



*MIDDLE EAST TECHNICAL UNIVERSITY*  
*COMPUTER ENGINEERING DEPARTMENT*



CENG 491 – Computer Engineering Design – I

## **SMART HOME PROJECT**

### SOFTWARE REQUIREMENTS SPECIFICATION



Abdullah Hasan Taher Bayrakdar

Anıl Ulutürk

Şerafettin Öztürk

Zeynep Mavuş

*Sponsored by*



## Table of Contents

1. Introduction .....	3
1.1 Purpose .....	3
1.2 Document Conventions.....	3
1.3 Intended Audience and Reading Suggestions.....	4
1.4 Project Scope .....	4
1.5 References.....	5
2. Overall Description .....	6
2.1 Product Perspective.....	6
2.2 Product Features .....	6
2.3 User Classes and Characteristics .....	7
2.4 Operating Environment.....	8
2.5 Design and Implementation Constraints .....	8
3. Specific Requirements.....	9
3.1 Non-functional Requirements.....	9
3.1.1 Performance requirements .....	9
3.1.2 Design constraints .....	9
3.2 Functional Requirements .....	9
3.2.1 Choose a Device.....	10
3.2.2 Turn On .....	11
3.2.3Get Sensor Info .....	12
3.2.4 Adjust.....	13
3.2.5 Update Sensor Info.....	14
3.2.6 Turn Off .....	15
4. Behavioral Model and Description .....	16
4.1. Description for Software Behavior.....	16
4.2 State Transition Diagrams .....	17
5. Planning .....	18
5.1 Team Structure.....	18
5.2 Basic Schedule.....	19
5.3 Process Model.....	20
6. Conclusion.....	21

# 1. Introduction

## *1.1 Purpose*

This document details the software requirements specification for the SMARTHOME project. When explaining the details, the IEEE standards for software requirements specification documents are followed.

Home automation systems provide certain functionalities for variety of devices, but many of them cannot extend their ability to respond technology which changes very quickly. In order to achieve this, a generic set of functionality and support for various home appliances needs to be generated. An easy to use, easy to deploy system with an ability to learn and predict home owners' and residents' activities based on previous knowledge will provide a more intelligent way of handling our homes.

SMARTHOME project is intended to serve this purpose and project details will be provided in the following sections.

## *1.2 Document Conventions*

SMARTHOME project's aims are currently not strictly defined. Many of the requirement specifications and use cases provided in the version 1.0 of this document are merely a starting point and will provide a perspective for the intended purpose of this project.

Technical details related to hardware and software systems and interfaces are not included in this document in detail. Regarding information and sources will be provided in the appendix sections at the end of the document.

### *1.3 Intended Audience and Reading Suggestions*

This Software Requirements document is intended for:

- Developers who can review project's capabilities and more easily understand where their efforts should be targeted to improve or add more features to it.
- Project testers who can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized.
- End users of this application who wish to read about what this project can do.

### *1.4 Project Scope*

SMARTHOME is a simple but quite versatile home automation system which consists of four tightly connected subsystems.

- Environment and device sensors collection
- Wireless network
- Coordinator Box
- User Interface

As home appliances provide various data from their own work conditions and status, sensor collection provided as a generic device analyzes the data and sends it through wireless network to coordinator box.

Receiving the data through local wireless system, coordinator box collects and stores the data on an online server. According to users' choices or previous statistics, it can act as a smart agent to provide certain reflexes in case of emergencies and possible threats.

All scenarios of actions and status information can be viewed through internet (web server) in which statistical data are accumulated. As well as viewing, user can send orders to the coordinator box in order to change home appliances' activities in real time.

### *1.5 References*

- Project’s development and distribution website at [ceng.metu.edu.tr](http://ceng.metu.edu.tr) local repository. It provides the project’s source code, a bug reporting and tracking system, and all the available file downloads of the project.
- IEEE Software Requirement Specifications document.

