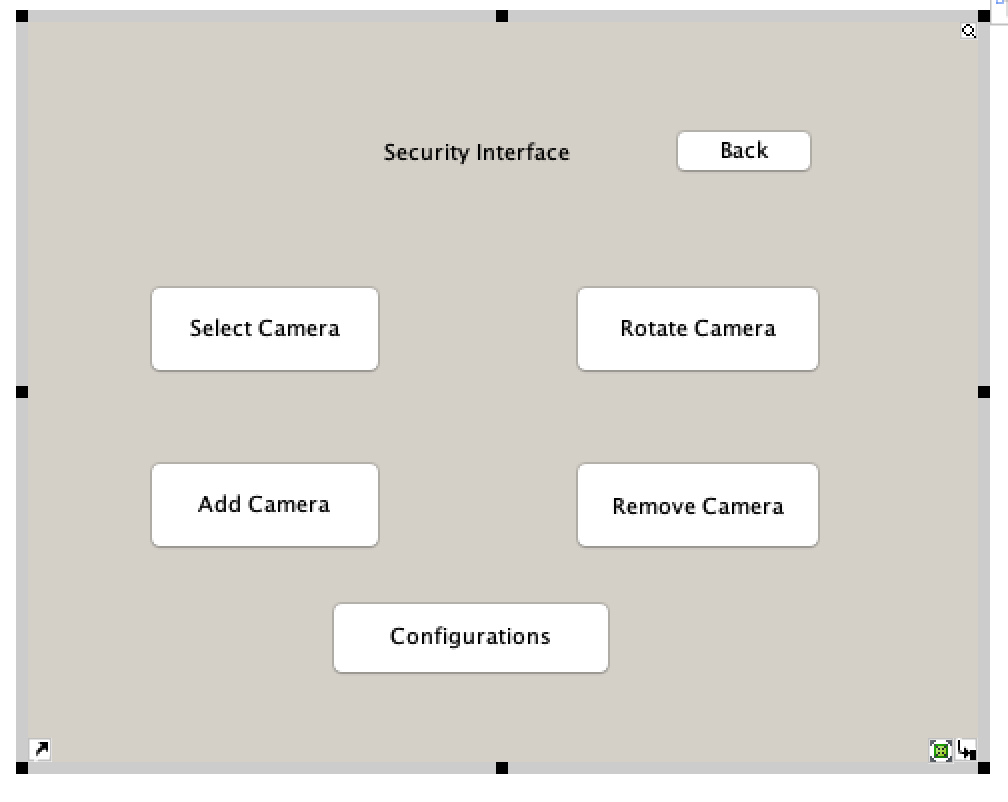
**5.6 Use case -- Sensors Mock Screen**

****

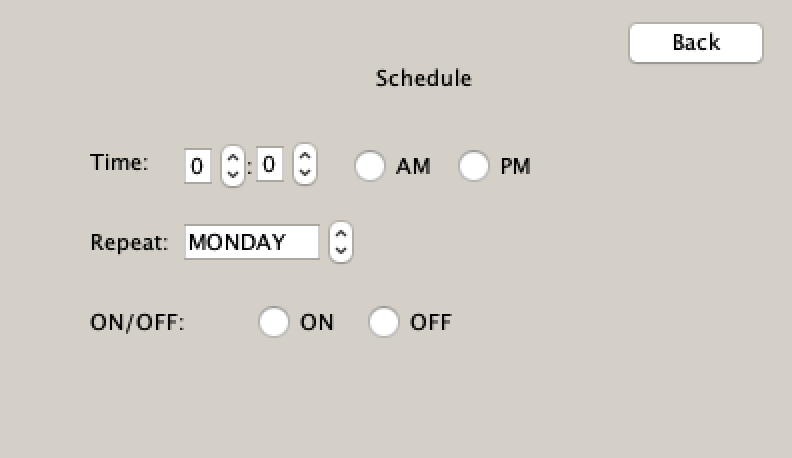
**5.7 Use case -- Lights Mock Screen**

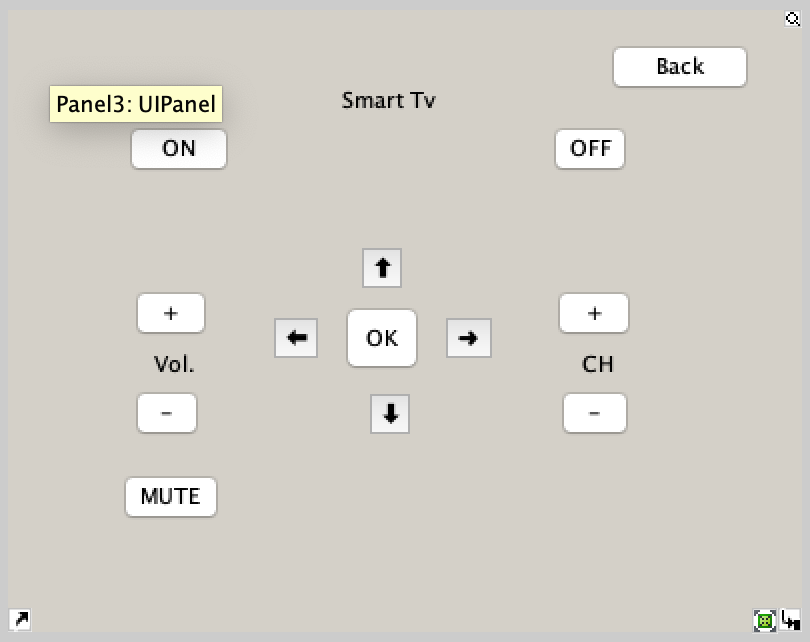
