DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION

KLARA AZAD KARIM

UNIVERSITI TEKNOLOGI MALAYSIA

UNIVERSITI TEKNOLOGI MALAYSIA

DECLARATION OF THESIS / UNDERGRADUATE PROJECT REPORT AND COPYRIGHT

Author's full name : KLARA AZAD KARIM

Date of Birth : 1/5/1998

Title : DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION

Academic Session : 2022-2023/7

I declare that this thesis is classified as:

☐ **CONFIDENTIAL** (Contains confidential information under the Official Secret Act 1972)*

☐ **RESTRICTED** (Contains restricted information as specified by the organization where research was done)*

☑ **OPEN ACCESS** I agree that my thesis to be published as online open access (full text)

1. I acknowledged that Universiti Teknologi Malaysia reserves the right as follows:
2. The thesis is the property of Universiti Teknologi Malaysia
3. The Library of Universiti Teknologi Malaysia has the right to make copies for the purpose of research only.
4. The Library has the right to make copies of the thesis for academic exchange.

Certified by:

SIGNATURE OF STUDENT	**SIGNATURE OF SUPERVISOR**
QU182SCSJ031	Mr. Mohammed Qadir Kheder
MATRIX NUMBER	**NAME OF SUPERVISOR**
Date: 14 FEB 2022	Date: 14 FEB 2022

May 2023

QIU Library

Sir,

CLASSIFICATION OF THESIS AS OPEN
DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION
KLARA AZAD KARIM

Please be informed that the above-mentioned thesis entitled “DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION” be classified as OPEN ACCESS.

Thank you.

Sincerely yours.

MR. MOHAMMED QADIR KHEDER, As Sulaymaniyah Iraq, +964 751 101 0337

"I hereby declare that we have read this thesis and in my opinion this thesis is suffcient in term of scope and quality for the award of the degree of BSc of Computer Science (Software Engineering)"

Signature	:	
Name of Supervisor	:	Mr. Mohammed Qadir Kheder
Date	:	5 JULY 2023

DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION

KLARA AZAD KARIM

A thesis submitted in fulfilment of the
requirements for the award of the degree of
Bachelor of Computer Science (Software Engineering)

School of Computing
Faculty of Engineering
Universiti Teknologi Malaysia

JULY 2022

DECLARATION

I declare that this thesis entitled *"DESIGN AND IMPLEMENTATION OF SMART HOME SYSTEM WEB BASED APPLICATION"* is the result of my own research except as cited in the references. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	.....
Name	:	KLARA AZAD KARIM
Date	:	14 FEBRUARY 2022

DEDICATION

This thesis is dedicated to my mother because she always supported me to not stop from getting more knowledge and helped me to get to my goals. also, it is dedicated to my father because of his great motivation that motivated me to never stop from getting more knowledges.

ACKNOWLEDGEMENT

During the progress of this thesis, I was get in touched with many people that they helped me so much to complete this thesis. I really want to thank my main thesis supervisor Mr. Mohammed Qadir for his contribution and suggestions to improve my thesis. Another person that I want to thank is my co-supervisor Dr. Mohd Razak Samingan because of his important advises that made this thesis even more powerful. Without their help, suggestion, and advises this thesis would not be that great as it is presented now.

ABSTRACT

As a consequence of different technology beginning to come out, smart homes become one of the affordable options for many people who want to make their life easier. This final BSc project will focus on implementing a smart home model that can be monitored by a mobile application. Using Arduino for this project will be necessary to control all the gadgets inside the smart house. The goal of this final project is to create a smart house that can monitor specific criteria as well as specific devices. The user should be able to control lights, air conditioner, cocking machines (microwave, oven), gas, doors, windows, and water by using a mobile application that is connected with Wi-Fi to be available every time, and everywhere. The application has a friendly interface, which is easy to use and learn. Furthermore, the importance of smart homes is pointed out for those people that are disabled because they can have control over most electrical devices that are available in their house. Finally, this final project has the solution for every problem that we face daily by implementing a smart home and the advantages of the smart home is focused on. For example, having a smart house, you are no longer worry about turning off or on lights when you are away from your house, and you are not coming back home to a cold house because using your phone you can turn on the heating on the way.

ABSTRAK

Akibat teknologi yang berbeza mula keluar, rumah pintar menjadi salah satu pilihan yang mampu dimiliki oleh ramai orang yang ingin menjadikan hidup mereka lebih mudah. Projek BSc akhir ini akan memberi tumpuan kepada pelaksanaan model rumah pintar yang boleh dipantau oleh aplikasi mudah alih. Menggunakan Arduino untuk projek ini diperlukan untuk mengawal semua alat di dalam rumah pintar. Matlamat projek akhir ini adalah untuk mewujudkan rumah pintar yang boleh memantau kriteria tertentu serta peranti tertentu. Pengguna seharusnya dapat mengawal lampu, penghawa dingin, mesin cocking (gelombang mikro, ketuhar), gas, pintu, tingkap dan air dengan menggunakan aplikasi mudah alih yang disambungkan dengan Wi-Fi untuk tersedia setiap masa dan di mana-mana sahaja. Aplikasi ini mempunyai antara muka yang mesra, yang mudah digunakan dan dipelajari. Selain itu, kepentingan rumah pintar dititikberatkan bagi mereka yang kurang upaya kerana mereka boleh mengawal kebanyakan peranti elektrik yang terdapat di rumah mereka. Akhir sekali, projek akhir ini mempunyai penyelesaian untuk setiap masalah yang kita hadapi setiap hari dengan melaksanakan rumah pintar dan kelebihan rumah pintar ditumpukan. Sebagai contoh, mempunyai rumah pintar, anda tidak lagi bimbang tentang mematikan atau menyalakan lampu apabila anda jauh dari rumah anda, dan anda tidak akan pulang ke rumah yang sejuk kerana menggunakan telefon anda, anda boleh menghidupkan pemanas pada cara.

Table of Contents

LIST OF TABLES

TABLE NO. **TITLE** **PAGE**

LIST OF FIGURES

FIGURE NO. **TITLE** **PAGE**

LIST OF APPENDICES

APPENDIX **TITLE**

Chapter 1

INTRODUCTION

1.1. Introduction

These days, almost everyone has a piece of smart devices whether it is a smart TV, smart phone, smart speaker with voice assistance, smart vacuum or a smart watch. Thus, smart is a word that is common and we hear it every day [1]. With the improvement of technology, human wants to use this improvement to make their daily life easier. Each technology developed made differences in the world. For example, you can switch off your lights now with your phone and monitor your house temperature with your mobile. These are some examples of technologies that made our life easier. How about combining all these technologies with other developed technologies to make our home smart? According to [2][3], a smart home is a house that is fitted with communication technologies and can be remotely controlled, or it has some kind of automation. Nowadays, most of companies are trying to produce smart devises that can be used in a smart house because people are interested in them and buying them. They are trying to make people control all the devices inside their house with an application inside your smart phone. Smart home has the advantage of saving time and money for the user.

We are living in a region that most of the people rarely have smart home, people are not familiar and too much cost to implement are reasons behind it. Until now, there is only one small city that they designed to be smart which is “Garden City Villas” [4]. People are using their phone daily, and technology become an increasingly important part of our everyday lives. However, most of the people in the developing countries are not as much familiar with technology as they are able to use it to control their house.

In this project, we need a small house model, Wi-Fi, ESP32, and Arduino microcontroller. The proposed system consists of two main parts. The first part is the web server that has buttons to monitor all the smart devises inside the house that is connected with ESP32 and Wi-Fi. The second part is the hardware Arduino UNO which is a microcontroller board based on the ATmega328P [5]. Arduino allows you to control all the home devices remotely from outside and monitor them from your smart phone. It can control the lighting, fan, door, window, garage door, flame sensor, and temperature sensor.

1.2. Problem background

Nowadays, people are become busy and lazy in life due to working hard for more than 8 to 10 hours per day. After people come back home, they need to sit in relax but, they cannot because they have to turn on all the lights, turn on the air conditioner, TV, and every electronic device. Moreover, many people may forget to turn off the devices after they left home, or they do not remember if they locked the door or not, they have to all way back home to make sure everything is fine or maybe ask a neighbour to make sure that they closed the door or not which is not secure. Finally, there are some families that they have disabled or sick people at home. They cannot leave them alone because they cannot walk and do the actions by themselves. Most of the time those families need someone to take care of them while some cannot since it cost too much for a person to take care of disabled and sick people at home.

Smart home system can solve all these problems that has been mentioned by having a smart home that is connected with a web server you can sit on your sofa or on your bed and control all the devices. Also, you do not need to worry about turning off or locking your door then the web server can work remotely. Furthermore, you do not need someone to watch out your sick or disabled relative at home while you can control whatever they want or you can teach them how the web server work to use it by themselves. The web server is be able to have more than one user.

1.3. Project aim

The aim of this project is to develop a small smart house that can be monitored by a web server to make people live in a comfort and relax life without worrying about switching electronic devices.

1.4. Objectives

1. To study the basic requirements for Smart Home system design and implementation
2. To propose basic architecture and design of Smart Home system in web-based application environment
3. To test the usability and functionality of the developed Smart Home prototype system

1.5. Project Scope

Smart Homes, also known as automated homes. Recently, the use of smart technology has developed so that almost any electrical component within the house can be included in the system.

- Smart home technology does not simply turn devices on and off, it can monitor the internal environment and the activities that are being undertaken while the house is occupied.
- Smart home can monitor the activities of the occupant of a home, independently operate devices in set predefined patterns or independently.
- Turning a regular home to a smart home by designing a model house and implement what is necessary to have in a smart house by using ESP32 and,

Arduino UNO, as shown in Figure 1, and connected with a web based mobile application that can be controlled from a smart phone.

- Database for recording motions that happened while using the system.

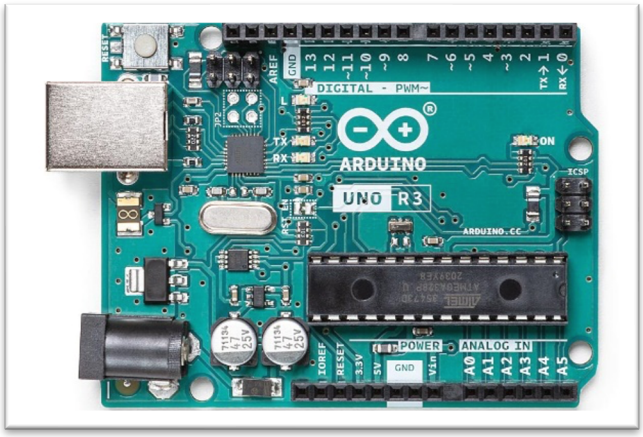


Figure 1-1 Arduino UNO

1.6. Report Organization

Chapter two present a review of the literature related to a smart home with a focus on comparing which system they used, communication interface, controller, user interface, application, and its benefits. Also, provides a deep review of several existing systems. The purpose of doing this literature review is to gain an understanding and familiarity with existing research about the topic. By reading chapter two, you will have a background about the smart home system. While chapter three is about methodology that the system will need to use, by reading chapter three you will be able to how the system will be created and how does it work, also, in chapter three it is discussed which type of Arduino will be used for creating smart home system. Chapter four is all about designing including diagrams and interface. After you read chapter four you can see how the system function and how the application will be look like from the interface. Finally, chapter five is about the summary of the system and why we need a smart home system in our country. Furthermore, the future work of this system has been written.

Chapter 2

LITERATURE REVIEW

2.1. Introduction

Nowadays, increasing in using smart devices to make people life easier, at the same time wanting to change a regular home to a smart home become popular. In this chapter, several existing systems are discussed in terms of software and tools, technologies, and benefits and drawbacks of these systems are also illustrated.

2.2. Review Current Existing Systems

In 2019, a project under the title of "Smart Home Automation System Using Internet of Things" [6], has been published. The study used Arduino microcontroller that is connected with your smart devices. The purpose of implementing this smart home is to save energy, home safety, user convenient, better control, comfortable, and provide peace for the user. Qusay in 2020 [7], published a paper which is focused on warning system inside smart home. The main purposes of this work are the system will show a variety of undesired events that has been noticed efficiently.

For example, fire and gas situations can be distinguished and the user will get notification about them via SMS messages, emails with attached pictures, etc. Besides, some proper actions can also be performed by the planned system including preventing fire via water and reducing gas concentration via air ventilation. The proposed system is very useful to avoid losses in incomes and human life caused by uninvited events. Also, a paper under the name of "Smart Home Automation System using Arduino microcontrollers" [8] has been published by Dan et al [8]. The system aids disabled

people and make their life easier. In 2020, a project under the title of "An IoT-Based smart home Automation system", [9] has been published by Cristina et al. They wanted to make people's life easier by applying smart home using qToggle to control a sequences of home appliances and sensors.

According to [9], the qToggle system is user friendly, flexible, and can be further developed by using dissimilar devices. They implemented the smart home to save valuable time, give control, and save money. Moreover, in 2021, Pawel et al [10], has been published a project under the title of "Smart Home Automation using Arduino UNO Rev2 Microcontroller". The system is to allow the automatization of the home appliances without the use of the internet thus removing the process of data collection and give its users full control over their privacy. Lastly, Akila and Dharmarajan [11] are published research which under the title of "A Low-cost implementation of smart home Automation using Arduino". They used Arduino microcontroller and android application. The benefit of the smart home project is to reduce cost, increase flexibility and it is energy efficient.

2.3. Current System Analysis

Table 2-1 Comparation between existing systems in terms of software, tools, technology with its pros and cons

No	Existing System	Software	Tool (s)	Pros	Cons	Technology
1	Smart Home Automation System Using Internet of Things	Smart devices	Arduino	Energy saving, home safety, user convenient, better control, comfortable, provide peace	Maintenance and repair issues.	Arduino, Power supply, voltage regulator, extension board, ESP8266 Wi-Fi module, flame sensor, temperature sensor
2	Arduino Based Smart Home Warning	SMS-Email	Arduino	Prevent income loses	Difficulty of Setup and Configuration	Arduino, GSM as a wireless communication, Sensors, Cameras

3	Smart Home Automation System using Arduino microcontrollers	Web application	Arduino	Make disabled people life easier	Significant installation costs	Arduino, sensors, Ethernet shield
4	An IoT-Based smart home Automation system	Smart devices	Raspberry Pi	Save money and time	Need license	Raspberry Pi, Internet/Ethernet
5	Smart Home Automation using Arduino UNO Rev2 Microcontroller	Android Application	Arduino	Control over their privacy	This system does not use internet that means it cannot be available every time, everywhere	Wi-Fi, sensor GUI, and Arduino
	A Low-cost implementation of smart home Automation using Arduino			Low cost, flexible, and energy efficient	Not secure, and only used on android system not supporting iOS	Arduino, Wi-Fi

Based on results which presents in the above table (2.1) most of the previous systems used Arduino microcontroller as a tool despite one paper used Raspberry Pi. For the software it is different, some preferred android application, some chosen smart devices, one of them used SMS and emails while the last paper preferred web application. Also, the table 2.1 displays the advantages and disadvantages of several reviewed studies. Overall, the advantages of smart home are more than the disadvantages, although cost and privacy are the most common disadvantages for having smart home. Moreover, building a smart home requires several technologies together. Most of the papers preferred using Wi-Fi, Arduino, cameras and sensors to build a smart home. There is only one reviewed paper preferred Raspberry Pi instead of Arduino and one paper not using the Internet to secure the home.

2.4. Differences between Arduino and Raspberry PI

Table below presents the primary differences between Arduino and Raspberry PI.

No.	Arduino	Raspberry PI
1	• It is a controller from Atmega family.	• It's another type of controller from ARM family [12].
2	• Arduino is a microcontroller.	• is a microprocessor.
3	• It is created to control the electrical devices that is connected to the circuit board in the system.	• It calculates data and produces valued outputs, controls components in a system based on the outcome of its calculation.
4	• It has a simple structure.	• It has a complex structure.
5	• Cheap	• Expensive.