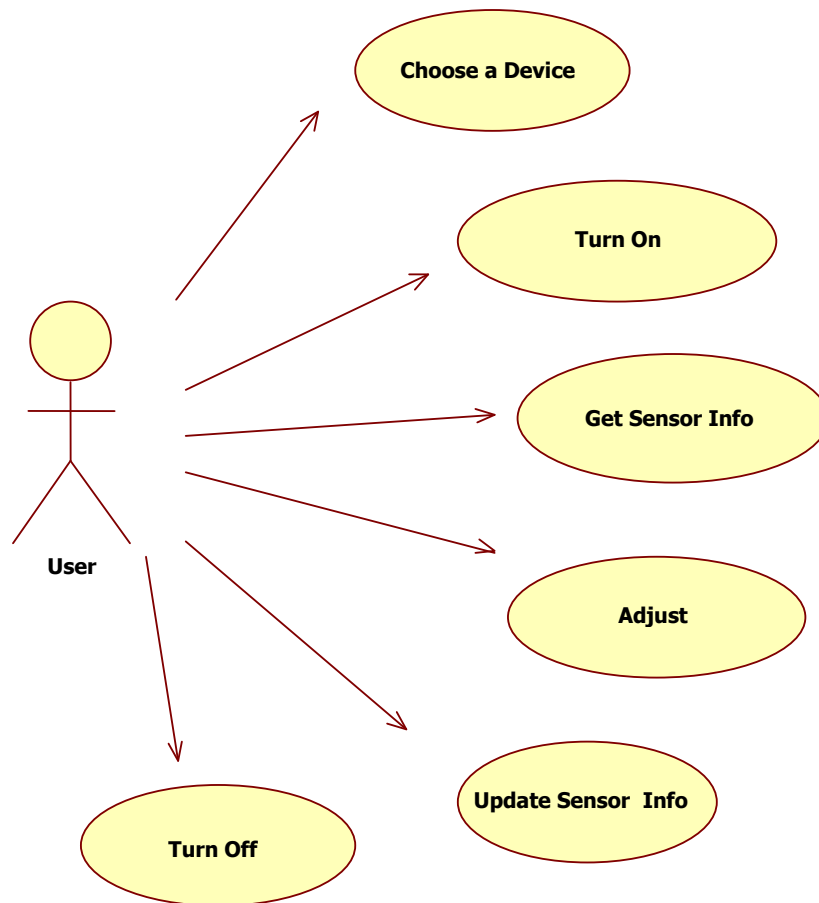
## 2. Overall Description

### *2.1 Product Perspective*

SMARTHOME is a generic solution for home automation enhanced with machine learning techniques. It uses a well-known wireless platform Zigbee for communication between devices and allows you to collect data and control home appliances from a single web interface. Main program running on Coordinator Box is open source with a GNU General Public License. In market, there are lots of home automation systems, but they lack the adaptability and generic way of application in variety of home appliances. Moreover, since SMARTHOME is open source; it allows collaboration of developers for new enhancements to the project.

### *2.2 Product Features*

In our home automation system, the user indicates the person purchasing the product to take advantage of energy efficiency for his/her home. The user basically has rights to choose a device, turn it on & off, getting & updating the sensor info of the device and adjust it according to the configurations types of the device. Their main functionalities are shown by the below use case diagram.



**Figure 1: Block diagram**

### *2.3 User Classes and Characteristics*

This project is intended to be used by various of user classes. These classes can be listed as follows:

1. Home owners:

– User interface provided by SMARTHOME is easy-to-use and user friendly, in turn allows an average computer user to control home appliances they own.

2. Open source software developers and contributors:

– People with good knowledge of Python programming language can contribute to the main program which runs on Coordinator Box

- Also, network programmers and web designers can easily extend features provided by web interface.

### 3. Software Engineers & Computer Scientists involved in machine learning.

- Statistical data collected by coordinator box can be used in various cases of everyday situations in an efficient way. Although we implemented substantial concepts regarding an ordinary home, people who are experienced in machine learning can always provide new ways of adaptation for SMARTHOME.

## *2.4 Operating Environment*

In this project, ZigBee network devices would be used to obtain a wireless communication between a master controller and home appliances as well as various sensors to collect data from home environment.

A master controller software working on ARM Based BeagleBoard will be developed. In addition, a database deployed on web server would be needed to keep the data collected from home appliances on the web and show these data when required.

PIC Microcontroller Board would also be used to simulate home appliances within the development process by imitating the response of home devices.

## *2.5 Design and Implementation Constraints*

- Smarthome system is a platform dependent. Embedded Linux working on a BeagleBoard device or PC with linux OS is a necessity.
- Work products such as documents will be in compliance with IEEE standards.