****

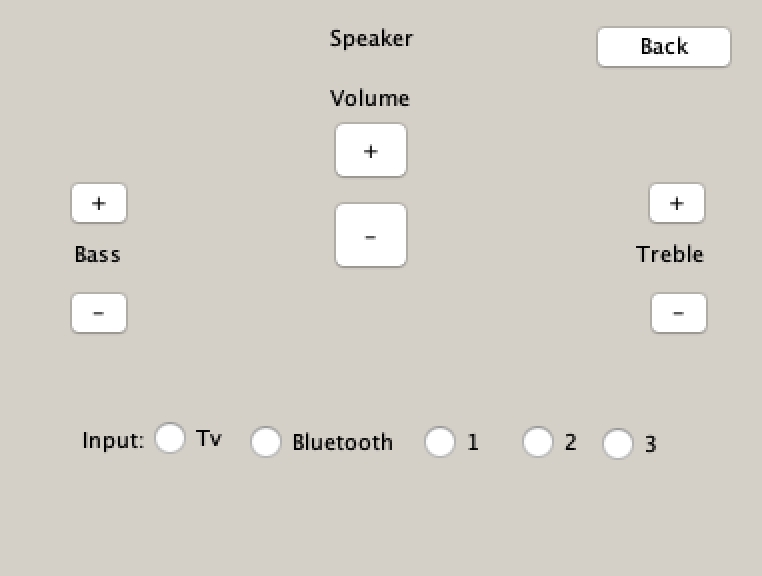
**5.8 Use case -- Camera Mock Screen**

****

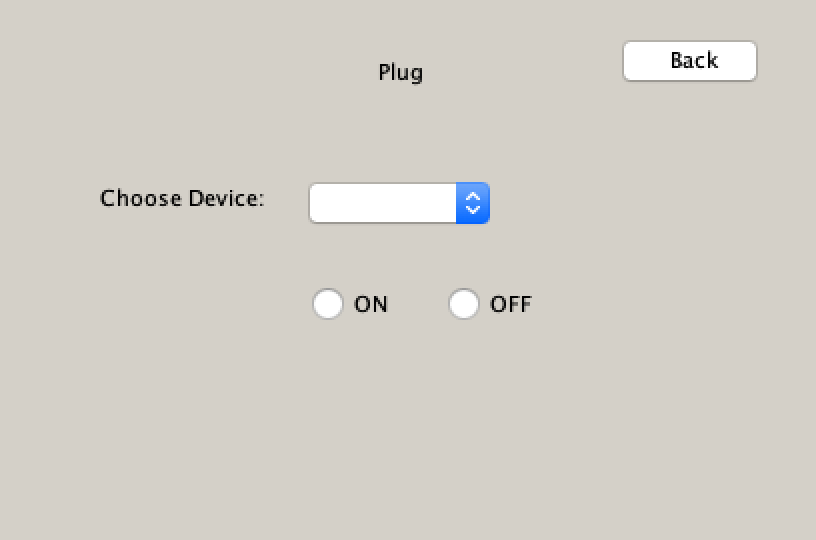
**5.13 Use case -- Schedule Mock Screen**