2.5. Chapter Summary

To sum up, having a smart house has several benefits for the owner of the house since it allows the owner to remotely have control over the house in many different sectors. One of them is maximizing the security of the house. Other benefits are controlling home functionalities like lights, doors, and temperature which most of them help to minimize energy use. From the comparison of these papers, it is clear that most of the researchers used Arduino because it is cheap, and it has a simple hardware and software structure. Also, when Arduino is used in a smart house it gives the ability to the user to control the house remotely from their phone. These smart houses depended on an application whether it is an android application or a web-based application to have control over the smart house. This comparison will aid this final project to achieve a foundation of knowledge on the topic and collect some important points about what to use. Also, it helps us to know which software, tools, and technologies are most suitable for the smart house which is planned to develop.

Chapter 3

METHODOLOGY

3.1. Introduction

After reviewing some other researches about this topic and collecting some ideas about what they have done, a smart home was planned to create that will not be redundant to their smart home. This section explains the methodology used to develop the software system. Developing a system is an essential part of software development which is helpful to divide the system development into some phases. For doing methodology in software development of a smart home, the first step is searching for a house plan that has all the living standards and can be converted to an intelligent house. The method of selecting this house plan was after checking to implement the parts that wanted to make it smart. Then, using the house plan, it is decided to construct the house with plywood. This step was about the house's structure and how wisely the house plan was chosen. After that, the critical steps will start. The next step is buying all the sensors, the Arduino UNO, ESP32, and the jumper wires used to connect the sensors and the light with the Arduino UNO. This chapter mentions the creating of the house and establishing the smart system that makes the house a smart house in detail.

3.2. Methodology Choice and Justification

The software development methodology is a process or series of processes used in software development [13]. As stated in [14], the Software Development Life Cycle (SDLC) is used by the software industry to design, develop and test high-quality software. There are six phases involved in software development referring to the SDLC, such as Planning, Analysis, Design, Implementation, Testing and Integration, and Maintenance as shown in Figure 3.1.

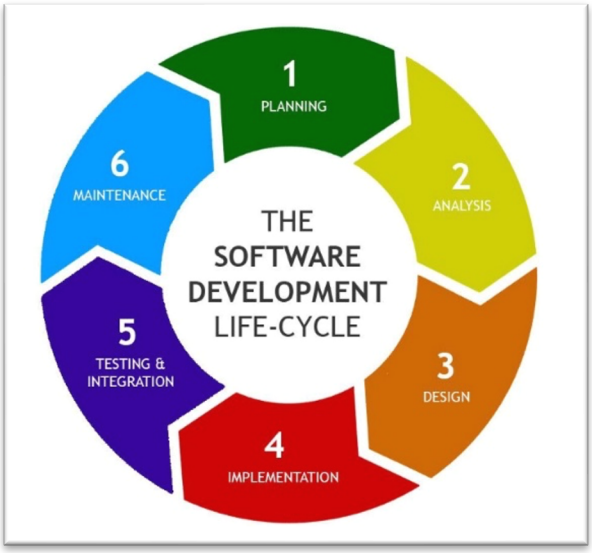


Figure 3-1 The Phases of the SDLC

This project will follow Scrum, a type of agile methodology characterized by cycles or stages of development known as sprints along with waterfall methodology specially for design modelling part. Agile methodology has many advantages, and it is one of the software development methods that are suitable for this project, according to the quality logic website [15].

Agile methodology is used for some reasons. The first reason is that it reduces technical dept to a minimum by adding defects, feature changes, or maintenance tasks to a backlog. Reviewing the team's backlog before planning what to do next makes it easy for the team to get the dept to the minimum. The second reason is that in agile software development the stakeholder is engaged, which is very important for the software to succeed. The third reason is that using agile gives a higher quality to the software because developers are not rushed to finish many tasks. This methodology helps developers to check their work before release and test and address those issues that may exist in the software.

The final reason is that, in agile, what is important is what should be delivered to the customer. User requirements are called user stories in this software development method because users have value. There are many types of agile methodology, but choosing Scrum has its advantages according to the 'Xpand it' [16] website. Focusing

on quality as a constant in the scrum method results in fewer mistakes. Also, the dynamics of this method allow developers to reorganize priorities, ensuring that sprints that have not yet been completed get more attention.

3.3. The Phases within Agile Methodology

Agile software system methodology is divided into six phases that lead the software to success. Those phases are requirement gathering, design, development, testing, deployment, and review.

3.3.1. Phase 1: Requirement Gathering

In this phase, the requirement of the software system should be gathered. Those requirements are what the system should do and contain. Our smart home requires turning on/off lights, locking/unlocking the door, opening/closing the window, and temperature and humidity sensor should read the humidity and temperature of the house, fan should automatically work when the temperature is over 25 Celsius and, fire alarm.

3.3.2. Phase 2: System Design

This phase provides the design of the web-based application interface and the house to implement the smart devices and use it to show people how the smart home works. The application's interface is easy to use, attractive, and easy to learn. HTML codes will be needed for having a web server. In addition, the project needs a small model house to show the work. For the house, we need a house plan, wood to create the house, and some furniture to make it more attractive.

3.3.3. Phase 3: Development and Coding

After finishing the second phase and setting all the plans and designs for the smart home, the application will be developed, and codes will be written. Controlling Arduino microcontroller needs coding so, for programming the Arduino, there is an application called Arduino IDE to write the coding with it also, we have ESP32 we use Arduino IDE application to program it. Moreover, for the web server, HTML must be used because it is a web-based application. Also, at first, the system may not have all the features because we may add new features or solve those problems that may occur. In addition, after finishing each phase, the system should be tested before going to the final testing phase.

3.3.4. Phase 4: Testing

At this step, the smart house structure is ready, and the system is developed. The smart house should work and can be remotely controlled by the web server. Throughout using the application and the system, the bugs and problems should be fixed, so this will give us a system and a smart house where all the futures are working smoothly.

3.3.5. Phase 5: Deployment

In the fifth phase, the smart house and its system development are done ready to use. Having this phase is to update or maintain any features if required from the user.

3.3.6. Phase 6: Review

This is the final stage of this system development methodology. The project is now ready to be presented to the advisors, who will provide feedback on the smart house. Then, either they will agree and have no suggestions for changes which means all the phases are done or offer suggestions for changes or new features which means the software development phases should be repeated.

3.4. Design Modeling

The design modelling provides some diagrams to represent the features of the software and how it works. A flow chart is a diagram that represents a workflow or process. In this section, there are two flowcharts to show how the system works.

1. Open Browser and homeX page to turn ON/OFF LED: It is a diagram (figure 3.2) to show the process of being able to see the home page and turn on/off the light.

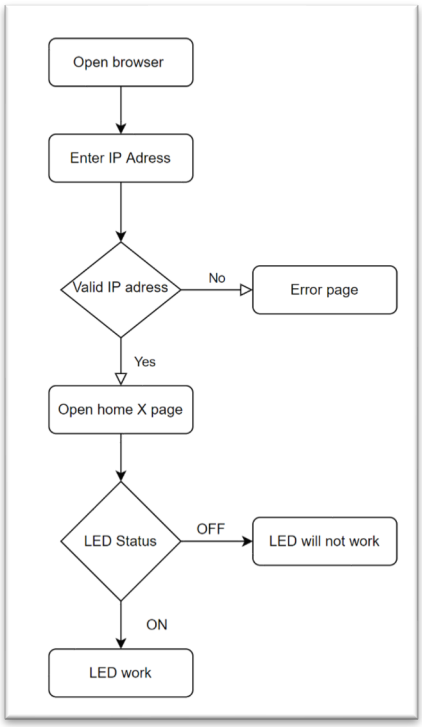


Figure 3-2 Open homeX page and turn ON/OFF LED

2. Flow chart for open/close the door: The second diagram below (figure 3.3) is about opening and closing the door.

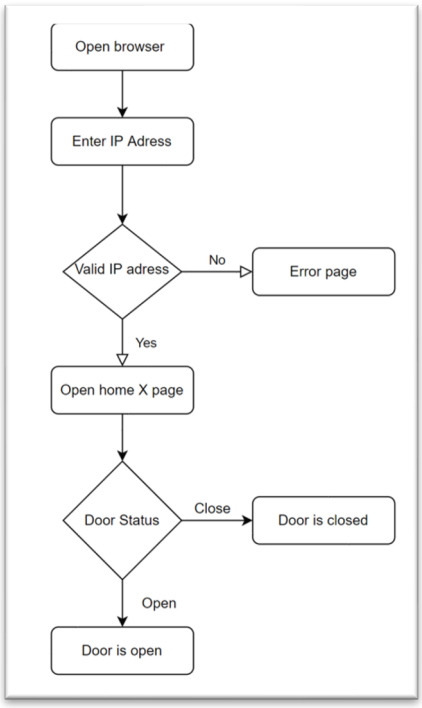


Figure 3-3 Open/Close Door

3. Flow chart for open/close the window: The third diagram (figure 3.4) is about open/close the window in homeX system.

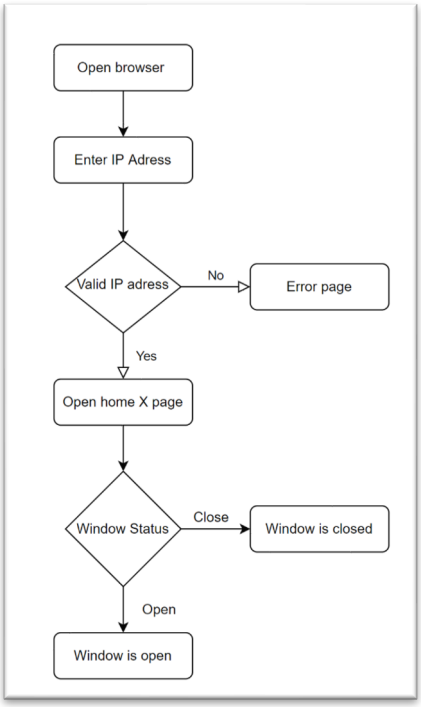


Figure 3-4 Open/Close Window

3.5. PSM Gantt Chart

A Gantt chart is a project management tool to see the dates and deadlines. Also, it provides a visual view of project tasks scheduled over time. The figure below illustrates the timetable for the whole project of the smart home that contains PSM 1 and PMS 2 that starts from November 2021 until February 2023. Figure 3.5.

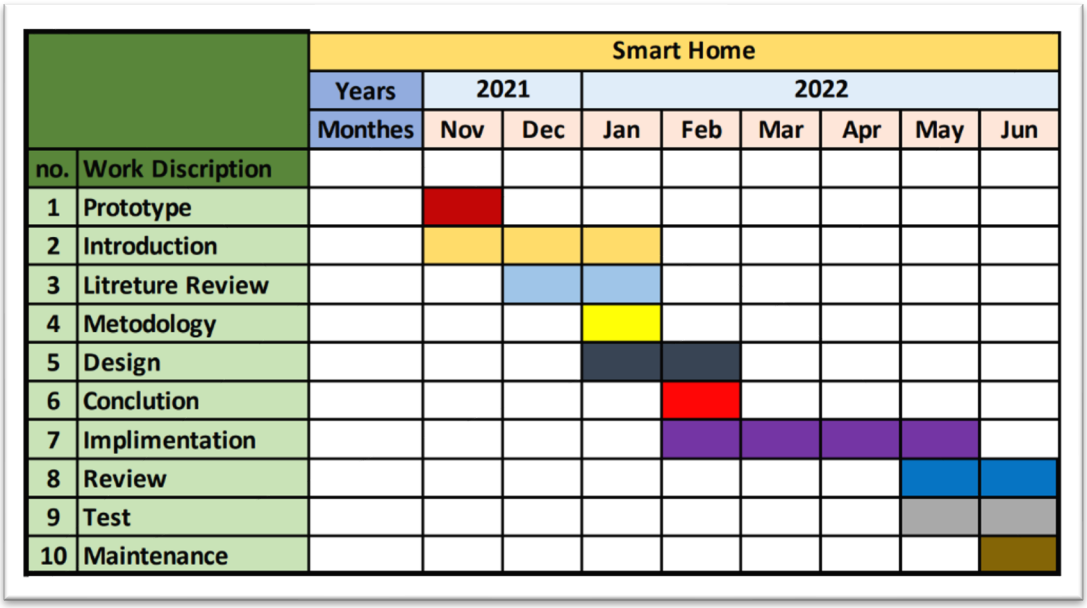


Figure 3-5 Gantt Chart

3.6. Tools and Technology

Table 3.1 provides the tools and technology that are used in implementing the smart home project.

Table 3-1 Tools and Technology

Tools	Technology
Arduino UNO	Wi-Fi, Sensor, Servomotor

3.7. System Requirements

3.7.1. Software Justification

Several software and tools are required to build a smart house, including Arduino IDE, an application for writing Arduino code, and Fritzing, which is an

application for designing the Arduino with its wires. Finally, there is UiZard, which is used for prototyping.

3.7.2. Hardware Justification

For this project, Arduino UNO and ESP32, an open-source microcontroller with input/output pins, is needed. The Arduino breadboard is also used which is a plastic board that electronics can be connected to it and used along with the Arduino UNO. Also, a flame detector sensor is used in hot kitchen that will alert the system if a fire is happening. Furthermore, a servomotor has been used in this project, which is a small motor connected to the Arduino and can produce velocity, is used for opening/closing windows, door, and garage door. Finally, each room has its own LEDs that can be control with the Arduino.

3.8. Chapter Summary

In conclusion, this chapter is mainly about developing the smart home system. There are some different methodologies to choose for developing a software system that each has some advantages and disadvantages. Agile methodology has been chosen for some reasons, and all these reasons were explained. Then, all the methodologies have some phases to follow, and agile has six different phases to follow. The phases of agile methodology are mentioned to explain what we are doing in that face for our smart house project. Then, to develop the smart house system, many different software and hardware are required.

Chapter 4

REQUIREMENTS ANALYSIS AND DESIGN

4.1. Introduction

Chapter three discussed about methodology and all the necessary applications, languages, and tools that the system needs. While this chapter is about designing the system for both web-server and house model. In addition, in this chapter, every diagram that the system needs will be showed to clarify the process of doing each function inside the system such as UML (Unified Modelling Language) diagrams, interface designs, and house model diagram.

4.2. UML Modeling

UML diagrams are those diagrams that help the developers understand and analyze the structural and behavioral of the system [17]. As stated by [17, 18], UML diagrams has many advantages for example, you can use it as a draft, and it will help the developers to visualize programming language. There are two types of UML diagrams which are, structural and behavioral. This chapter contains both types of UML diagram, first the behavioral that includes use case diagram, sequence diagram, and activity diagram.

4.2.1. Use Case Diagram

Use case diagram is a diagram that shows the user's interaction with the system [18]. The smart home system use case diagram shown in Figure 4.1, which has three actors: user, ESP32, and Arduino.