[Refer to 3.3.2 Design constraints]



## **3. Specific Requirements**

### ***3.1 Non-functional Requirements***

#### **3.1.1 Performance requirements**

The resulting home automation system should perform on home appliances and various sensors having the ability of compatibility more than the ones in the market. Additionally a more general communication protocol and wireless controller hardware should be used to make the system perform on much more devices. The system should access data in reasonable time. The data transfers between the devices such as actuators and sensors with master controller should not exceed the time limit of 3 seconds and lie under the throughput of 250 Kbps. Additionally, the system should service with the 7 days / 24 hours availability. The system should work smoothly with other existing connection networks at home.

#### **3.1.2 Design constraints**

The reporting of the project should be in IEEE standards and its diagrams should be drawn in UML standards. The interface between the system components should be well described to make the user control easier. As an environment constraint, master controller software should be developed on Linux system. Moreover, there is another constraint on wireless communication protocol. In this system, ZigBee wireless protocol should be used to make the devices communicate. The transmitted information between the devices should be carried in encrypted form, as a security constraint.

### ***3.2 Functional Requirements***

In order to make the user manipulate the system, we need 5 general functions working on a home appliance included in the home automation system with the help of sensors and required connections. Each function is explained with the use cases below.

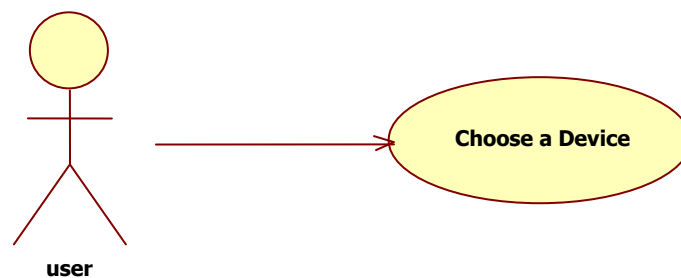
### 3.2.1 Choose a Device

#### 3.2.1.1 Background Information

There will be different kinds of home appliances included in the home automation system.

#### 3.2.1.2 Action/Response Sequences

##### 3.2.1.2.1 Diagram



##### 3.2.1.2.2 Description

Actor	User
Purpose	The aim of this function to enable the user to choose a device from the context.
Precondition	The system should be on mode and the user should start the application.
Trigger	The user should choose a device from the menu by clicking on it.

##### 3.2.1.2.3 Normal Flow of Events

1. The user selects the device and sees its current condition.
2. The user can manipulate the appliance with given commands.

##### 3.2.1.2.4 Alternative Flow of Events

1. The user tries to use other functions before choosing a device.
2. The system gives a warning message.

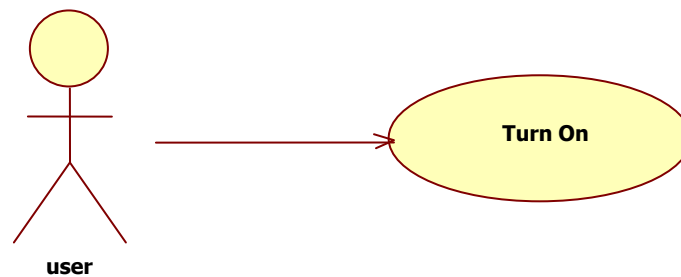
### 3.2.2 Turn On

#### 3.2.2.1 Background Information

There will be a selected device to perform the action on before the user requests this action.

#### 3.2.2.2 Action/Response Sequences

##### 3.2.2.2.1 Diagram



##### 3.2.2.2.2 Description

Actor	User
Purpose	The aim of this function to enable the user to turn on the chosen device.
Precondition	The user should select a device.
Trigger	The user should click on the turn on button.

##### 3.2.2.2.3 Normal Flow of Events

1. The user clicks on the turn on button and waits for the system to start it.
2. The user turns on the appliance with the default adjustments.

##### 3.2.2.2.4 Alternative Flow of Events

1. The user tries to turn off or wants to make adjustments on the device before using this functionality.
2. The system makes a request to make the user turn on the device, initially.