****

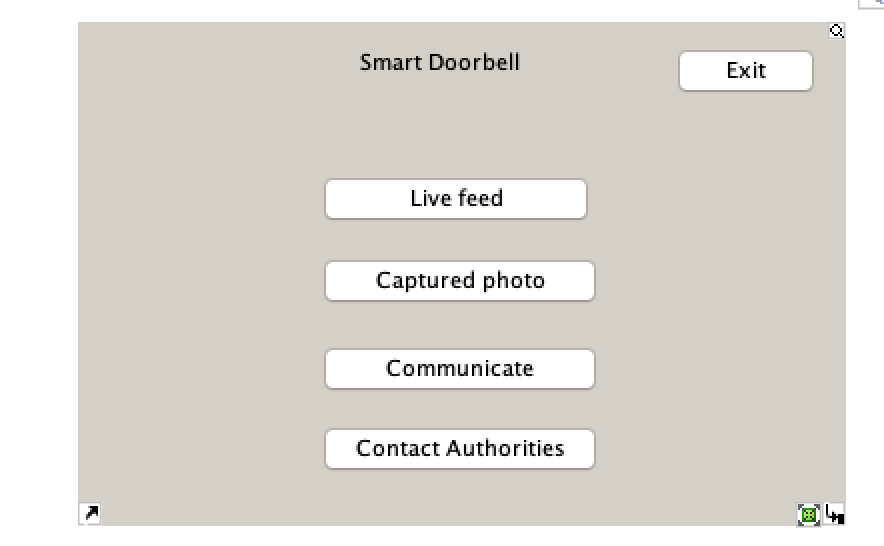
**5.14 Use case -- Smart TV Mock Screen**

**5.15 Use case -- Smart Speaker Mock Screen**

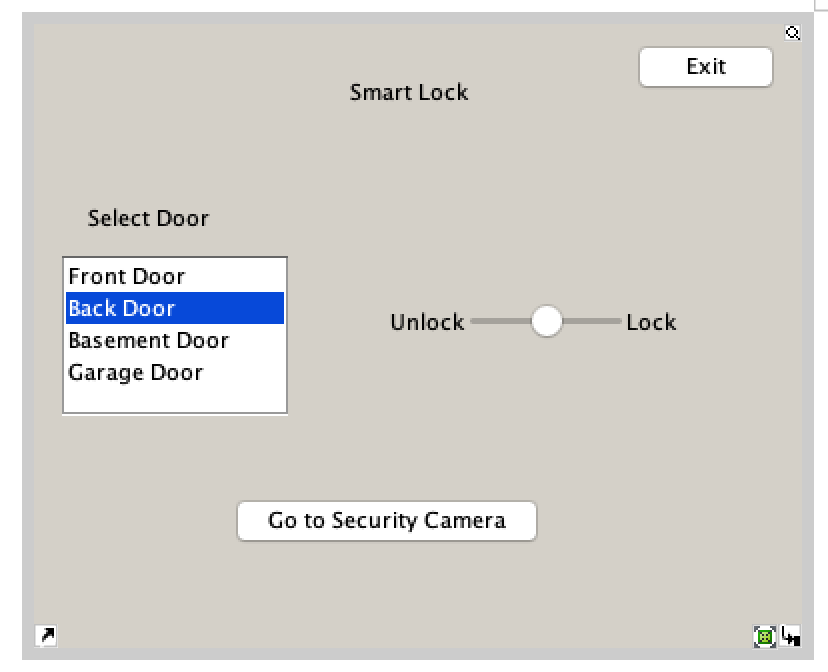
**5.16 Use case -- SmartPlug Mock Screen**