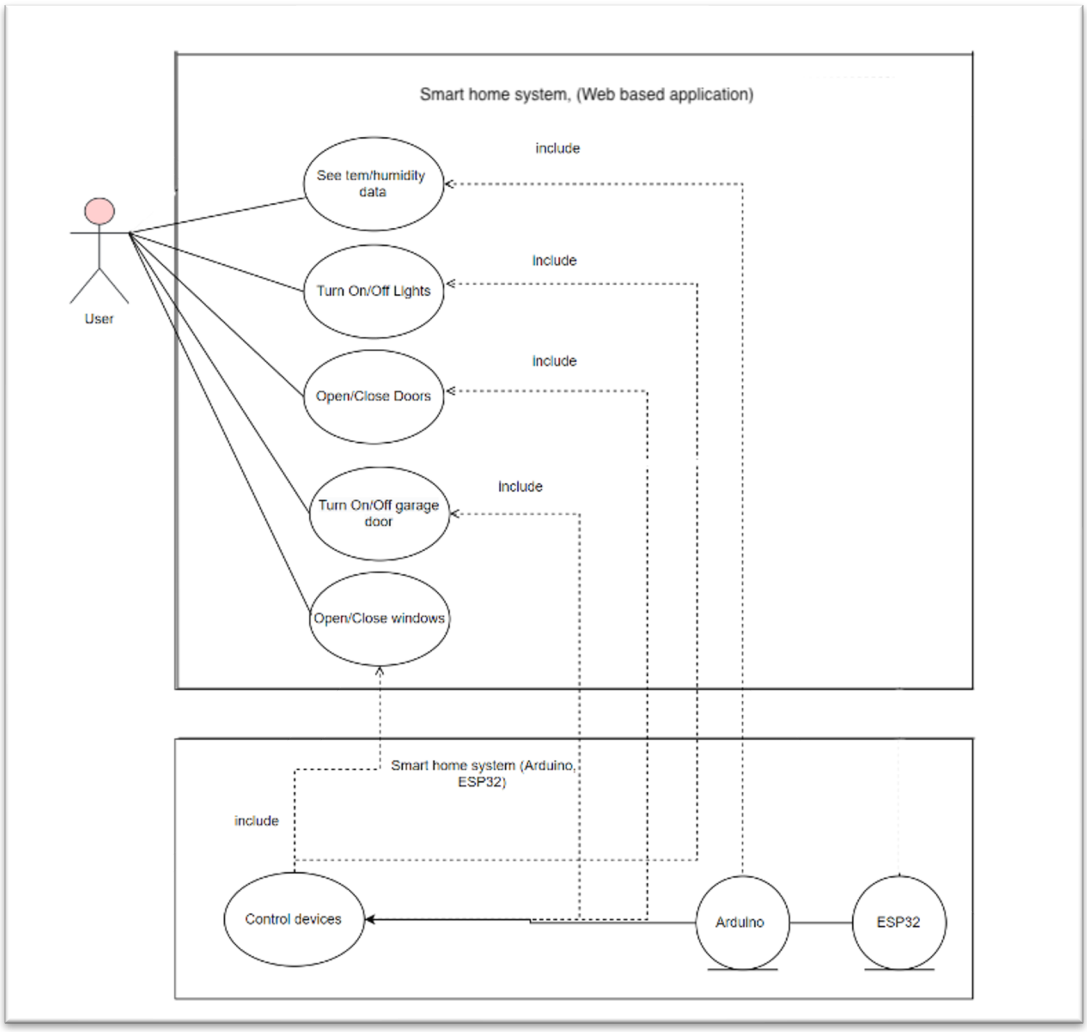


Figure 4-1 Use Case Diagram

4.2.2. Sequence Diagram

Sequence diagram is the interaction between objects in the system [17, 18]. In this chapter there are three sequence diagrams to explain the login steps, register, and light device. The following figures which are 4.2, 4.3, and 4.4 are the sequence diagrams of the smart home system web-based application.

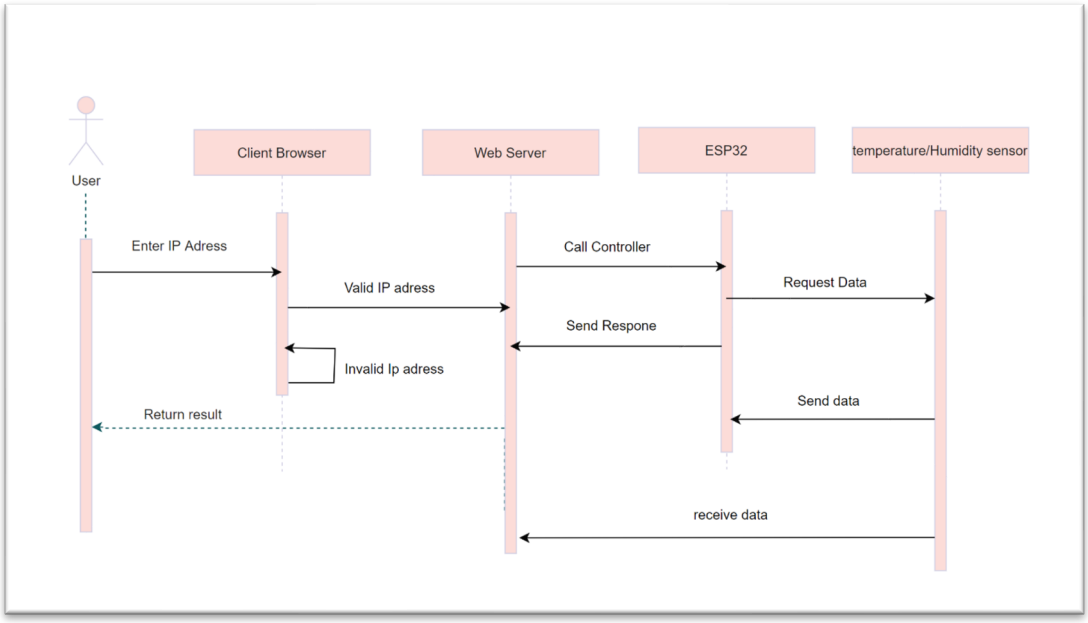


Figure 4-2 User monitor humidity and temperature of the house

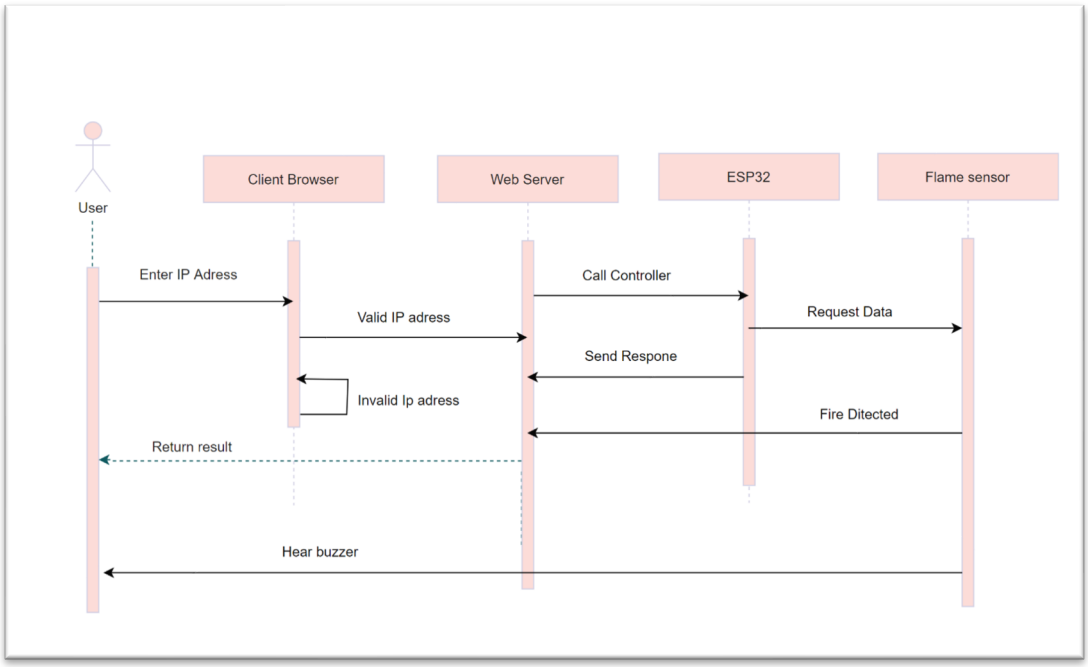


Figure 4-3 sequence diagram for Flame detector and user will be able to hear the buzzer

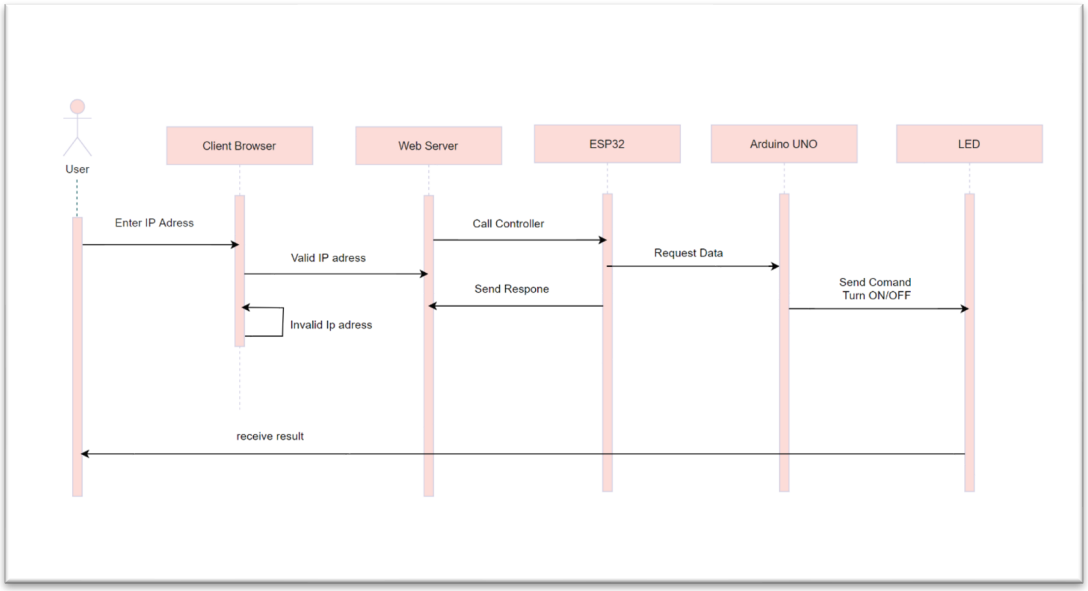


Figure 4-4 On/Off Lights Sequence Diagram

4.2.3. Activity Diagram

Activity diagram represent stepwise activities and actions [17]. Figures: 4.5, , and 4.6 shown below are the activity diagrams for open homeX web server, and turn on the light respectively.

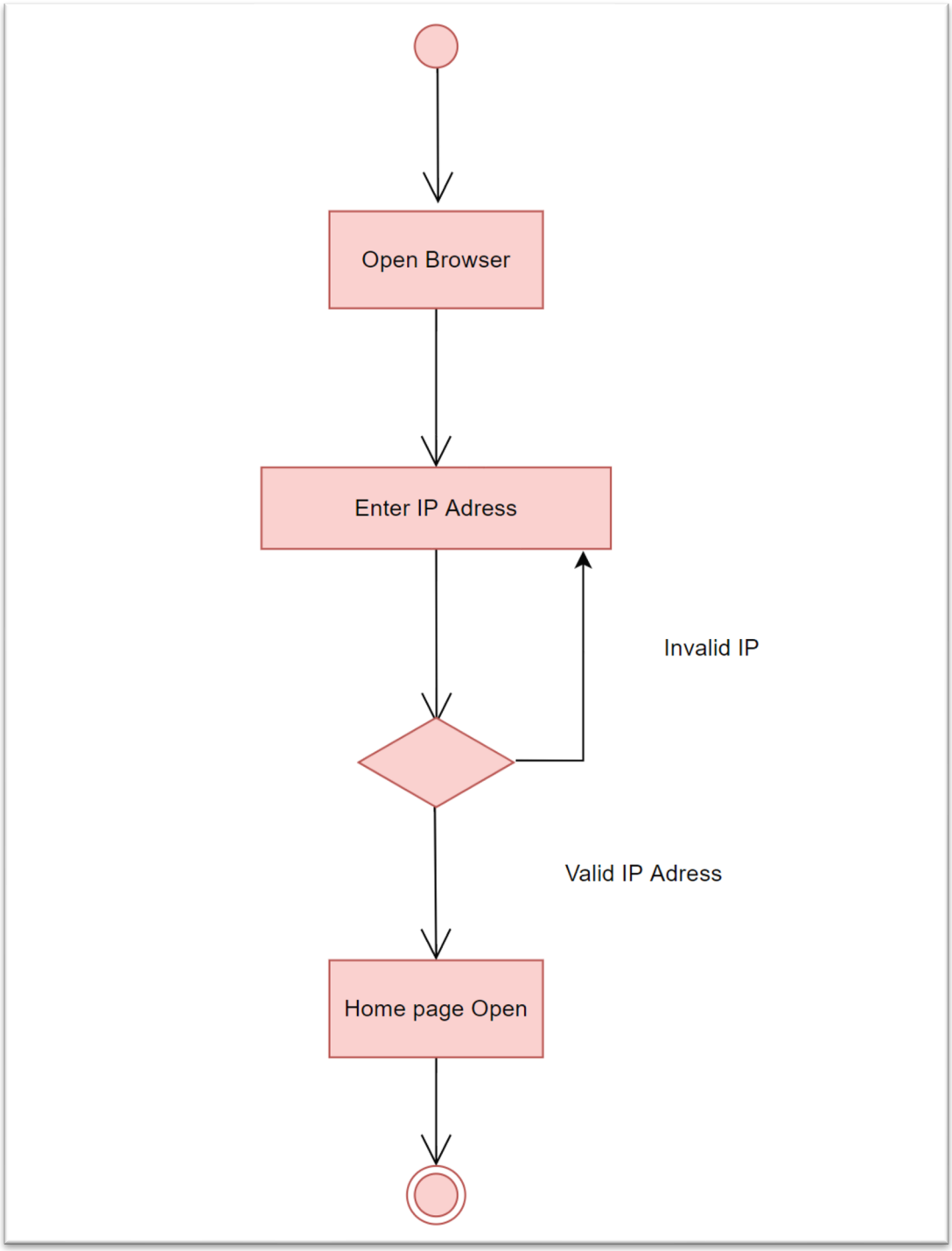


Figure 4-5 User's open homeX web server Activity Diagram

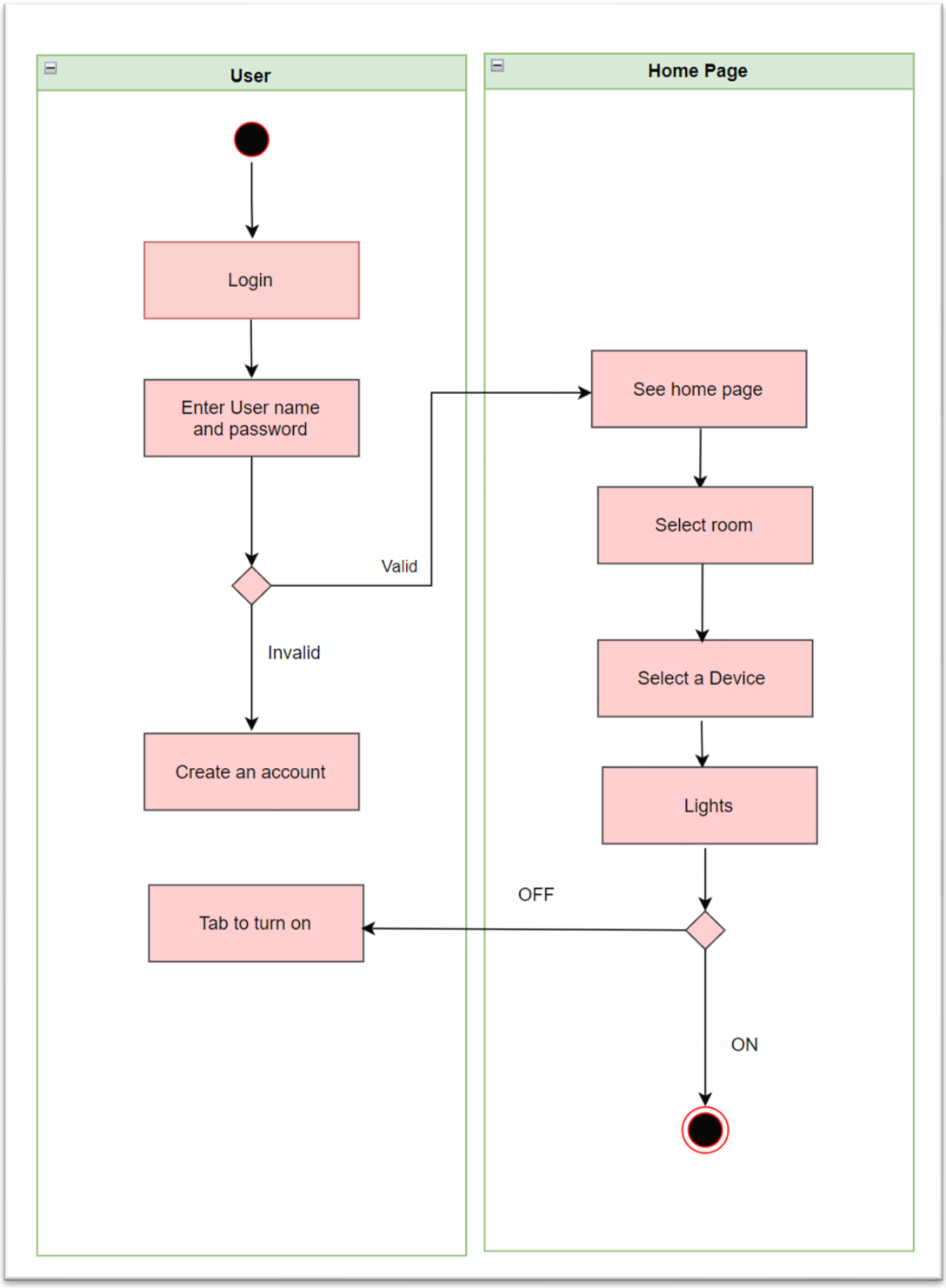


Figure 4-6 On/Off Light Activity Diagram

4.2.4. Class Diagram

Class diagram is used to describe the structure of the whole system [17]. There are three parts in class diagram which are class name, class attributes and class methods. In this class diagram that shown in Figure 4.8, there are seven classes with its own attributes and, methods.

