**A HOME AUTOMATION SYSTEM**

**18-088**

**Design Document (DD) for Component “Exterior Home Management” a sub- functionality of “ Smart Home Automation System”**

|  |  |
| --- | --- |
| **Name** | **IT Number** |
| O.Y.Navarathneraja | IT15050786 |

B.Sc. Special Honors Degree Department of Information

Sri Lanka Institute of Information Technology Sri Lanka

March 2018

# DECLARATION

I hereby declare that the project work entitled “*Home Automation System*”, submitted to the Sri Lankan Institute of Information Technology is record of an original work done by me, under the immense guidance and supervision of our Supervisor *Mr. Yashas Mallawarachchi.* The Project work is submitted in the partial fulfillment of the requirement for the award of the degree of Bachelor of Science (Special Honors) in Information Technology. The Results embodied in this report have not been submitted to any other University or Institution for the award of any degree or diploma. Information derived from the published or unpublished work of others has been acknowledged in the text and a list of references is given.

……………………

*O.Y.Navarathneraja IT15050786*

**Table of Contents**

[DECLARATION 2](#_bookmark0)

[Introduction 5](#_bookmark1)

* 1. [Purpose 5](#_bookmark2)
  2. [Scope 5](#_bookmark3)
  3. [Definitions, Acronyms, and Abbreviations 5](#_bookmark4)
  4. [Overview 6](#_bookmark6)

1. [Overall Descriptions 6](#_bookmark7)
   1. [Product Perspective 7](#_bookmark8)
      1. [System interface 7](#_bookmark9)
      2. [User interfaces 8](#_bookmark11)
      3. [Hardware interfaces 8](#_bookmark13)
      4. [Software interfaces 9](#_bookmark14)
      5. [Communication interfaces 9](#_bookmark15)
      6. [Memory constraints 9](#_bookmark16)
      7. [Operations 10](#_bookmark17)
      8. [Site adaptation requirements 10](#_bookmark18)
   2. [Product functions 10](#_bookmark19)
      1. [Automating and Controlling the Functionality of the Garage Gate and Light. 10](#_bookmark20)
      2. [Automation and Controlling the Landscape Lighting, per Natural Light. 14](#_bookmark25)
   3. [Constraints 17](#_bookmark30)
   4. [Assumptions and dependencies 17](#_bookmark31)
   5. [Apportioning of requirements 17](#_bookmark32)
2. [Specific requirements 18](#_bookmark33)
   1. [External interface requirements 18](#_bookmark34)
      1. [User interfaces 18](#_bookmark35)
      2. [Hardware interfaces 18](#_bookmark36)
3. [Supporting information 20](#_bookmark37)
   1. [References 20](#_bookmark38)
   2. [Appendices 20](#_bookmark39)

Tables

[Table 1 : Acronyms 5](#_bookmark5)

[Table 2 : Use case - To Automate the Functionality of the garage 11](#_bookmark21)

[Table 3 : Use case To Automate the Functionality of the Garage Light 12](#_bookmark22)

[Table 4 : Use case- To Automate the Functionality of the Landscape Lighting 14](#_bookmark26)

Figures

[Figure 1: System Interaction with Hardware and Software 7](#_bookmark10)

[Figure 2 :Exterior Home Management Interface 8](#_bookmark12)

[Figure 3 : Wiring the 2 – channel replay 13](#_bookmark23)

[Figure 4 : Connecting with the Garage Sensor 13](#_bookmark24)

[Figure 5 : Light Sensor Module 15](#_bookmark27)

[Figure 6 : Photo-resistor / LDR 15](#_bookmark28)

[Figure 7 : Connecting the Module with the Raspberry Pi 16](#_bookmark29)

# Introduction

## Purpose

The purpose of Software Design Document is to present a detailed description of the Designs of the Home Automation System, which is created to ensure users their Home is Safe. The Design Document provides the designs used by designers to upgrade or modify the current design of the system and by the developers as guidelines to implement the project.

## Scope

This document gives an elaborated description of an individual feature of the Home automation system, the Exterior Management of the Home. It specifies the structure and design of the exterior components in the House, and how the software communicates with the micro-controllers.