****

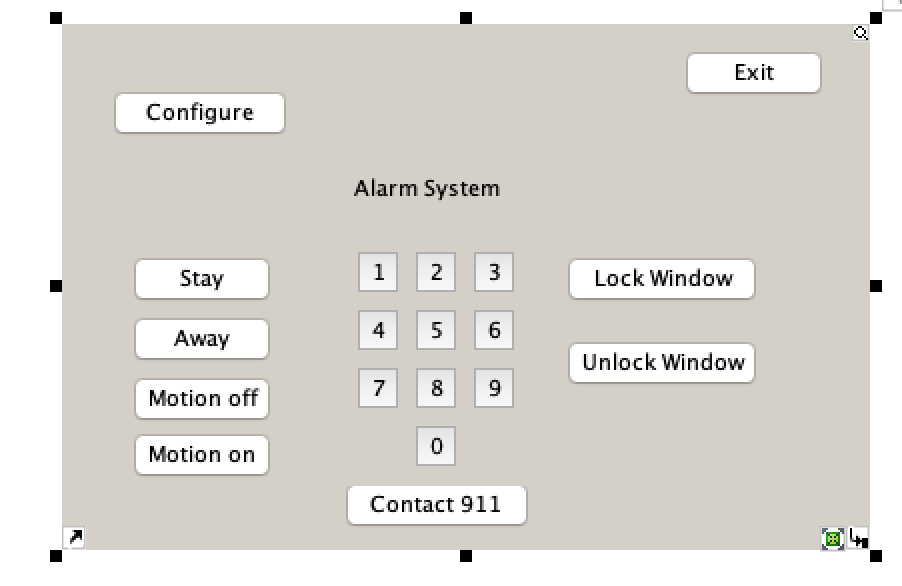
**5.17 Use case -- Smart Doorbell**

****

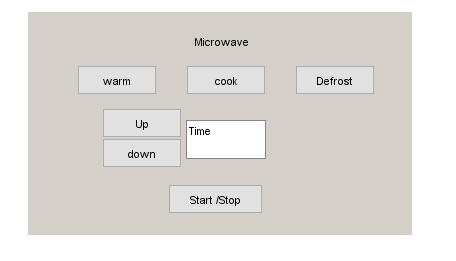
**5.18 Use case -- Smart Lock**

****

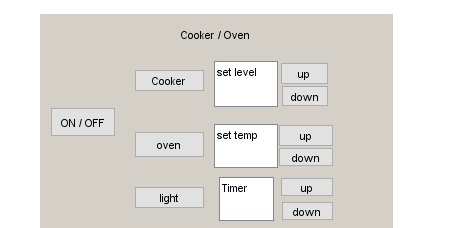
**5.19 Use case -- Alarm System**

****

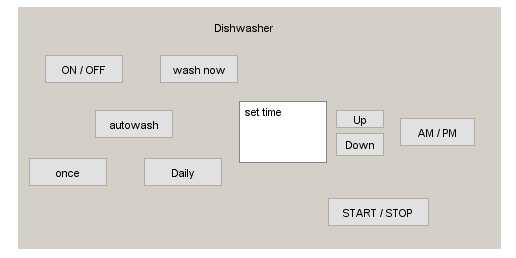
**5.20 Use case -- Smart microwave**

****

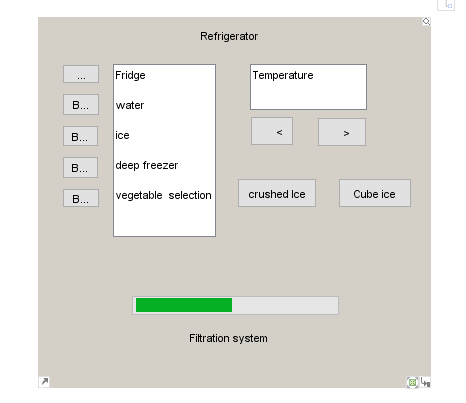
**5.20 Use case -- Smart Oven-cooker**

****

**5.21 Use case -- Smart Dishwasher**

****

**5.21 Use case -- Smart Refrigerator**

****

# **6. Change Management Process**

*Identify and describe the process that will be used to update the SRS, as needed, when project scope or requirements change. Who can submit changes and by what means, and how will these changes be approved.*

# **7. Appendices**

*Appendices may be used to provide additional (and hopefully helpful) information. If present, the SRS should explicitly state whether the information contained within an appendix is to be considered as a part of the SRS’s overall set of requirements.*

*Example Appendices could include (initial) conceptual documents for the software project, marketing materials, minutes of meetings with the customer(s), etc.*

## **A.0 Glossary**

***lists all the terminology used in this documentation, such as http, admin, student, https, database etc***

## **A.1 Appendix 1 //If any others**

## **A.2 Appendix 2**