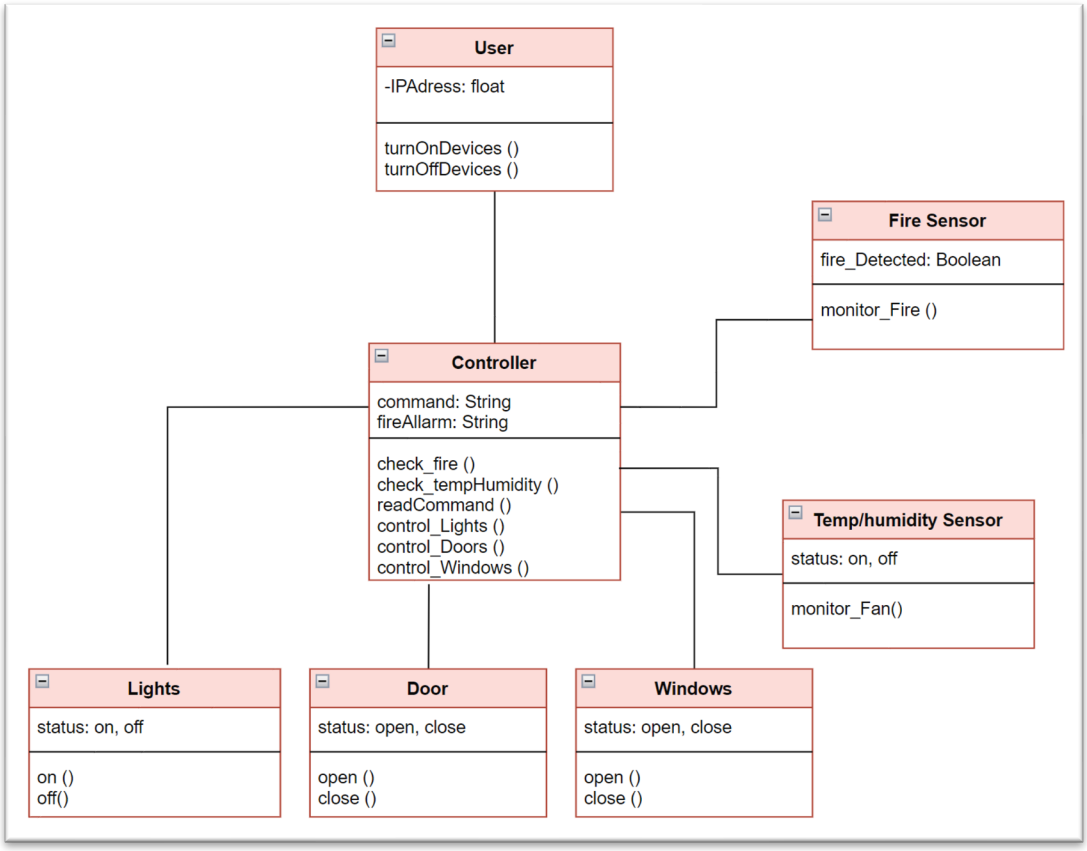


Figure 4-7 The Proposed System's Class Diagram

4.3. Interface Design

Interface design is a process of design that shows the user how the system will work before implementation. The user interface design should be easy to use and easy to learn so, in this web-based application the design that are used in is simple that user

can easily understand. There are two figures to show the user how is the system and how it works.

The first one (figure 4.9) is the first page that the user enter the IP address and the home page will open and it is ready to use. In the first figure that shown below contains the IP address, the name of the web server, the user's name, temperature, humidity and flame statues information. Also, it contains LED1, LED2, LED3 buttons and its status. In (figure 4.10) the remaining button has been shown such as LED4, LED5, Garage door, In door, and window status with its buttons.

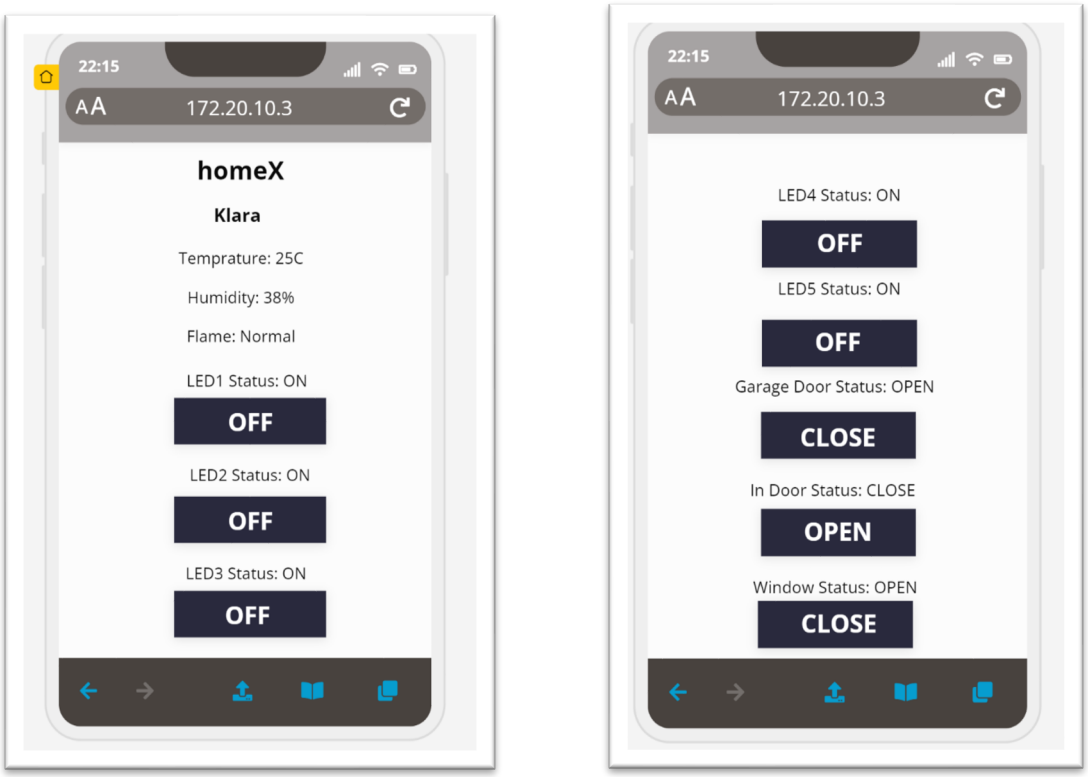


Figure 4-8 First and Second Page

4.4. House Design Plan

The house plan has been created by SketchUp software to design the house that in this project will be used as a model to show the works. It contains six rooms, bathroom and, garage, as shown in (Figure 4.9), (Figure 4.10).

Figure 4-9 The House Model Design

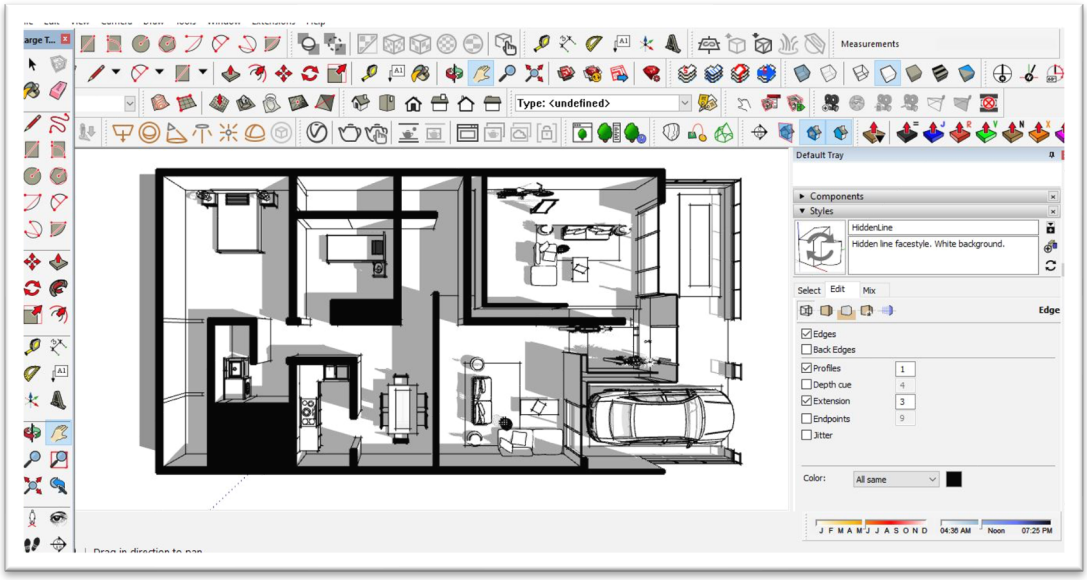


Figure 4-10 Top View of the House using Sketch-up Software

4.5. Chapter Summary

In conclusion, this chapter is mostly focused on the design of the system and how the system works by showing them as diagrams like UML diagrams. Also, the interfaces of the web-based application are shown along with the house plan. In addition, the next chapter will be chapter five that is all about the conclusion of the theoretical part of it.

Chapter 5

IMPLEMENTATION, AND TESTING

5.1. Introduction

After designing and planning all the part of the house. It is the point of doing all the actions and put all of the part together to build the smart house. The house was built with plywood and put it together using bolts and glue. All the microcontrollers were placed under the house to be not visible and can be reaches from all the rooms of the house. A developer board were put over the Arduino UNO board and all the devices including the ESP32 microcontroller were jointed with the developer board. Three 3.7V batteries are giving the Arduino UNO power. A capacitor of 470 □F was used to store electrostatic energy in an electric field, and give this energy to the circuit, when necessary. A capacitor was used because it allows the AC to pass but block the flow of DC to avoid a hazardous breakdown of the circuit. Also, a 1k□ resister was attached to the developer board to limit the current flow, adjust signal levels, bias active elements, and terminate transmission lines among other uses. Moreover, a transistor BC547 is used for quick switching of the fan and pulse-width modulation because we need to control the speed of a fan. Also, the main reason of having this transistor is to not burn the Arduino pin, so when a command was given from the Arduino pin to the resister, and then the resister will give the command to the transistor and finally from the transistor to the fan. The final element of the electrical power part is a diode iN4007. The purpose of having it is to convert alternating current signals (AC) to direct current signals (DC) in electronic products. The fundamental property of a diode is its tendency to conduct electric current in only one direction to not get back to the cycle and crash it. In this system, we have two different sources of electrical power, the Arduino will get power from the three batteries, and the ESP32 will have a different power source which is from a power bank.

The main purpose of using ESP is to make the connection between the web-server and the Arduino uno through using Wi-Fi. Then, the ESP32 has been connected with the Wi-Fi and Arduino throughout TX, RX and GND. In this system, computer will not be used, expect in the beginning. In the beginning, the ESP32 and the laptop should be on the same internet. First, you select the port in this system, the ESP32 port name is COM4. After that, in the Arduino IDE application, there is a serial monitor that will connect the ESP and the server together to give you the IP address that can be used in any browser to open the web server on any smart device.

Inside the house, wires and jumper wires were used to make the connection between the devices inside the house with the Arduino. Wires used for LED, and jumper wires used for the three servo motors and the two sensors. The three servo motors were connected with the window, indoor door, and the garage door. They all connected to the pins over the developer board an get command from the Arduino. Then, the flame sensor was placed in the kitchen. Anytime the sensor senses any smoke it will make a sound and shows in the web-server that a fire alarm is played. Another sensor that placed is DHT22 which is used for getting temperature and humidity. This sensor is connected to the kitchen fan. When the temperature of the room is equal and greater than 20C, it will directly turn on the fan. The sensor is given an if command that will directly do it is job without getting back to the user to turn on the fan or turn of command.

Thar was all the hardware system of the smart house. Now, the most important part is the software system that runs all the LED lights, servo motors, and the sensors. The software system that runs the smart house was two parts. There are two different files which are the Arduino UNO file and the ESP32 file. For the Arduino UNO codes, the Arduino IDE program was used to write the codes. The code file contains all the libraries that were used to control all the devices inside the house. For example, there is a specific library that is used to control the LEDs of each room from the web-server. Also, for all the other devices are the same. The only difference is that for the two sensors, their libraries are different because the sensor has an if command and has to use a logical statement without getting back to the user it will decide what to do. However, the other devices, LED lights and servo motors, will get command from the

user. The other code file is for the ESP32. This code was writing using Arduino Ide, but there are HTML codes inside that are the web-server codes. Inside the code, the Wi-Fi name and password should be written. While the code is running, it will connect the ESP32 with the WIFI and give you back the IP address.

One of the important parts is testing the house system to see what are the feedback of other people and see how the house is functioning. I managed with my university friends and family to do a testing for each function that is in the system such as the door, window, garage door, LED lights. After doing the testing and looking at the feedbacks, I did some improvements in the house. First, the fan was not connected with the DHT22 sensor based on the testers comment I decided to change it to connected with the sensor to turn on whenever the temperature is 25C and above. Moreover, they suggested to hide every wire inside the house to under the house which was a great idea that gave the house more beauty.

5.2. Inside The Arduino Code we have:

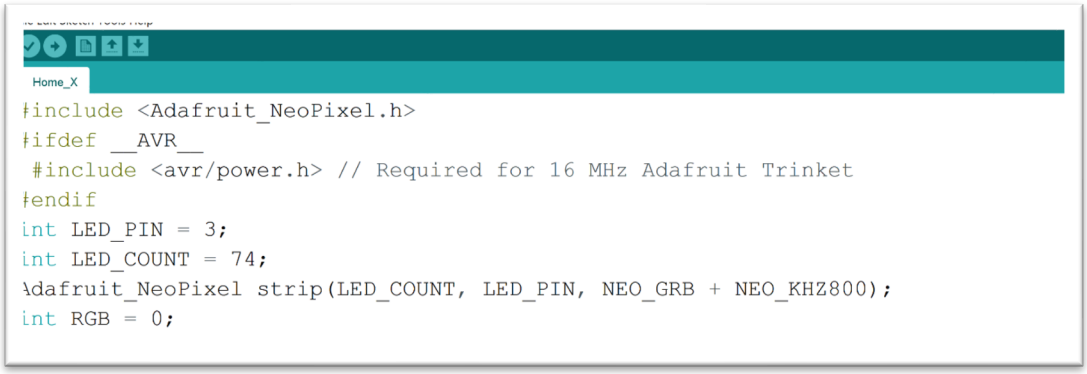


Figure 5-1 Code Segment 1

`#include <Adafruit_NeoPixel.h>` this is a library for controlling the LEDs
`int LED_COUNT = 74;` This means that we have 74 LEDs
`int LED_PIN = 3;` All the LEDs will get the command from Pin 3 on the Arduino Board
`Adafruit_NeoPixel strip(LED_COUNT, LED_PIN, NEO_GRB + NEO_KHZ800);` here we have created an object named strip which has LED count, LED Pin from this parameter we will give it back to its library to work properly.

`int RGB = 0;` to change the LED status from red to green to blue.

Figure 5-2 Code Segment 2

Code above is for servo motor libraries which responsible for opening and closing doors, windows and the garage door. We have created three objects for them which are garage_servo, InDoor_Servo, and Window_servo.