This section of the system, provides the user with an automated Garage controlling system; as well as the ability to control the opening and closing of the gate remotely. Furthermore, this section handles the landscape lighting, when the natural limit becomes limited, the porch light automatically turns on; and accordingly, it switches off when dawn approaches.

## Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| SDD | Software Design Document |
| UI | User interfaces |
| API | Application programming interface |
| RAM | Random-access memory |
| LDR | Light dependent resister |
| IEEE | Institute of Electrical and Electronics Engineers |
| VCC | Voltage Common Collector (IC Power supply pin) |
| GND | Ground (Earth) |
| GPIO | General Purpose Input/output |

**Table 1 : Acronyms**

## Overview

This document is written per the standards for Software Design Documentation explained in “*IEEE Recommended Practice for Software Design Documentatio*n”.

The first section in this document discusses the designs for the System, using Use case diagrams; the next section shows samples of UI designs.

The rest of the individualized document will discuss on the Product perspective, the User Characteristics, the Constraints that will limit the development implementation, Assumptions made prior to the development; furthermore, since our system interacts with hardware components, the Design Document will consist of the Specific requirements for the Embedded system oriented project which discuss on the descriptive interfaces such as system interface, user interface, Hardware and Software interfaces, etc., next we move on to the Architectural Design which describes the High level diagram, the hardware and software requirements, the Risk Mitigation Plan that would be taken if there is a system failure, and the Cost benefit Analysis; Finally the System software attributes are discussed in the document.

# Overall Descriptions

**Image processing and IOT server communication**

This Functionality through Image Processing, will detect the home resident’s license plate number and automatically open the garage gate when the vehicle is couple of meters from the garage. Furthermore, once the car has entered the garage, it will monitor the distance and once it’s in a safe distance, the gate sensor will close the Garage Gate. Hence saving ample time, and effort of the user.

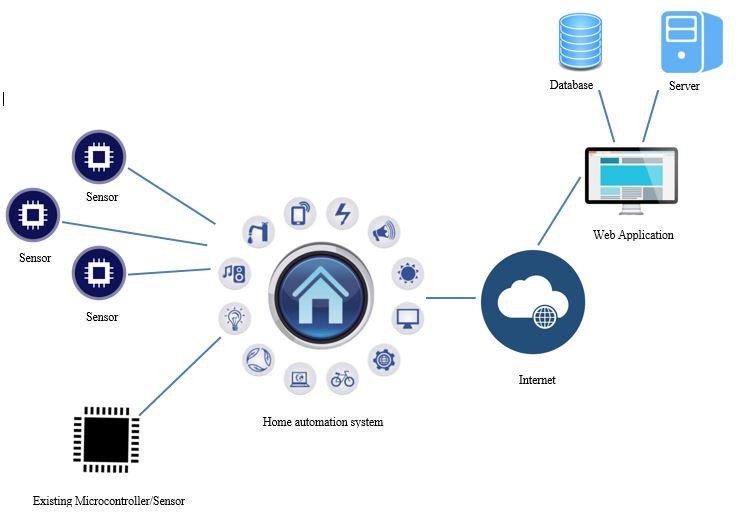
That is not all that our system will handle, Through UV light sensor detection, a light outside the Home (Porch Light) will switch on when it gets closer to dusk (Evening). This light detection mechanism will also handle the lighting system inside the Garage, i.e. if there is limited light supply inside the garage, and if a person or the vehicle is entering the garage, a light will be turned on; this would not only save the person some time from finding the light switch but also avoid causing any injury.

If the Gate is left unlocked, or the Light was not switched off; Alert notifications are sent to the Owners Phone and henceforth, close or switch it off.