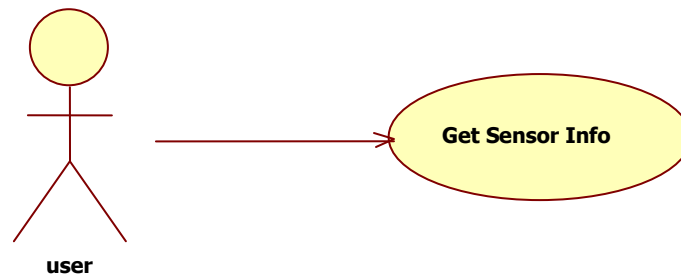
### 3.2.3 Get Sensor Info

#### 3.2.3.1 Background Information

There will be sensors placed into the home appliances to perform this action.

#### 3.2.3.2 Action/Response Sequences

##### 3.2.2.3.1 Diagram



##### 3.2.2.3.2 Description

Actor	User
Purpose	The aim of this function to enable the user to learn info about the device from the sensors.
Precondition	The device should have atruely working sensor connected to the system.
Trigger	The user should click on the 'get the current condition' button.

##### 3.2.2.3.3 Normal Flow of Events

1. After clicking on the above mentioned button, the system starts to communicate with the device.
2. A window appears on the screen to make the user learn the condition of the device.

##### 3.2.2.3.4 Alternative Flow of Events

1. The sensor data may not be transferred properly.
2. A warning should appear on the screen indicating that the info could not be reached by the system and request the user to try again.

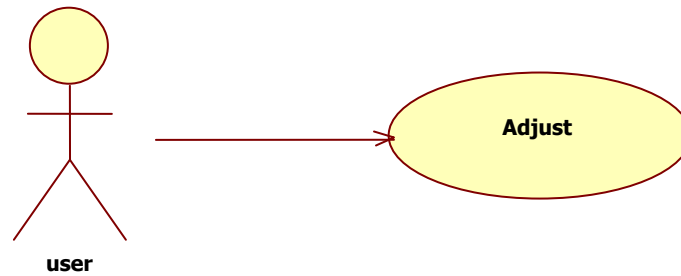
### 3.2.4 Adjust

#### 3.2.4.1 Background Information

Well described adjustment panels on the interface will be provided for each device included in the system.

#### 3.2.4.2 Action/Response Sequences

##### 3.2.2.4.1 Diagram



##### 3.2.2.4.2 Description

Actor	User
Purpose	The aim of this function is to enable the user to make the adjustments according to the device capabilities.
Precondition	The system should switch on the true adjustment menu according to the chosen device and make sure that sensor info is not obsolete by updating the sensor info.
Trigger	The user should enter the adjustment type and content into the adjustment panel.

##### 3.2.2.4.3 Normal Flow of Events

1. This functionality is device dependent, so generally it can be stated that the user should click on the adjust button.
2. The device dependent adjustment panel appears on the screen.

##### 3.2.2.4.4 Alternative Flow of Events

1. The user can enter ambiguous adjustments into the panel.
2. The system should give a warning, and continues with the previous adjustments.

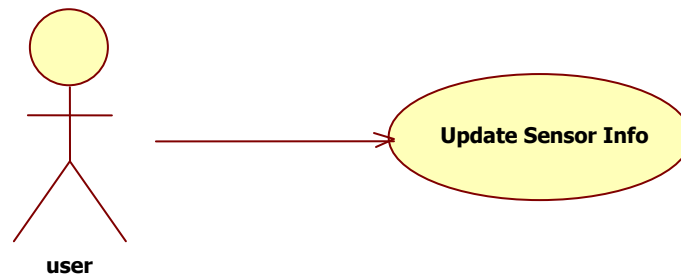
### 3.2.5 Update Sensor Info

#### 3.2.4.1 Background Information

There will be sensors placed into the home appliances to perform this action.

#### 3.2.4.2 Action/Response Sequences

##### 3.2.2.4.1 Diagram



##### 3.2.2.4.2 Description

Actor	User
Purpose	The aim of this function is to enable the user to update the sensor info anytime s/he wants.
Precondition	The device should have a truly working sensor connected to the system.
Trigger	The user should click on the update button.

##### 3.2.2.4.3 Normal Flow of Events

1. The user clicks on the update button.
2. The system updates the sensor info beside the periodic updating times of the system.

##### 3.2.2.4.4 Alternative Flow of Events

1. This command can conflict with the periodic update times.
2. The system continues with the periodic updating without making extra function call.