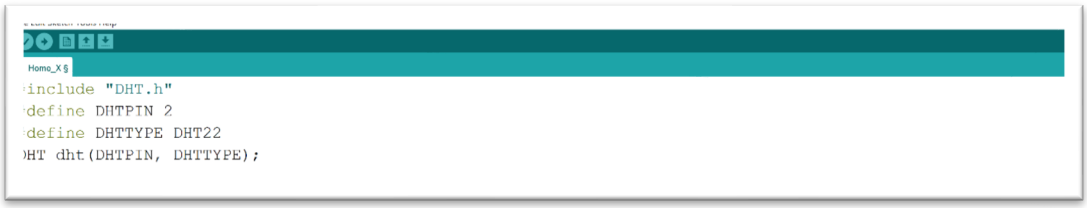


Figure 5-3 Code Segment 3

`#include "DHT.h"` Another library for DHT which is a sensor for humidity and temperature
`#define DHTTYPE DHT22` this is the model of the sensor that we have declared
`DHT dht(DHTPIN, DHTTYPE);` And we created an object that contains two targets which are the PIN and the model.

```
Home_X §

const byte Fan_PIN = A0;
const byte Flame_PIN = 4;
byte Flame_State = 0;
const byte Buzzer_PIN = A1;

String command = "";
```

Figure 5-4 Code Segment 4

`const byte Fan_PIN = A0;` This means the fan is in Analog 0.
`const byte Flame_PIN = 4;` the flame sensor is connected with digital number four
`const byte Buzzer_PIN = A1;` The buzzer pin is on analog A1
`String command = "";` This is for checking and getting the command between the ESP32 and the Arduino. Whenever a command send from the ESP32 will get into this code and will apply the Arduino conditions. We have the same code inside the ESP32 too.

5.3. Inside the ESP32 Code

```
ile Edit Sketch Tools Help

homeX

#include <WiFi.h>
#include <WebServer.h>

const char* ssid = "clara";  // Enter SSID here
const char* password = "11111111";  //Enter Password here
WebServer server(80);

bool LED1_Status = LOW;
bool LED2_Status = LOW;
bool LED3_Status = LOW;
bool LED4_Status = LOW;
bool LED5_Status = LOW;
bool GarageDoor_Status = LOW;
bool InDoor_Status = LOW;
bool Window_Status = LOW;

int Temprature = 0;
int Humidity = 0;
bool Flame = false;

String command = "";
```

Figure 5-5 Code Segment 5

First, we should declare two libraries which are **`#include <WiFi.h>`** to connect to wifi and **`#include <WebServer.h>`** that is dedicated to a web server

`const char* ssid = "clara";` This is for writing our WiFi name

`const char* password = "11111111";` This is for entering the WiFi password

`WebServer server (80);` We selected port 80 for our web server

```
bool LED1_Status = LOW;
bool LED2_Status = LOW;
bool LED3_Status = LOW;
bool LED4_Status = LOW;
bool LED5_Status = LOW;
bool GarageDoor_Status = LOW;
bool InDoor_Status = LOW;
bool Window_Status = LOW;
```

```
void setup() {
  Serial.begin(115200);
  while (WiFi.status() != WL_CONNECTED) {
    delay(3000);
    Serial.print(".");
    WiFi.begin(ssid, password);
  }
  Serial.println("");
  Serial.println("WiFi connected..!");
  Serial.print("Got IP: ");  Serial.println(WiFi.localIP());
```

Figure 5-6 Code Segment 6

The codes above are for the LED status we set is as bool because for LEDs we only have two status which are high and low. In this case all LEDs are off which means in LOW status. Also, for the windows and doors there are only two status which are open and close which means LOW and HIGH since the doors and the windows are close the status are equal to LOW.

```
int Temprature = 0;
int Humidity = 0;
bool Flame = false;
```

These are for temperature, humidity that starts from zero and the bool flame is equal to false means there is no fire inside the house and also has two statuses only which are true and false.

Same as the Arduino we have **`String command = "";`** to get the command and data from the Arduino and apply ESP32 conditions.

Figure 5-7 Code Segment 7

Inside the void setup we have serial begin to get the data from the Arduino and they should have the same number which is 115200. We have a while loop for whenever the wifi is not connected wait three seconds and put dots to make sure its not connected yet and then search for the wifi again. This loop will work until its connected to the Wi-Fi.

```
Serial.println("");
Serial.println("WiFi connected..!");
Serial.print("Got IP: ");  Serial.println(WiFi.localIP());
```

After its connected to the Wi-Fi you will get the IP address.

Figure 5-8 Code Segment 8

```
void handle_OnConnect() {
  server.send(200, "text/html", SendHTML());
}
```

The 200 is just a number to make sure that the web server is open, and it is has Send HTML function.

Figure 5-9 Code Segment 9

void handle_NotFound(){server.send(404, "text/plain", "Not found");}

This will give us error 404 and tells us not found. We have declared all the functions inside Void setup and all of them has ID.

5.4. web application we have HTML codes

```
String SendHTML(){
  String ptr = "<!DOCTYPE html> <html>\n";
```

In this send HTML we have a string that has a variable ptr that contains information about the HTML for example every five seconds refresh the web server also, we have details and titles of the web based application.

Chapter 6

CONCLUSION AND RECOMMANDATION

6.1. Introduction

A smart home system (SHS) with a web-server is a system to lets the user remotely monitor their home. Today, with the improvement of technology, most of the devices and environment around human become smart which mean that they can understand and take an action with a command. Sometimes the command is given by the person or the device can understand by itself some particular things. These smart technologies are all made human life easier and safer because fire occurs daily on houses due to not turning off a device or a light. Also, there are sometimes that someone leaves the house then realize that he/she might forget to lock the door. The project aims to make life easier and safer in the house by providing an automatic window that can be opened and closed from your smartphone. Also, another important thing is that all the lights are smart and connected to Wi-Fi, so that can be controlled all from the web-server. In addition, for the safety of the house, flame sensors will take action whenever they detect fire.

What is required for making this project is an Arduino UNO microcontroller, ESP32, sensor, a model house, and a smartphone that can run the web-server. First of all, the Arduino and the ESP32 should be programmed and connected to the Wi-Fi, and connected with the devices that you need to monitor it all by using the Arduino IDE to write its codes. Second, create a web-server and put it inside the ESP32 code to work as a web-server. Third, a model house is required to implement and show the users how the smart home works. Finally, all the systems should be connected and work with WiFi to let the user get the benefit of using or controlling their home remotely.

6.2. Achievement

After implementing all the plans and following all the procedure that were planned, the smart house is ready. After connecting it to the Wi-Fi and connecting it with the phone throughout the IP address, the web-server will be open. The web-server has all the control bottoms that is require to control the smart house. The user can easily control the doors and the window from just touching the phone screen. Then, the phone will send the command to the ESP32 through the Wi-Fi, the ESP32 will send it to the Arduino UNO, and the Arduino will apply the conditions and send to the controlled devices.

6.3. Suggestion

Using a smart home in Kurdistan is rare since it is new here, and many people still cannot believe in technology to do the work for them, another reason is that smart homes cost too much. Having a project like this is necessary to show people how their life will be changed and easy after changing from a regular home to a smart home. Moreover, people who are living in Kurdistan are always stressed and living in worry so, a smart home system can decrease all this stress. Finally, In Kurdistan, many issues happen in terms of electricity problems while you have a smart home you can make sure that every device is turned off and see your home from your phone to make sure that everything is fine.

REFERENCES

Hayes, A. (2021). Smart home. Investopedia. Retrieved December 5, 2021, from https://www.investopedia.com/terms/s/smart-home.asp.

Zhilibayev, S., Ziyashev, A., Zhelambayeva, A., Yessembayev, A., Yazici, A., & Ever, E. (2020). Low cost smart house implementation with sensory information analysis and face recognition. 2020 12th International Conference on Knowledge and Smart Technology (KST). https://doi.org/10.1109/kst48564.2020.9059401

Gota, D.-I., Puscasiu, A., Fanca, A., Miclea, L., & Valean, H. (2020). Smart Home Automation System using Arduino microcontrollers. 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR). https://doi.org/10.1109/aqtr49680.2020.9129989.

Home - Garden City Home Garden City Villas. Garden City Home. (2021). Retrieved December 5, 2021, from https://gardencitysul.com/.

Arduino Uno REV3. Arduino Official Store. (n.d.). Retrieved December 5, 2021, from https://store.arduino.cc/products/arduino-uno-rev3/.

Singh, U., & Ansari, M. A. (2019). Smart Home Automation System using internet of things. 2019 2nd International Conference on Power Energy, Environment and Intelligent Control (PEEIC). https://doi.org/10.1109/peeic47157.2019.8976842

Sarhan, Q. I. (2020). Arduino based Smart Home Warning System. 2020 IEEE 6th International Conference on Control Science and Systems Engineering (ICCSSE). https://doi.org/10.1109/iccsse50399.2020.9171939.

Gota, D.-I., Puscasiu, A., Fanca, A., Miclea, L., & Valean, H. (2020). Smart Home Automation System using Arduino microcontrollers. 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR). https://doi.org/10.1109/aqtr49680.2020.9129989.

Gota, D.-I., Puscasiu, A., Fanca, A., Miclea, L., & Valean, H. (2020). Smart Home Automation System using Arduino microcontrollers. 2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR). https://doi.org/10.1109/aqtr49680.2020.9129989.

Borkar, P., Dhakate , A., & Amrute, A. (n.d.). Smart home automation using Arduino Uno REV2 microcontroller. Retrieved December 31, 2021, from https://easychair.org/publications/preprint_download/FKk2.

Stolojescu-Crisan, C., Crisan, C., & Butunoi, B.-P. (2021, May 30). An IOT-based Smart Home Automation System. MDPI. Retrieved December 23, 2021, from https://www.mdpi.com/1424-8220/21/11/3784#cite.

Difference between Arduino and Raspberry Pi. GeeksforGeeks. (2020, January 3). Retrieved December 31, 2021, from https://www.geeksforgeeks.org/difference-between- arduino-and-raspberry-pi/.

What are software development methodologies? Alliance Software. (2021, October 26). Retrieved January 20, 2022, from https://www.alliancesoftware.com.au/introduction-software-development-methodologies/.

Software Development Life Cycle (SDLC). SDLC - Overview. (n.d.). Retrieved January 20, 2022, from https://www.tutorialspoint.com/sdlc/sdlc_overview.html

10 reasons to use agile software development. QualityLogic. (2019, July 18). Retrieved January 15, 2022, from https://www.qualitylogic.com/2019/07/18/10-reasons-to-use-agile-software-development/.

Lamelas, A., & Lamelas, A. (2021, March 31). Top 5 main agile methodologies: Advantages and disadvantages. Xpand IT. Retrieved January 20, 2022, from https://www.xpand-it.com/blog/top-5-agile-methodologies/.

Rumpe, B. (2016). Modeling with UML (pp. 1-281). Cham: Springer.

Maylawati, D. S., Darmalaksana, W., & Ramdhani, M. A. (2018). Systematic design of expert system using unified modelling language. In IOP Conference Series: Materials Science and Engineering (Vol. 288, No. 1, p. 012047). IOP Publishing.

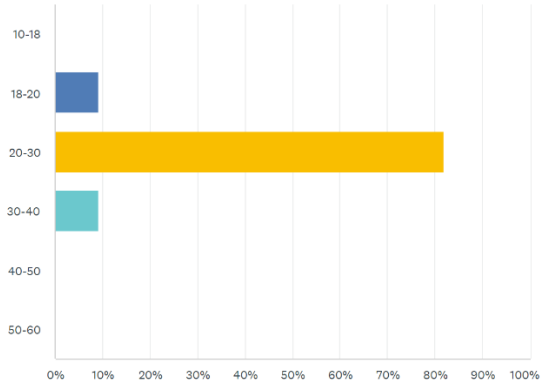
Appendix A
Interview

Q1

Customize Save as

Age

Answered: 22 Skipped: 0



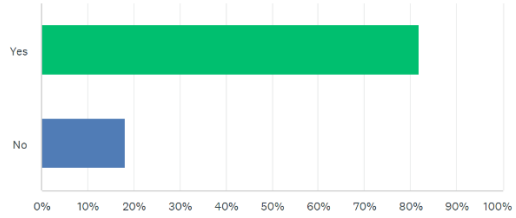
ANSWER CHOICES	RESPONSES	
10-18	0.00%	0
18-20	9.09%	2
20-30	81.82%	18
30-40	9.09%	2
40-50	0.00%	0
50-60	0.00%	0
TOTAL		22

Q2

Customize Save as

Have you ever heard about a smart house?

Answered: 22 Skipped: 0



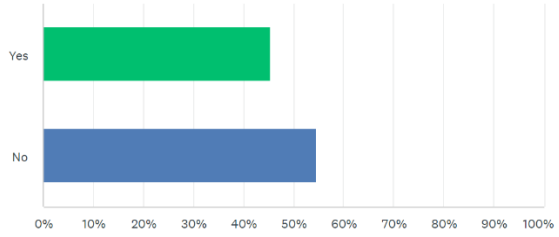
ANSWER CHOICES	RESPONSES	
Yes	81.82%	18
No	18.18%	4
TOTAL		22

Q3

Customize Save as

Have you ever used smart technology in your house?

Answered: 22 Skipped: 0



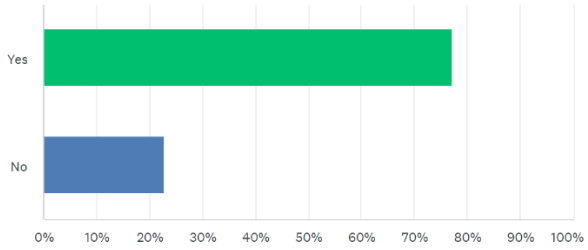
ANSWER CHOICES	RESPONSES	
Yes	45.45%	10
No	54.55%	12
TOTAL		22

Q4

Customize Save as

Do you think that a smart home will cost too much?

Answered: 22 Skipped: 0



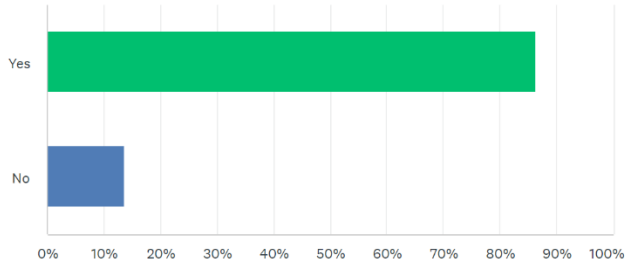
ANSWER CHOICES	RESPONSES	
Yes	77.27%	17
No	22.73%	5
TOTAL		22

Q5

Customize Save as

Do you want to lay on your bed and turn off the light?

Answered: 22 Skipped: 0



ANSWER CHOICES	RESPONSES	
Yes	86.36%	19
No	13.64%	3
TOTAL		22

Appendix B Software design document (SDD)

1. Introduction

A collection of documents as well as resources called design documentation covers every element of your product design. Also, all diagrams that are related to design part such as system architecture design, user interface design and use case diagram has been mentioned in this documentation.

Purpose

This SDD describe Users, product features, project deadlines, all crucial implementation information, and design choices that have been approved by stakeholders should all be included in the documentation.

Scope

The software product is to completely describe a system's design so that software development can proceed with an understanding of what needs to be built and how it is being built.

Definitions, Acronyms and Abbreviation

SDD: Software Design Documentation

MVC: Model View Controller

1.1 References

Cadier, A. (2021, July). Unidentified Users of Design Documentation. In *International Conference on Human-Computer Interaction* (pp. 11-16). Springer, Cham.

Permatasari, D. I., Fahrul Hardiansyah, F., Ainun Wakhidah, M., & Bagus Afridian Rasyid, M. (2021, September). UX Design Documentation Application Using The Five Planes Method. In *6th International Conference on Sustainable Information Engineering and Technology 2021* (pp. 29-32).

1.2 Overview

This document contains several diagrams and design such as interface design, architecture design, sequence diagram, and class diagram.

2. System Architectural Design

Architecture Style and Rationale

The chosen architecture would be MVC which means view model controller. Every time data changes, it updates the view and manages the flow of data into the model object.