## Product Perspective

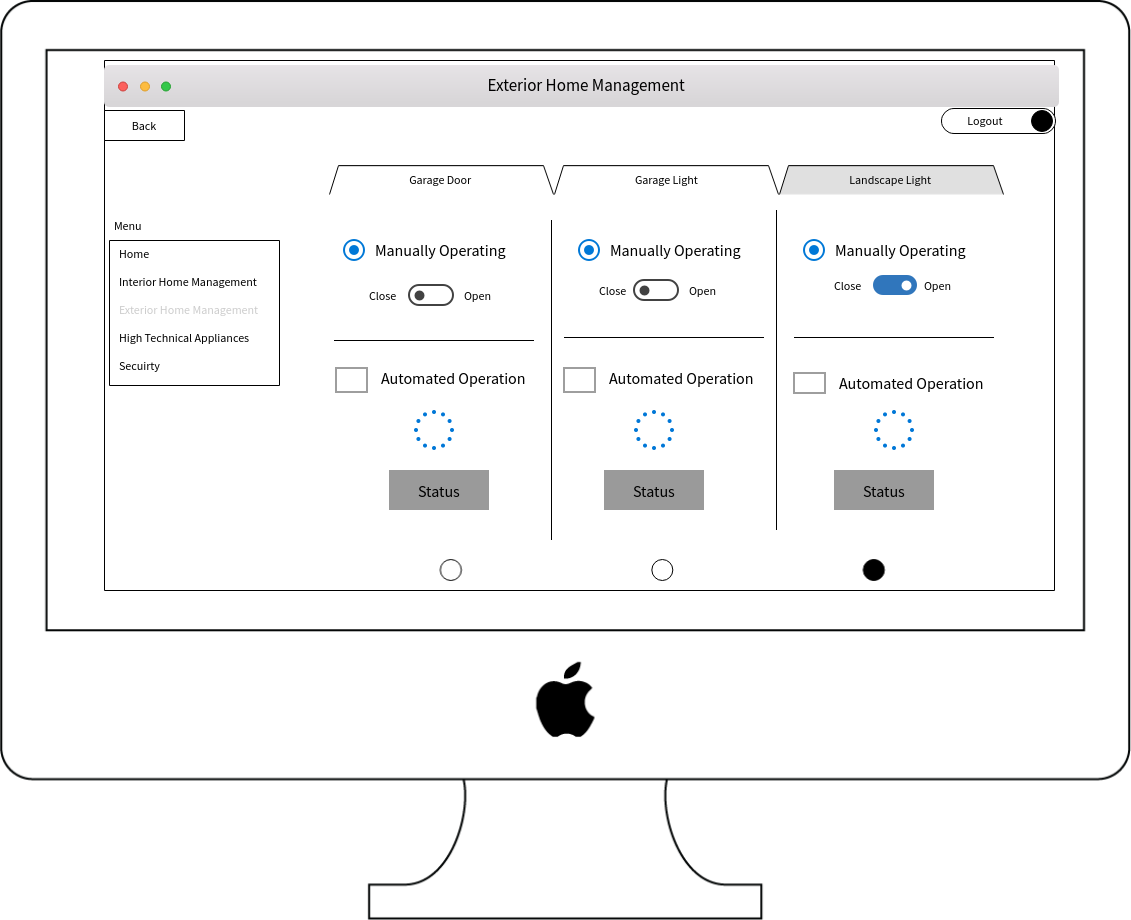
### System interface

This diagram, allows the reader and user of the document to orient themselves to the design and flow of the functions.



**Figure 1: System Interaction with Hardware and Software.**

### User interfaces



**Figure 2 :Exterior Home Management Interface.**

### Hardware interfaces

Hardware Required for the implementation of the sub-function “Exterior Home Management” the Sub-functionality of “Smart Home Automation System” :

* Raspberry pi
* Light Sensor module
* Photo-resistor
* Capacitor
* Garage Sensor
* Breadboard
* Relay
* Wires

### Software interfaces

Software’s Required for the sub-function “Exterior Home Management” the Sub- functionality of “Smart Home Automation System”:

* Apache
* MY SQL
* Bootstrap 3
* VNC Viewer
* WebStorm

### Communication interfaces

Software’s Required for the sub-function “Exterior Home Management” the Sub- functionality of “Smart Home Automation System”:

* Database Connection
* Internet Connection
* Server Connection
* HTTP protocol use to communicate with the cloud since the system handles its data in a cloud.

### Memory constraints

Memory requirement for implementing the functionalities:

* WebStorm Hardware requirement is:
  + 1 GB RAM Minimum
* Apache Server requirements is:
  + 512 MB RAM
  + 20Gb IDE disc

### Operations

1. User is logged into to the system, using the given credentials
2. User navigates to the Exterior Home Management section.
3. User can Control the Functionality of the Garage Gate through the application manually.
4. User can Control the Functionality of the Landscape Lighting as desired through the application manually.