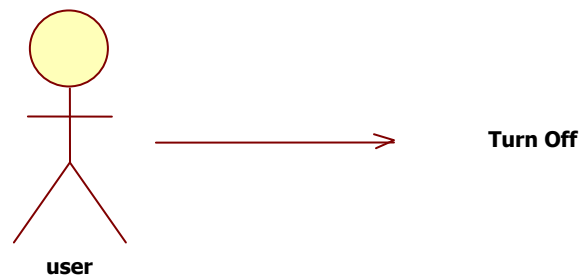
### 3.2.6 Turn Off

#### 3.2.4.1 Background Information

There will be a selected and powered on device to perform the action on before the user requests this action.

#### 3.2.4.2 Action/Response Sequences

##### 3.2.2.4.1 Diagram



##### 3.2.2.4.2 Description

Actor	User
Purpose	The aim of this function is to stop the working of the chosen home appliance.
Precondition	The device should be turned on.
Trigger	The user should click on the turn off button.

##### 3.2.2.4.4 Normal Flow of Events

1. The user clicks on the turn off button after selecting the device.
2. The system powers the device down and inform the user.

##### 3.2.2.4.4 Alternative Flow of Events

1. The user can try to turn off a device without turning on it before.
2. The system creates a warning message to display on the screen.

## 4. Behavioral Model and Description

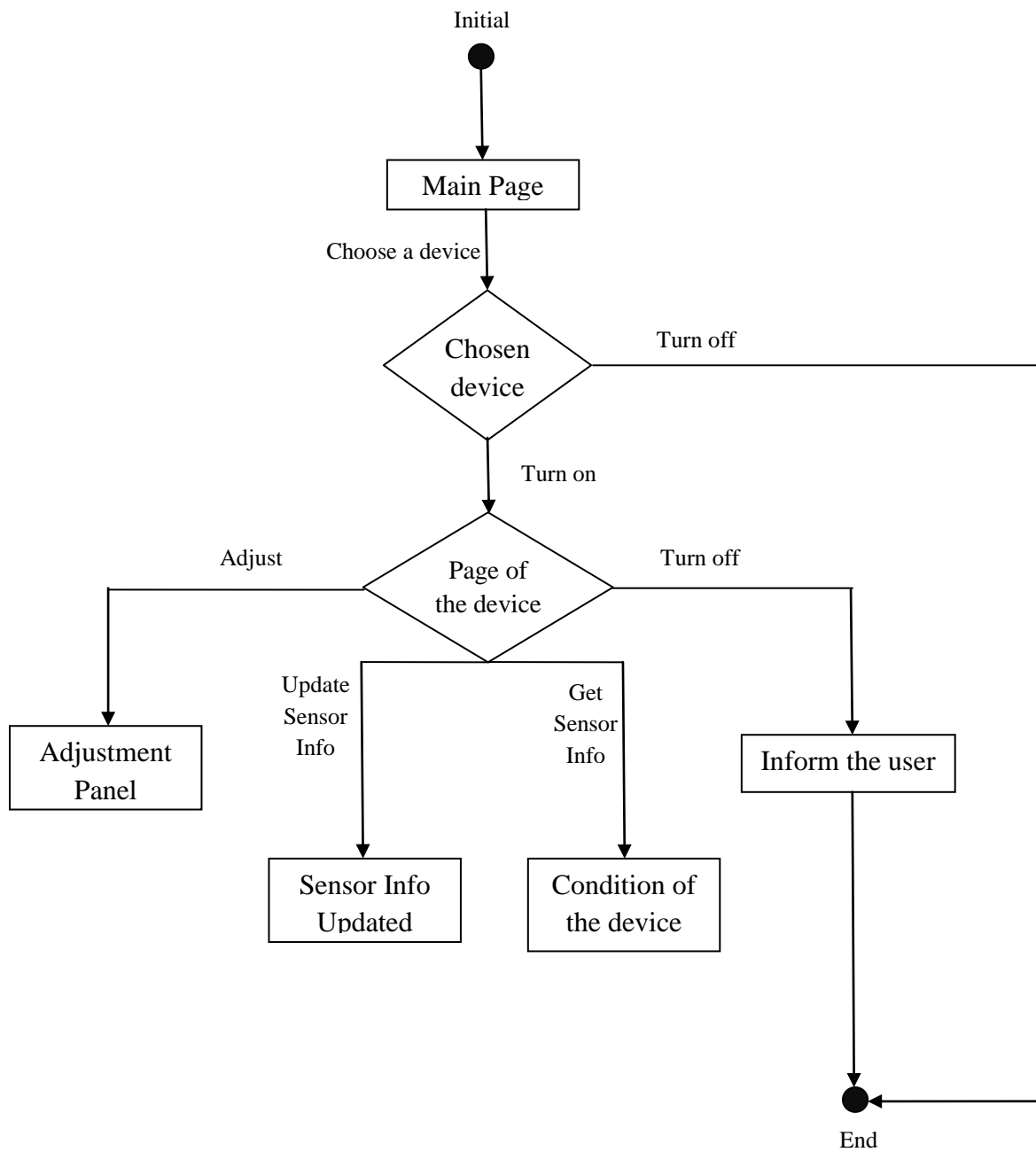
### *4.1. Description for Software Behavior*