Architecture Model

The MVC has been selected for this project the (figure 2.1) below has been explained the first part is the view, the second part is the controller and the last part is the module. The view model contains the user interface and the controller is the Arduino microcontroller, finally the module contains the devices that can be controlled by the controller.

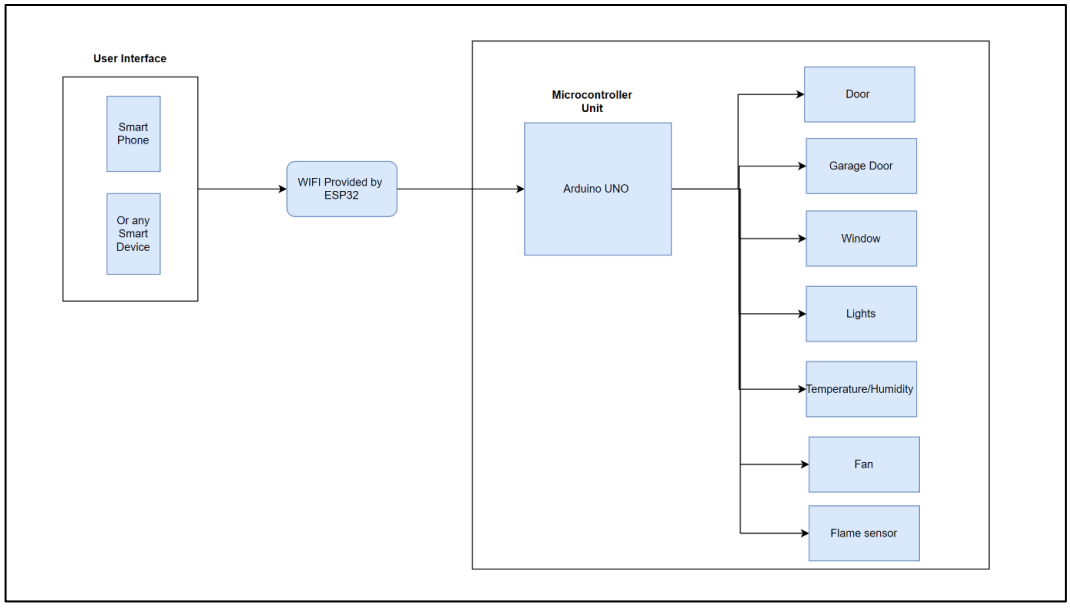


Figure 2.1: System Architecture of <Smart Home System>

2.1 Use Case Diagram

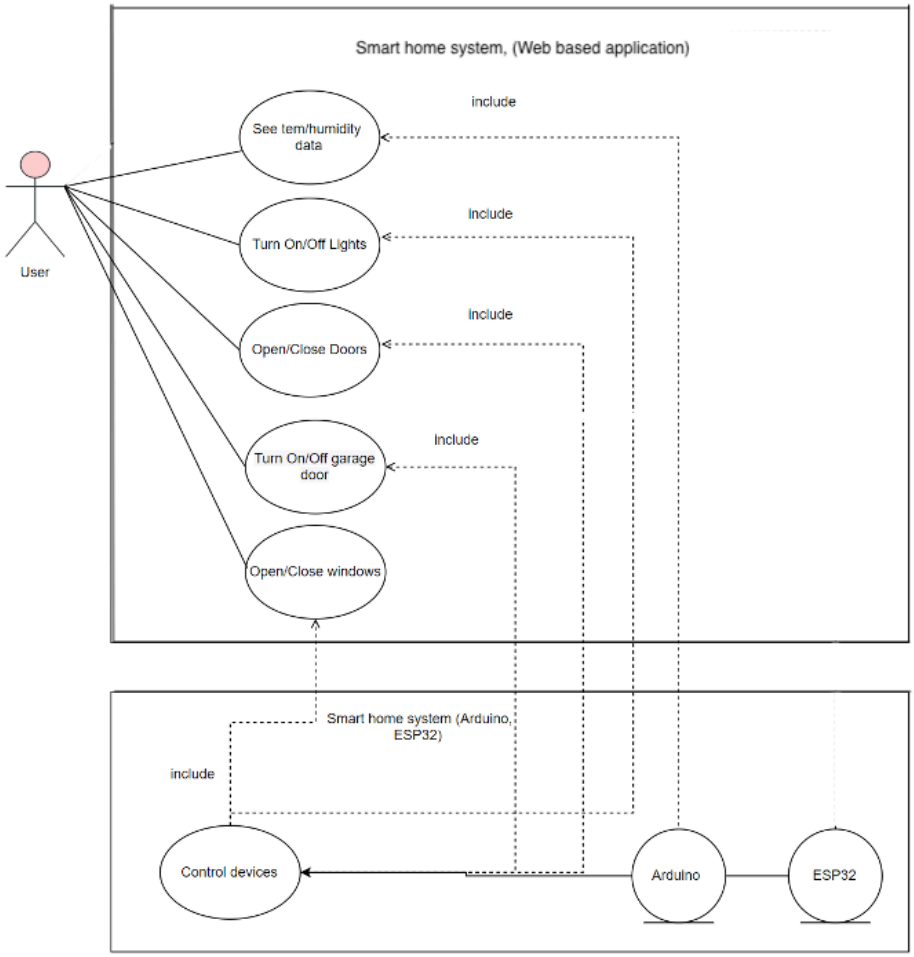


Figure 2.2: Use Case Diagram of <The smart home system>

3. Detailed Description of Components

A component diagram, often called a UML component diagram, shows how the physical parts of a system are wired up and organized. The (figure 3.1) shown below is explain the three model with its component which are view, model, and controller. The view model contains one package which is main view for user, and the second one is the model in MVC architecture contains one package which is a web server. Finally, a controller that contains two package for the Arduino and the ESP32.

3.1 Complete Package Diagram

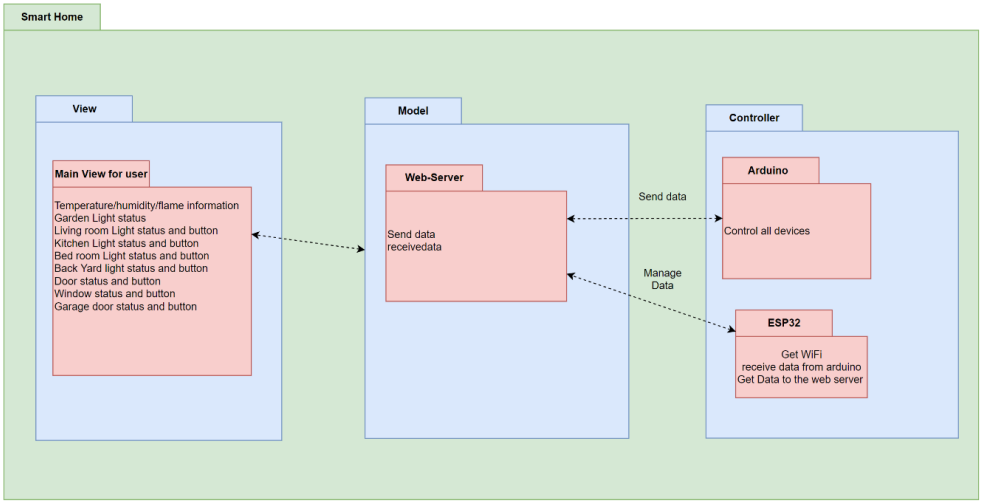


Figure 3.1: Complete package diagram of <Smart Home System>

3.2 Component Model

A component model is a description of the requirements that components must meet as well as the procedures and techniques used in their assembly.

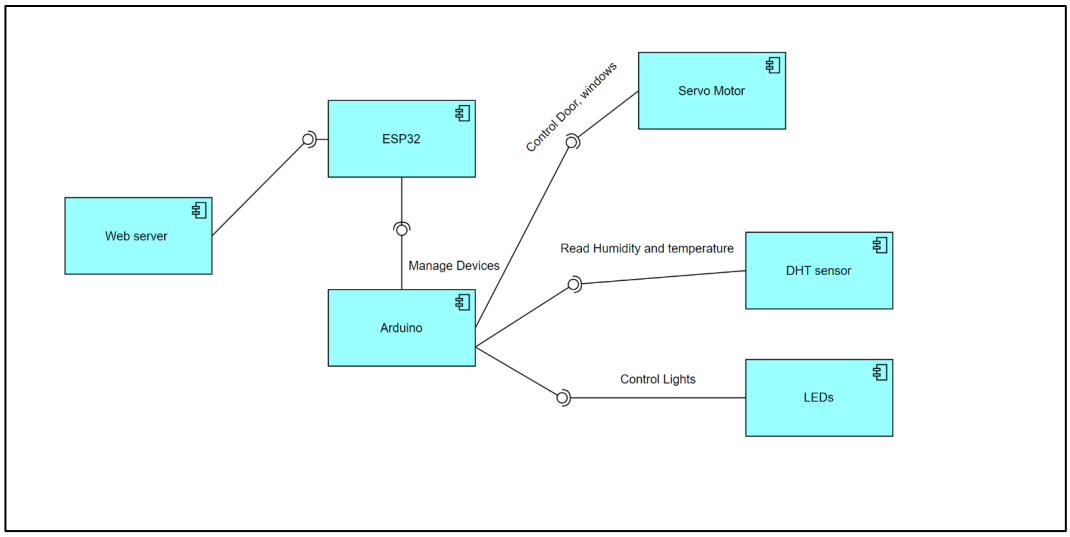


Figure 3.2: Component Diagram of <Smart Home System>

3.2.1.1 Class Diagram

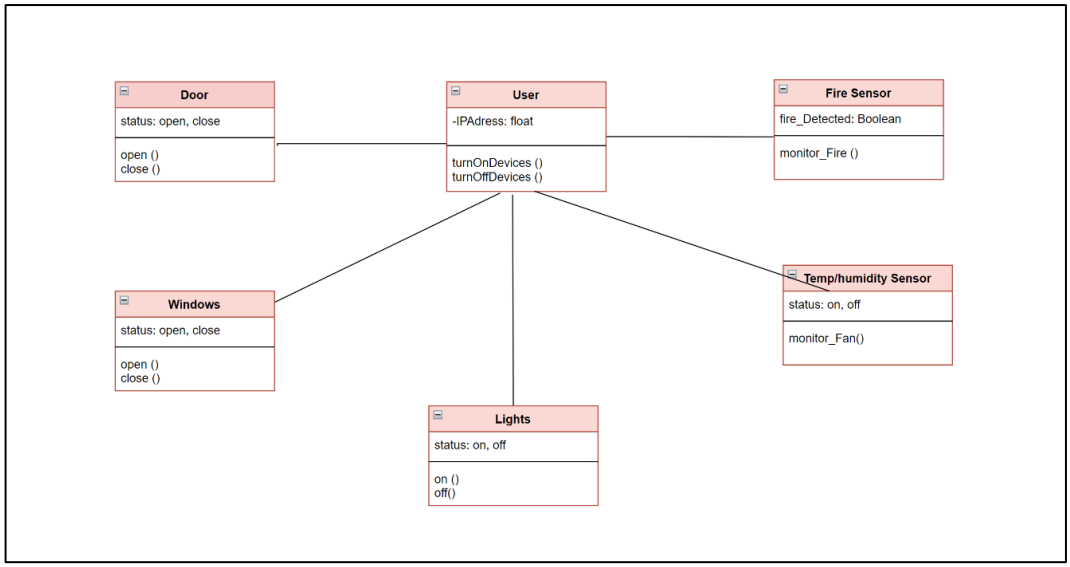


Figure 3.2: Class diagram for <Order Entry Package>

3.2.1.2 Sequence Diagrams

a) SD001: Sequence diagram for receive data to the user from the DHT sensor.

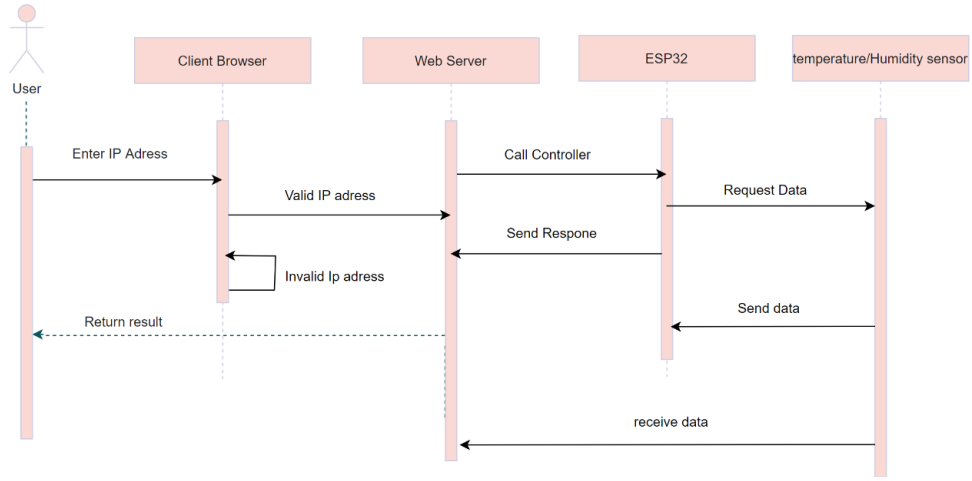


Figure 3.3: Sequence Diagram of <C receive data to the user from the DHT sensor >

b) SD002: Sequence diagram for user to hear the buzzer when flame sensor detect fire

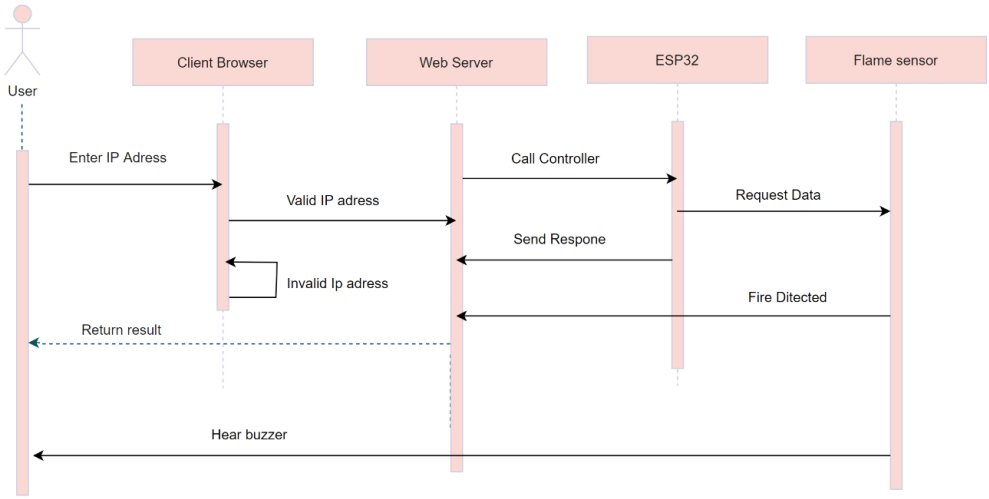


Figure 3.4: Sequence Diagram of <Fire detector>

4. User Interface Design

4.1 Overview of User Interface

Interface design is a process of design that shows the user how the system will work before implementation. The user interface design should be easy to use and easy to learn so, in this web-server the design that are used in is simple that user can easily understand. There are two figures (figure 4.1 and 4.2) to show the web server interface and one (figure 4.3) to show the house interface in section 4.2.