### Site adaptation requirements

Site Adaptations Required for “Smart Home Automation System”:

* Application must always up and running in the cloud. User must have Internet connection to control home from anywhere at any time.
* Raspberry pi should have a power supply all the time.

## Product functions

### Automating and Controlling the Functionality of the Garage Gate and Light.

This Component is a sub component of the Exterior Management System module.

For this functionality to work, first the camera and hardware component (sensor and Micro-controller) must be connected in its relevant places.

A minimum of two cameras must be fixed, to monitor the vehicle; the first camera (Camera A) must be placed outside the garage, and the second camera must be placed inside the garage.

|  |  |
| --- | --- |
| ID | EM 01 |
| Title | Automating Garage Door |
| Description | The Garage Door will automatically Open when the Vehicle approaches |
| Actor | System , User |
| Pre-condition | Owner vehicle license plate is saved. Hardware components are connected. |
| Post-condition | When Owners vehicle is close by, the Garage door opens |
| Main Success Scenario | 1. Camera captures the approached vehicle 2. Compares with saved license plate number 3. Garage door opens for the vehicle 4. Vehicle enters the Garage 5. Camera monitors the distance 6. Compares with the set threshold 7. Garage Door Closes |
| Extension | 2a. The approaching vehicle is not the Home Owners Vehicle 2a.1 The Garage Door remains closed  6a. The threshold is not safe.  6a.1 A Beep sound notification is sent to the application to alert the owner.  6a.1.as the user can manual off the alert. |

**Table 2 : *Use case - To Automate the Functionality of the garage***

|  |  |
| --- | --- |
| ID | EM 03 |
| Title | Automating Garage Light |
| Description | The Garage Lighting will automatically turn On, when the garage door opens and natural light is limited |
| Actor | System , User |
| Pre-condition | Garage door is opening.  Hardware components are connected. |
| Post-condition | Garage Light is on, when the vehicle is entering the garage. |
| Main Success Scenario | 1. The Owners vehicle is entering the Garage 2. The sensor detects the Limited Light in the Garage 3. The Garage light is turned On 4. The Garage Door closes 5. The Owner leaves the room 6. The Camera Detects there is no-one in the Garage 7. The Light is automatically turned off |
| Extension | 2. A. There is sufficient light in the Room  2. A.1 The Light remains switched off.  5. A. The owner manually turns the Light Off.  5. A.1 The Sensor gets updated on the status of the Light. |

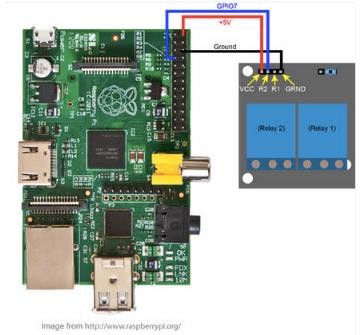
**Table 3 : *Use case To Automate the Functionality of the Garage Light***

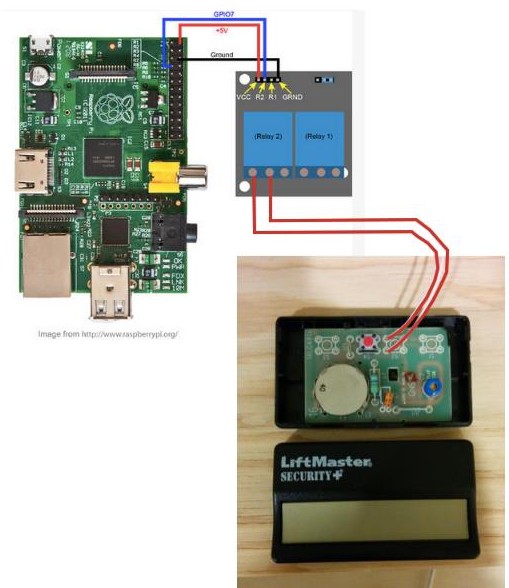
The equipment required to complete the Raspberry Pi Garage Door Sensor is: Raspberry pi

Relay Wires

For quick testing purpose with the GPIO; I connected the pins to an LED, to check if it lights up.

Next, Replace the LED with the relay; reconnect the wires as shown below.