When user opens the Web based GUI from a browser window, he/she sees a main page with a list of devices connected to the coordinator box, with few limited details about their current status. From this list, user can select the device he/she wants to control or view and reach a detailed page for that specific device. Every home appliance has different features which can be controlled and monitored through this system, and all of these applicable information will be provided in the subpage of the selected device.

After a device is selected, user can see all features of that device that can be adjusted through web interface. After editing specific features user can use “Adjust” button to send report about these changed features to home appliances. After a reasonable time for data sending/receiving protocols (around 4-10 seconds in a fast network), the device to be controlled will respond to these changes and sensor information will be updated. User can also click on Update Sensor Info button to order these changes manually. Other than issuing feature changes for appliances, user can also use Turn Off and Turn On buttons in this page to control the device itself and start-stop its progress in real time.



## 4.2 State Transition Diagrams



**Figure 2: State Transition Diagram of the system**

## 5. Planning

### 5.1 Team Structure

Our contact from the sponsor company Arcelik is İhsan Mert Özçelik with whom we discuss about our project process nearly once a week. Since project has interdependent hardware parts that can be combined after certain development phases, we decided to split our team into four. Team structure is listed below with their corresponding responsibilities.

- Abdullah Hasan Taher Bayrakdar - Web GUI Development and Server Base in Linux
- Anıl Ulutürk - Zigbee Wireless Networking and Data Modeling
- Zeynep Mavuş – Embedded Development with connection to Zigbee end devices
- Şerafettin Öztürk - Linux development with connection to Zigbee coordinator devices

Even though project has been split into two parts, each member will have information about each part of the project at the end of development cycles defined in next subsections.

## 5.2 Basic Schedule

The Gantt chart that shows the basic time schedule of the project is as follows with the tasks and dates included.