4.2 Screen Images

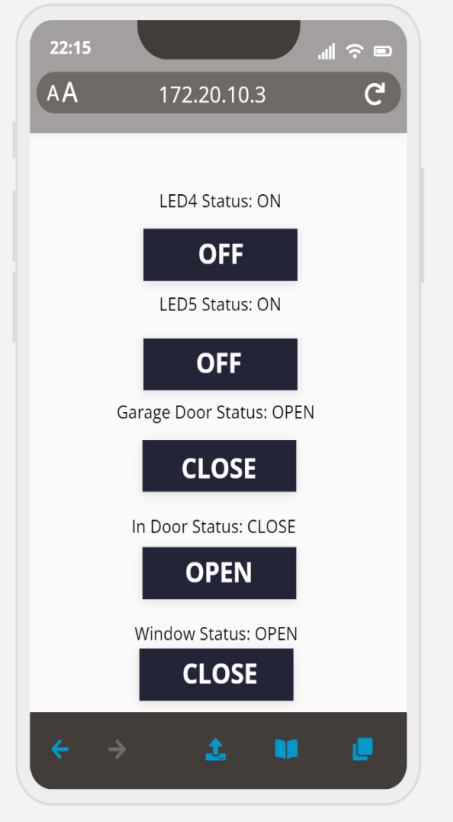


Figure 4.1 User Interface design on the web server

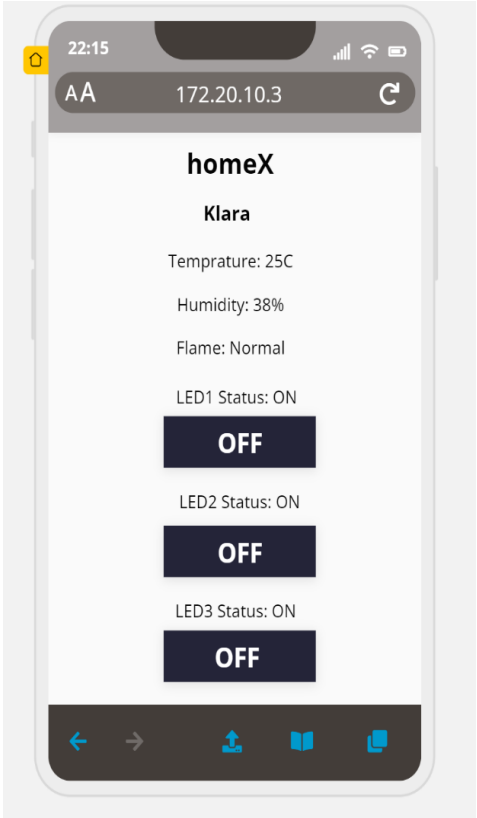


Figure 4. User Interface Design on Web server

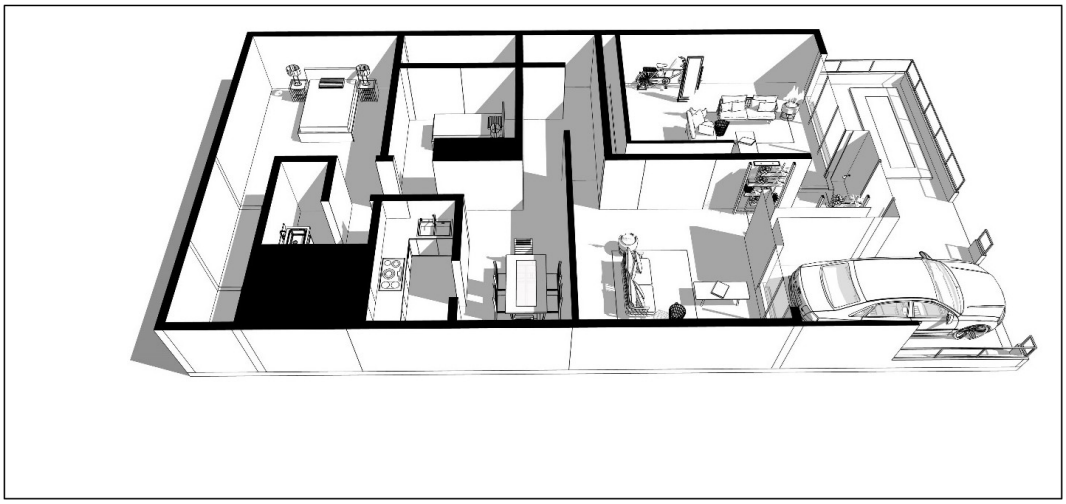


Figure 4.3 House interface Design

5. Requirements Matrix

	P001	P002	P003
UC001	X		
UC002	X		
UC002	X		
UC004	X		
UC005	X		

Appendix C Software Requirement Specification (SRS)

1. Introduction

An SRS paves the way for agreement between the client and the provider over the functionality of the software product. SRS offers a reference for software/final product validation. High-quality products and software require high-quality SRS. The cost of development is decreased by a high-quality SRS.

2. Purpose

The purpose of having SRS is to reduces the amount of time and effort needed by developers to complete specified tasks as well as the cost of development. A good SRS outlines the interactions that an application will have with the hardware of the system, other applications, and users in a wide range of real-world scenarios.

3. Scope

Smart Homes, also known as automated homes. Recently, the use of smart technology has developed so that almost any electrical component within the house can be included in the system. Turning a regular home to a smart home by designing a model house and implement what is necessary to have in a smart house by using Arduino UNO, ESP32, and connected with a web server that can be controlled from a smart phone. Smart home can monitor the activities of the occupant of a home, independently operate devices in set predefined patterns or independently.

4. Definitions, Acronyms and Abbreviation

SRS: Software requirement specification

5. References

Ali, S. W., Ahmed, Q. A., & Shafi, I. (2018, February). Process to enhance the quality of software requirement specification document. In *2018 International Conference on Engineering and Emerging Technologies (ICEET)* (pp. 1-7). IEEE.

Sabriye, A. O. J. A., & Zainon, W. M. N. W. (2017, May). A framework for detecting ambiguity in software requirement specification. In *2017 8th International Conference on Information Technology (ICIT)* (pp. 209-213). IEEE.

6. Overview

This documentation will describe what the software will be able to do and how it will work. Many diagrams will be put inside this SRS documentation.

7. Overall Description

This section will provide diagrams such as use case diagram, block diagram, system interface, user interface, hardware interface, and system function.

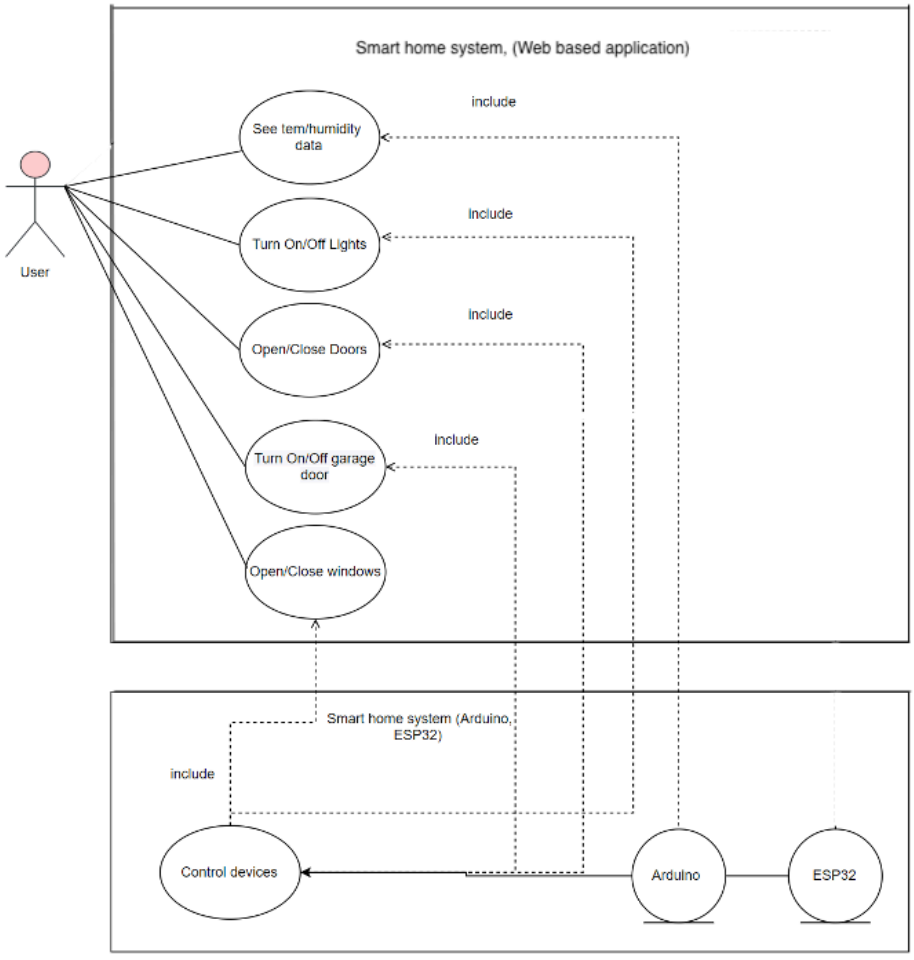


Figure 2.1: Use Case Diagram of <Smart Home System>

8. Product Perspective

In this system two microcontroller has been used one of them is Arduino UNO and the other one is ESP32. Both microcontrollers need power supply, the Arduino will get the power from three 3.7-volt batteries and the ESP32 will get the power from the USB either from the laptop power or a power bank. This system contains 74 LEDs, a door, a garage door, windows, fan and two sensors. The (figure 2.2) which is a block diagram will explain the system and the sub system.

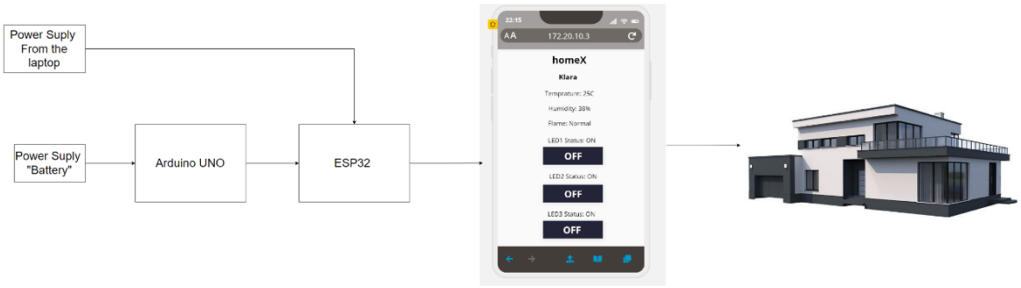


Figure 2.2 Block Diagram <Smart Home system>

9. System Interfaces

Interface design is a process of design that shows the user how the system will work before implementation. The user interface design should be easy to use and easy to learn so, in this web-based application the design that are used in is simple that user can easily understand. There are two figures to show the user how is the system and how it works as shown in (Figure 2.3) and (Figure 2.4)

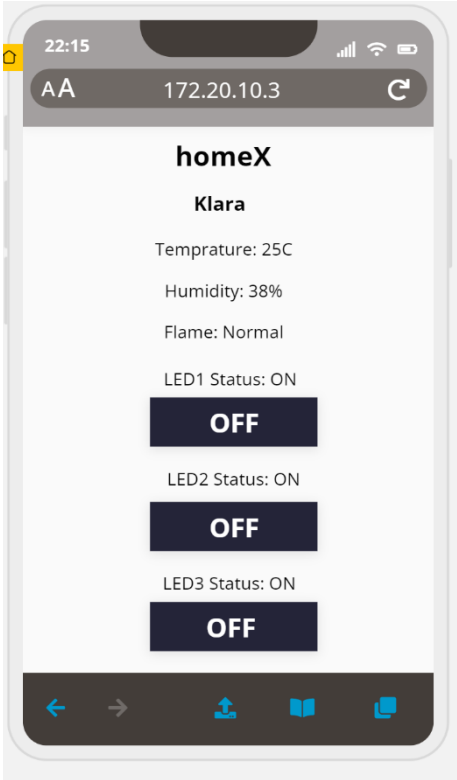


Figure 2.3 Interface Design First Page

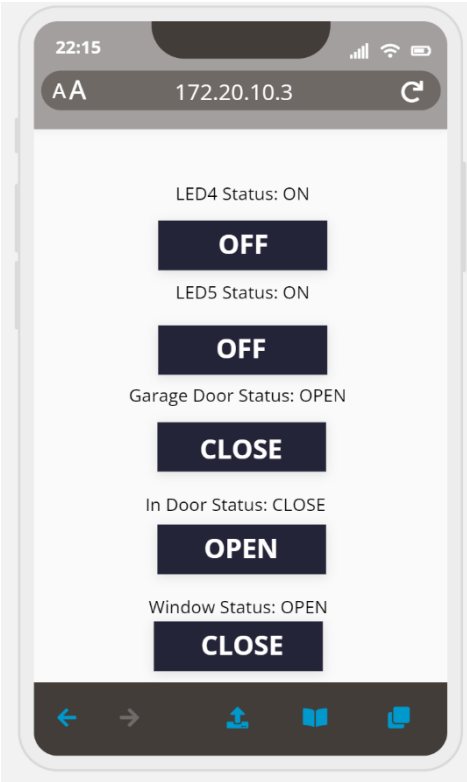


Figure 2.4 Interface Design Second Page

10. User Interfaces

This system will interact with the user from a smart device screen. The user will get the IP address, after getting the IP address the user can copy and paste the IP in any smart device that has browser. The user should write the right IP address to open the homeX page and they will be able to control the devices, any wrong IP address will not be allowed and it get you to the error page.

11. Hardware Interfaces

Arduino UNO and ESP32, an open-source microcontroller with input/output pins, is needed. The Arduino breadboard is also used which is a plastic board that electronics can be connected to it and used along with the Arduino UNO. Also, a flame detector sensor is used in hot kitchen that will alert the system if a fire is happening. Furthermore, a servomotor has been used in this project, which is a small motor connected to the Arduino and can produce velocity, is used for opening/closing windows, door, and garage door. Finally, each room has its own LEDs that can be control with the Arduino.

12. Software Interfaces

Several software and tools are required to build a smart house, including Arduino IDE, an application for writing Arduino code, and Fritzing, which is an application for designing the Arduino with its wires. Finally, there is UiZard, which is used for prototyping. Coding in Arduino IDE are written in C++ and for the web server HTML code has been used. The libraries that have been used for the Arduino are Adafruit_NeoPixel.h, avr/power.h, Servo.h, DHT.h. The libraries that have been used for ESP32 are WiFi.h, WebServer.h

13. Communication Interfaces

1. Power
2. Local Wi-Fi

14. Product Functions

Based on the use case diagram that shown in (figure 2.1) the system has three actors including the ESP32 and the Arduino. The user will be able to open the browser and enter the IP address, also, he/she be able to control the devices such as LEDs, window, and door. Moreover, the user will be able to see the temperature and humidity of the house. The system has Arduino as an actor because the Arduino is responsible for controlling the devices. ESP32 is also an actor because it provides the Wi-Fi for the system and connect the web server with the Arduino.

15. Specific Requirements

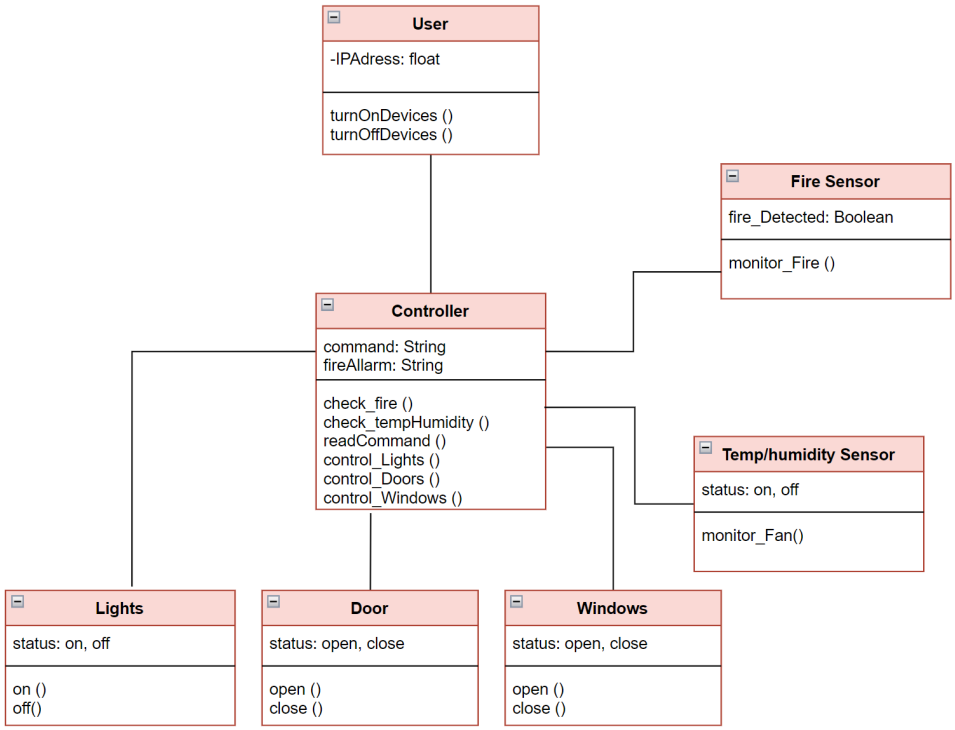


Figure 3.1: Domain Model of <Smart Home System>

16. System Features

Module <Smart Home System>

The diagram below is the use case for the user only which is a part of the whole usecase that has been mentioned in (figure 2.1). In this diagram (figure 3.2) the user will be able to enter the valid IP address and control the devices such as open/close the door, window, garage door, and turn on/off the LEDs, also, the user will be able to see the data of the temperature and humidity of the room.