**Figure 3 : *Wiring the 2 – channel replay***

**Figure 4 *: Connecting with the Garage Sensor***

### Automation and Controlling the Landscape Lighting, per Natural Light.

This is another subset component of the Exterior Home Management system for “Home Automation System”.

This Functionality provides the user with automatic landscape lighting, by monitoring the intensity of natural Light.

|  |  |
| --- | --- |
| ID | EM 02 |
| Title | Automating Landscape Light |
| Description | The landscape Lighting will automatically turn On, when natural light is limited. |
| Actor | System , User |
| Pre-condition | Hardware components are connected. |
| Post-condition | Depending on the Natural Light, the Landscape light is switched on |
| Main Success Scenario | The sensor detects the UV Light  In the Evening, When the UV light is low The Landscape light is switched on  During Dawn, when the UV light gets high The Landscape Light is switched off |
| Extension | 1.a When the sensor is blocked or fails , an Alert is sent to the user |

**Table 4 : *Use case- To Automate the Functionality of the Landscape Lighting***

The equipment required to complete the Raspberry Pi Light Sensor is:

* Raspberry Pi
* SD Card
* Wi-Fi dongle
* Light Sensor
* Capacitor
* Breadboard
* Wires

The LDR is the most crucial equipment for this feature to successfully work; without it we would not be able to detect whether the surrounding is dark or light

The purpose of the capacitor is to act as a battery charging up whilst receiving power and discharging whilst there is no power.

The next vital process is building the circuit correctly;

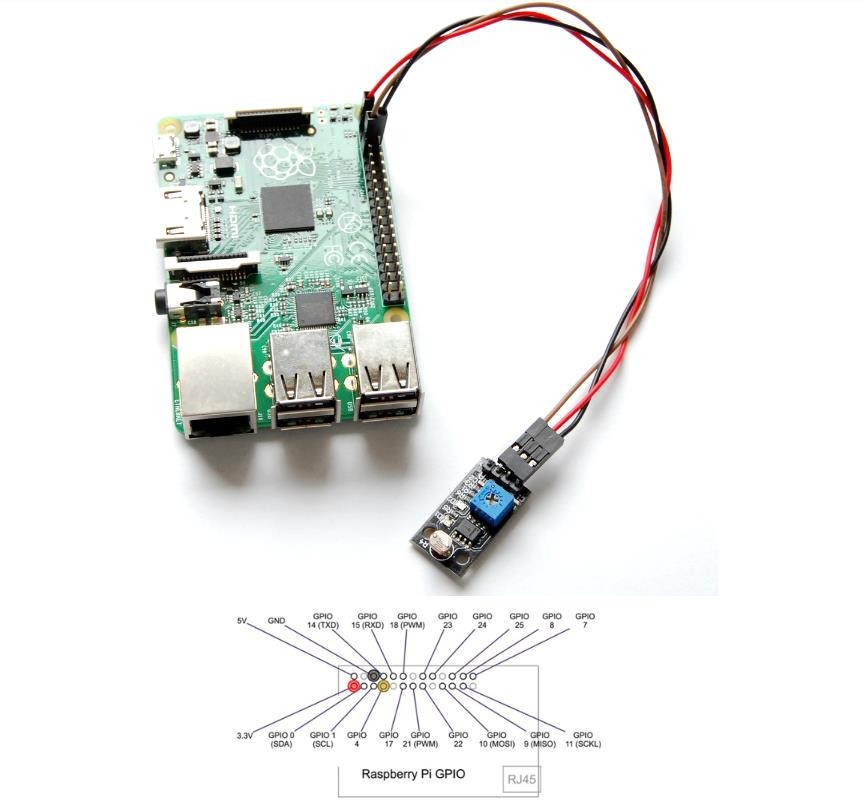
The Light sensor module has 3 wires: VCC, GND, and Signal. We must use the Dupont wires to connect the light sensor module to the GPIO pins on the Raspberry Pi.

The below diagrams will provide a guidance on setting up the circuit.



**Figure 5 *: Light Sensor Module***

**Figure 6 *: Photo-resistor / LDR***



**Figure 7 *: Connecting the Module with the Raspberry Pi***

Raspberry Pi Light Sensor Module 3.3v P1 —————————– VCC (V)

GND P6 —————————- GND (G)

GPIO4 P7 —————————– SIGNAL (S)

Once the Circuit is built correctly, the next process is to code the Raspberry Pi Light sensor.