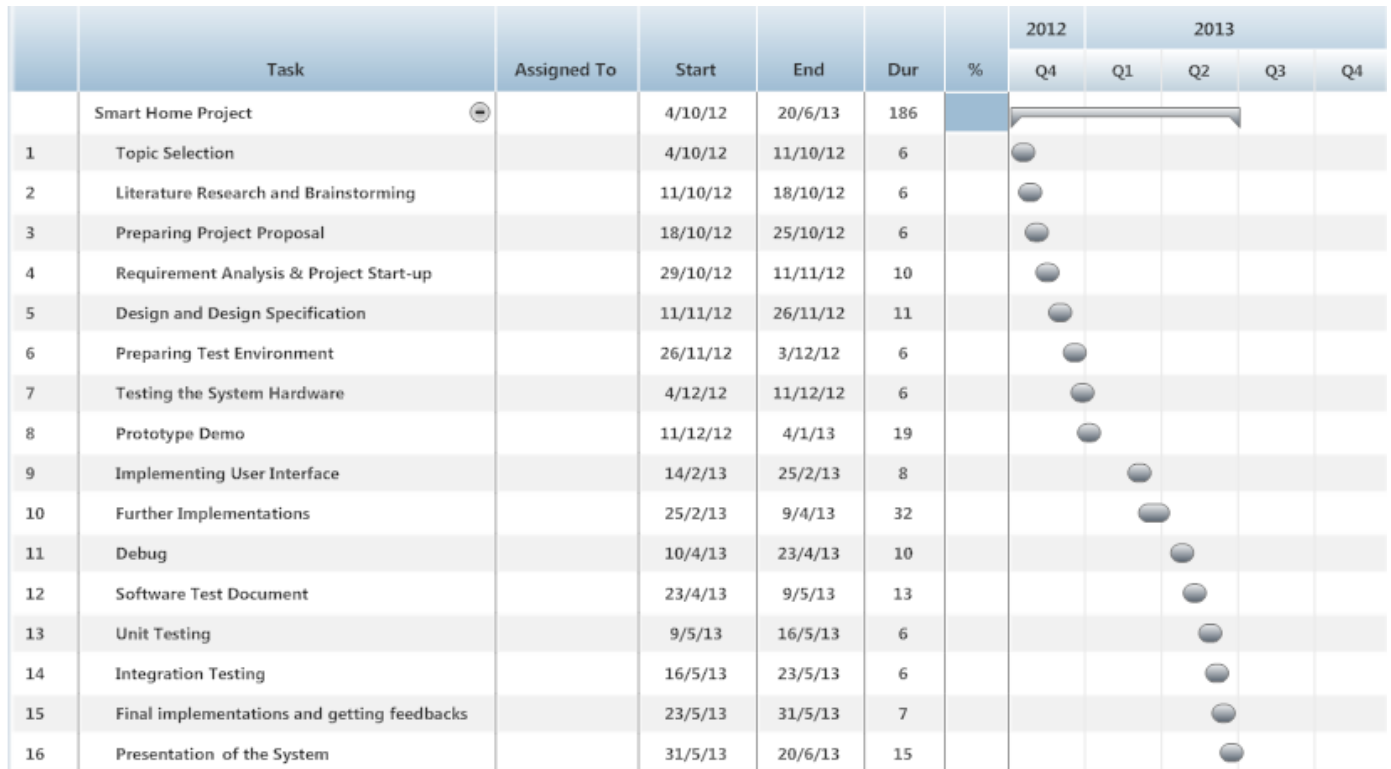
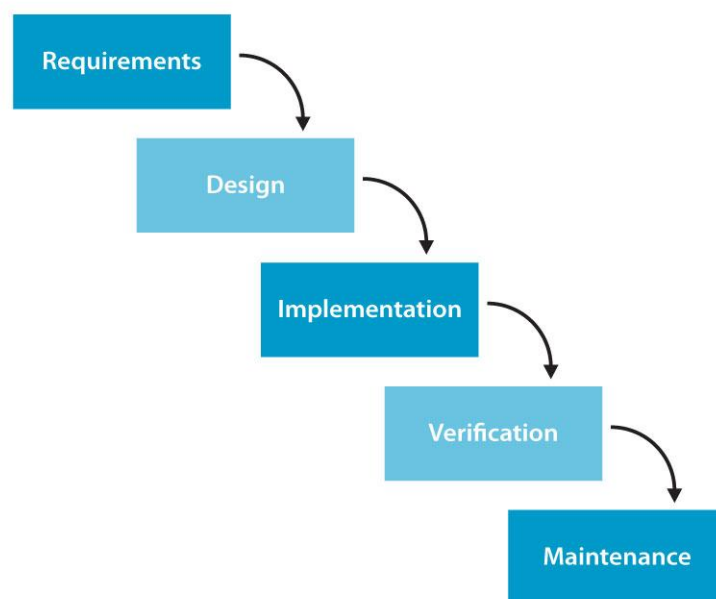


Figure3: Gantt chart

### 5.3 Process Model

Main model we will use in this project is Waterfall Lifecycle. First we have done research about market and currently developed home automation systems. Discussing their pros and cons, we decided on extra features we may add to come up with an interesting design. According to Waterfall model, Requirements Analysis and Specification was the next step. After the completion of the first version of this SRS document we came to end of second step and got started to design process. As many of the project's design choices were not clear, we decided to leave behavioral and data model definitions to be next step in our process model. After design phase, we will continue with implementation. When implementation of two subparts of the project becomes near completion, we will deal with combining, maintenance and testing of full set of requirements in this document.



**Figure4:Waterfall Model**

## 6. Conclusion

This SRS document gives information about the SMARTHOME project which is an easy to deploy home automation system enhanced with machine learning features. At the beginning of this document, problem is defined and similar products in market are compared with our product. Afterwards, functionality of the application, requirements of the application, performance, attributes and design constraints are clarified in following subsections. In the overall description part, all of the functions that this application will perform are explained in detail for developers and future contributors. In the previous part, approximate plan of the project's progress is described tentatively. This document will hopefully constitute the base of design, development, and testing phases of the SMARTHOME project.