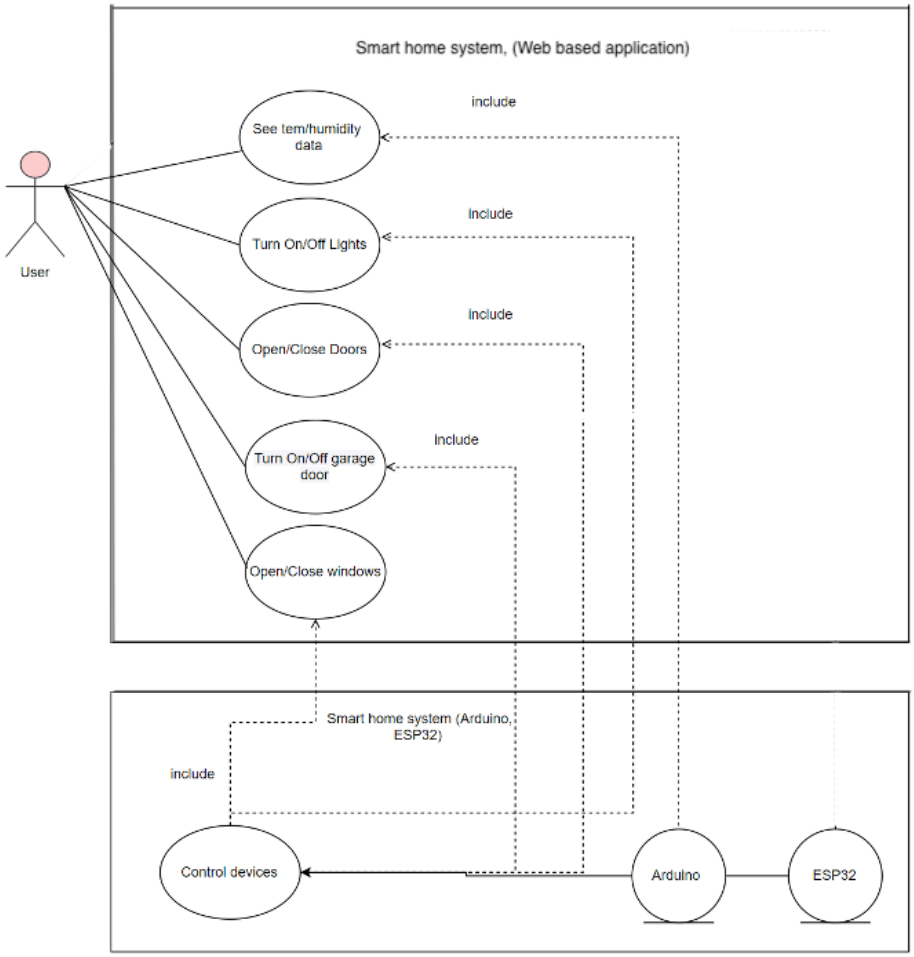


Figure 3.2: <User functionality >

1. **UC001: Use Case <Open/Close door>**

#include <Servo.h>

Servo InDoor_Servo;

InDoor_Servo.write(100)

2. **UC002: Use Case <Open/Close Garage door>**

#include <Servo.h>

Servo Garage_Servo;

Garage_Servo.write(90);

3.1.1.3 UC003: Use Case <Open/Close Window>

#include <Servo.h>

Servo Window_Servo;

Window_Servo.write(100);

3.1.1.4 UC004: Use Case <Humidity and Temperature>

```
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT22
DHT dht(DHTPIN, DHTTYPE);
byte MAX_Temp = 25;
```

3.1.1.5 UC005: Use Case <Turn on/off LEDs>

```
LEDs(1, 11, true);
        if(command == "LED1: ON"){
            LEDs(1, 11, true);
          }
          else if(command == "LED1: OFF"){
            LEDs(1, 11, false);
  }
```

Use Case Name	**Open/Close door**	
Use Case ID	**UC001**	
Triggering Event	**In this use case the user will be able to control the door.**	
Description	**The user can open and close the indoor**	
Stakeholder	**User**	
Pre-Condition	**User should enter valid IP Address to open homeX page**	
Post-Condition	**Tab the button to change the state from open to close or vice versa**	
Flow of events	**User** **1. Enter IP Address** **2. Open HomeX** **3. Tab the button**	**System** **1. The system is check if the IP is valid** **2. The system will open homeX** **3. The system will change the state**

3. Table 3.1: Use Case Description for <Open/Close door>

Use Case Name	Open/Close Garage door	
Use Case ID	UC002	
Triggering Event	In this use case the user will be able to control the Open/Close Garage door.	
Description	The user can open and close the Garage door	
Stakeholder	User	
Pre-Condition	User should enter valid IP Address to open homeX page	
Post-Condition	Tab the button to change the state from open to close or vice versa	
Flow of events	User 1. Enter IP Address 2. Open HomeX 3. Tab the button	System 1. The system is check if the IP is valid 2. The system will open homeX 3. The system will change the state

3.1.1.4 Table 3.2: Use Case Description for <Open/Close Garage door >

Use Case Name	Open/Close Garage door	
Use Case ID	UC003	
Triggering Event	In this use case the user will be able to control the Open/Close the window.	
Description	The user can open and close the window.	
Stakeholder	User	
Pre-Condition	User should enter valid IP Address to open homeX page	
Post-Condition	Tab the button to change the state from open to close or vice versa	
Flow of events	User 1. Enter IP Address 2. Open HomeX 3. Tab the button	System 1. The system is check if the IP is valid 2. The system will open homeX

		3. The system will change the state
Use Case Name	**Humidity and Temperature**	
Use Case ID	**UC004**	
Triggering Event	**In this use case the user will be able to monitor the humidity and temperature of the house**	
Description	**The user can see the humidity percentages and the temperature of the house**	
Stakeholder	**User**	
Pre-Condition	**User should enter valid IP Address to open homeX page**	
Post-Condition	**At the top of the web server the user can see information**	
Flow of events	**User** 1. Enter IP Address 2. Open HomeX	**System** 1. The system is check if the IP is valid 2. The system will open homeX 3. The system will show you the information

3.1.1.5 Table 3.3: Use Case Description for <Open/Close window>

Table 3.4: Use Case Description for < Humidity and Temperature >

Use Case Name	Turn on/off LEDs	
Use Case ID	UC004	
Triggering Event	In this use case the user will be able to turn on/off the lights	
Description	The user can turn on/off the lights	
Stakeholder	User	
Pre-Condition	User should enter valid IP Address to open homeX page	
Post-Condition	Tab the button to change the state from on to off or vice versa	
Flow of events	User 4. Enter IP Address 5. Open HomeX 6. Tab the button	System 4. The system is check if the IP is valid 5. The system will open homeX 6. The system will change the state

3.1.1.6 Table 3.5: Use Case Description for < Turn on/off LEDs >

SYSTEM SEQUENCE DIAGRAM

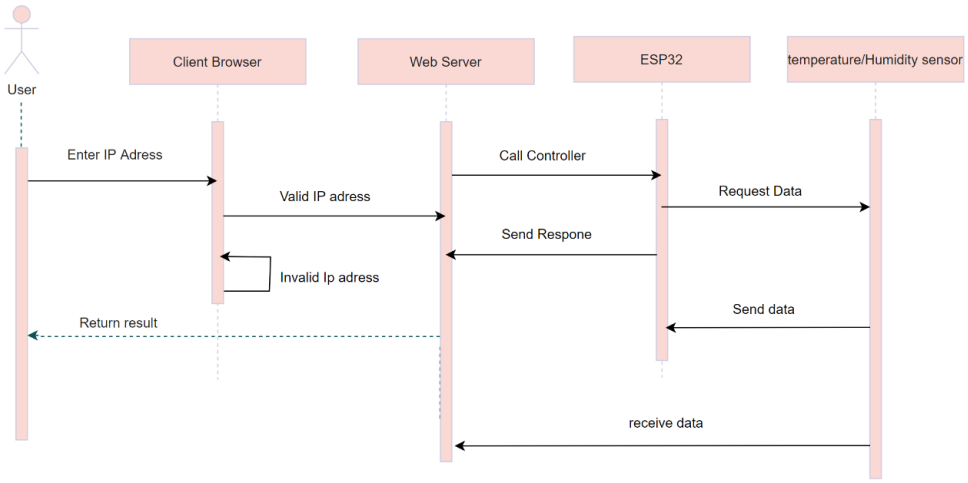


Figure 3.3: System Sequence Diagram of < Humidity and Temperature >

ACTIVITY DIAGRAM

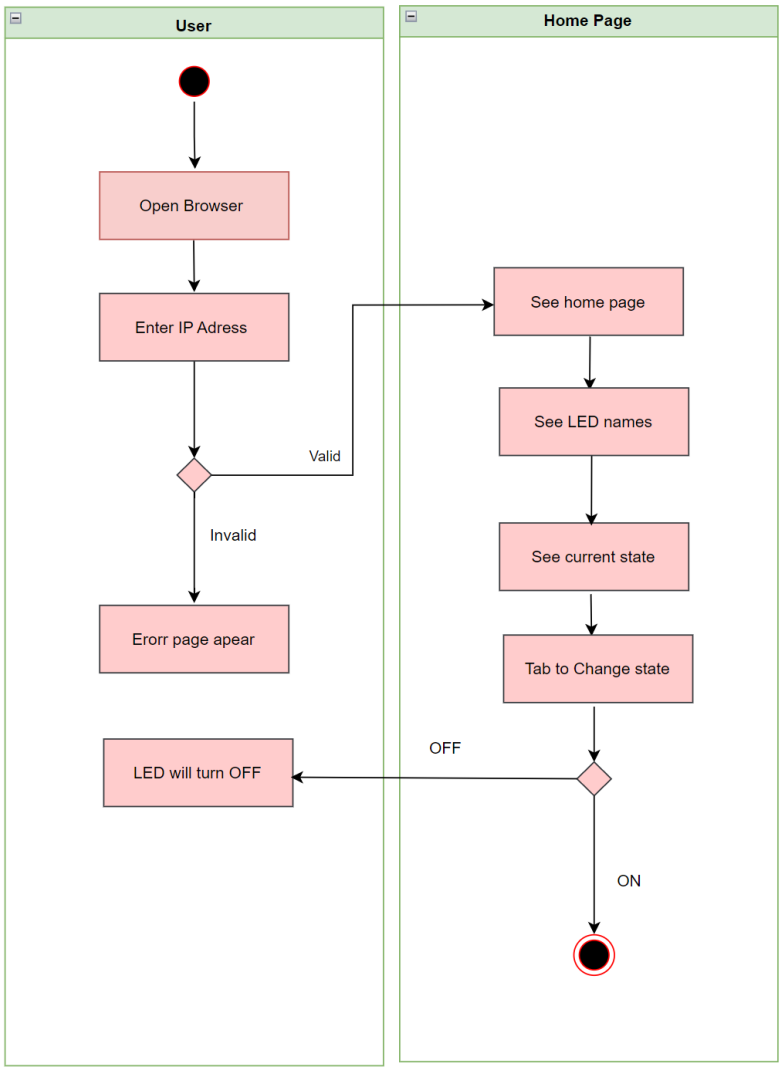


Figure 3.4: Activity Diagram of < Turn on/off LEDs >

SYSTEM REQUIREMENTS

Non- Functional Requirements:

1. Usability
2. Maintainability
3. Adaptability

Functional Requirements:

1. Should have flame sensor
2. Should have temperature sensor
3. Should be open on any smart devices

4. Should be connected with Wi-Fi
5. Should be able to turn on/off the lights
6. Should be able to open/close door
7. Should be able to open/close windows.

Appendix D Software Testing Documentation (STD)

1. Introduction

A product's non-functional testing includes testing its documentation. It might be a form of black-box testing that verifies that system updates and improvements have been documented by making sure that documentation about how to use the system matches what the system really performs

Purpose

The main purpose of testing is to make sure that the system is fully working and there is no error.

1.1 Scope

The scope of the test is to let the user test the product to make sure that they can understand the functionality and write note about their comment to make sure that you change and maintain everything before finalizing the system.

1.2 Definitions, Acronyms and Abbreviation

STD: SOFTWARE TESTING DOCUMENT

1.3 References

Sugali, K. (2021). Software Testing: Issues and Challenges of Artificial Intelligence & Machine Learning.

Graham, D., Black, R., & Van Veenendaal, E. (2021). *Foundations of software testing ISTQB Certification*. Cengage Learning.

1.4 System Overview

Finding bugs in a built product is the process of software testing. Additionally, it helps in the detection of flaws, gaps, and missing requirements by determining whether the actual findings can be reconciled with the anticipated outcomes.

Test Cases, Data and Expected Results

1.5 Test TC001 for Module <Open/Close Door>

This test contains the following test cases:
UC001: User will be able to open and close the door

Test Case ID	Input data	Expected result	Actual result	Pass / Fail
TC001_01_01	No Internet	Error page occur	successful	**Pass**
TC001_01_03	Invalid IP	Not opening the homeX	successful	**Pass**
TC001_01_04	Valid IP	Open homeX home page	successful	**Pass**
TC001_01_05	Valid IP and Status Open	The door will Open	successful	**Pass**
TC001_01_06	Valid IP and Status Close	The door will close	successful	**Pass**

1.6 Test TC002 for Module <Open/Close Garage Door>

UC002: User will be able to open and close the garage door

Test Case ID	Input data	Expected result	Actual result	Pass / Fail
TC001_02_01	No Internet	Error page occur	successful	**Pass**
TC001_02_03	Invalid IP	Not opening the homeX	successful	**Pass**
TC001_02_04	Valid IP	Open homeX home page	successful	**Pass**
TC001_02_05	Valid IP and Status Open	The garage door will Open	successful	**Pass**
TC001_02_06	Valid IP and Status Close	The garage door will close	successful	**Pass**

1.7 Test TC003 for Module <Open/Close Window>

UC003: User will be able to open and close the Window

Test Case ID	Input data	Expected result	Actual result	Pass / Fail
TC001_01_01	No Internet	Error page occur	successful	**Pass**
TC001_01_03	Invalid IP	Not opening the homeX	successful	**Pass**
TC001_01_04	Valid IP	Open homeX home page	successful	**Pass**
TC001_01_05	Valid IP and Status Open	The window will Open	successful	**Pass**

TC001_01_06	Valid IP and Status Close	The window will close	successful	**Pass**

1.8 Test TC004 for Module <Humidity and Temperature>

UC004: User will be able to see humidity and temperature

Test Case ID	Input data	Expected result	Actual result	Pass / Fail
TC001_04_01	No Internet	Error page occur	successful	**Pass**
TC001_04_03	Invalid IP	Not opening the homeX	successful	**Pass**
TC001_04_04	Valid IP	Open homeX home page	successful	**Pass**
TC001_04_05	Valid IP temperature => 25	The fan will work	successful	**Pass**
TC001_04_06	Valid IP temperature<25	The Fan will not work	successful	**Pass**

1.9 Test TC003 for Module <Open/Close Window>

UC005: User will be able to open and close the Window

Test Case ID	Input data	Expected result	Actual result	Pass / Fail
TC001_05_01	No Internet	Error page occur	successful	**Pass**
TC001_05_03	Invalid IP	Not opening the homeX	successful	**Pass**
TC001_05_04	Valid IP	Open homeX home page	successful	**Pass**
TC001_05_05	Valid IP and Status ON	The LED will work	successful	**Pass**
TC001_05_06	Valid IP and Status OFF	The door will not work	successful	**Pass**