The initial process before coding may begin is to import the GPIO Package that is required to communicate with the GPIO Pins.

#!/usr/local/bin/python import RPi.GPIO as GPIO

Once the GPIO mode is set to GPIO.BOARD, this means all the numbering we use in the script will refer to the physical numbers of the pins in the breadboard.

Python Coding with Raspberry Pi connects the project to the real word.

To Validate the Feature, Control the Light exposure on the LDR by covering it. The LDR values changes the amount of Light exposed on the LDR.

## Constraints

* Automating Garage Door

Testing on an Actual Garage could be challenging.

* Automating Garage Light

There are no Constraints on this section.

* Automating Landscape Lighting

The possibility of the sensor mal-functioning during Stormy days.

## Assumptions and dependencies

* There should be a strong internet connection for the application to receive alerts and notifications.
* The Mobile device has sufficient storage to handle data
* The modules and components are connected correctly.

## Apportioning of requirements

To begin implementation, the hardware components must be connected and tested.

Once hardware connection is complete, The order of implementation for Exterior Management Functionality sub-functionality of “Home Automation System” is;

automating the Garage Door functionality, Automating the Garage Light and landscape lighting according to natural light.

Henceforth the main priority goes to building the Hardware and next ensuring a successful completion of the above mentioned components.

# Specific requirements

## External interface requirements

### User interfaces

Image processing to detect and capture the License plate number of the residence vehicles.

Open the Garage gate when the vehicle is couple of meters away.

Check if the vehicle is at a safe distance, once inside the garage, and close the garage gate.

Automatically switch on the Home Porch Light, when nearing dusk.

Automatically Switch on the garage light, if the light is limited, when the vehicle is approaching.

The flexibility for the user to manually switch on the Lights or Open the Garage door through the mobile application, without having the hassle of finding the keys or Light switches.

### Hardware interfaces

The Hardware components that the system interacts with is:

* **Photo-Resistor (Light Sensor)**

Also, known as Light-dependent Resistor (LDR) or Photo-cell are considered as light sensitive devices which is used to indicate the presence or absence of light.

In the dark, the Resistance is high, measures up to 1MΩ; but when the Resistor is exposed to Light, the Resistance Drops to about a few *ohms*.

* **Light Sensor Module**

Light Sensor Modules are known as photoelectric devices because it converts light energy (Photons) into electrical (electrons) signals.

Light Sensors is a passive device, and is commonly known as “Photoelectric Device”.

* The Working Voltage is 3.3 – 5 V
* The Output current >= 15 mA, which can directly light up an LED light.
* The Output from the Light Sensor is Digital Switching (High or Low voltage on

*pin D*) and Analog Signals (Voltage Output on *pin A*).

Photoelectric devices are categorized into two main categories:

* Photo-emissive or Photo-voltaic: Devices which Generate Electricity when illuminated.
* Photo-resistors or Photo-conductors: Devices that change their electrical properties.

***Photoconductive Light Sensor*** are Semi-conductor devices that use light energy to control the flow of electrons and henceforth the current flowing through them. They are commonly called ***as Light Dependent Resistors (LDR).***

# Supporting information

## References

1. Light dependent resistors, [Online]

*Available:* <http://www.resistorguide.com/photoresistor/> [Accessed: 10-May-2018]

1. Garage Door operations using Raspberry Pi, [Online]

*Available:* <http://www.instructables.com/id/Raspberry-Pi-Garage-Door-Opener/> [Accessed: 11-May-2018]

1. Light Sensor Module [Online]

*Available*: [http://www.uugear.com/portfolio/using-light-sensor-module-with-raspberry-](http://www.uugear.com/portfolio/using-light-sensor-module-with-raspberry-pi/) [pi/](http://www.uugear.com/portfolio/using-light-sensor-module-with-raspberry-pi/) [Accessed: 6-May-2018]

1. “Raspberry Pi” En.wikipedia.org, [Online]

*Available*: https://en.wikipedia.org/wiki/Raspberry\_Pi [Accessed: 4-May-2018]

1. “Relay” En.wikipedia.org, [Online]

*Available*: https://en.wikipedia.org/wiki/Relay [Accessed: 13-May-2018]

## Appendices

---