
VOICE CONTROL BASED HOME AUTOMATION SYSTEM

16:332:567 SOFTWARE ENGINEERING COURSE PROJECT
DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

FALL, 2013

EDITED BY

LI XINYU

GU YUE

GUO JIAQI

NAYYAR VIDUR

SHU CHANG

ZHANG LUFAN

*The State University of New Jersey
Rutgers*

FOR DETAILS

[HTTPS://SITES.GOOGLE.COM/SITE/VCHOMEAUTOMATION/](https://sites.google.com/site/vchomeautomation/)

Contents

1	CONTRIBUTION BREAKDOWN	3
2	Customer Statement of Requirements(CSR)	4
2.1	Problem Statement	4
2.1.1	Identify the human languages	4
2.1.2	Abundant operations for elements	4
2.1.3	Application on cellphone platform	5
2.1.4	Feedback information	5
2.1.5	Wireless communication and remote controlling	5
2.1.6	User data	6
2.1.7	Check the status of elements	7
2.2	Glossary of Terms	7
3	System Requirements	11
3.1	Enumerated Functional Requirements	11
3.2	Enumerated Nonfunctional Requirements	12
3.3	On-Screen Appearance Requirements	14
4	Functional Requirements Specification	16
4.1	Stakeholders	16
4.1.1	Internal parts	16
4.1.2	External parts	16
4.2	Actors and Goals	16
4.3	Use Cases	16
4.3.1	Casual Description	16
4.3.2	Use Case Diagram	19
4.3.3	Traceability Matrix	26
4.3.4	Fully-Dressed Description	26
4.4	System Sequence Diagrams	28
5	User Interface Specification	34
5.1	Preliminary Design	34
5.1.1	Log In	34
5.1.2	Main Interface	37
5.1.3	Music Interface	40
5.1.4	Light interface	43
5.2	User Effort Estimation	45
5.2.1	User login	45
5.2.2	User logout	46
5.2.3	Light controlling	46
5.2.4	Music controlling	46
5.2.5	Modify the brightness	46
5.2.6	Modify the volume of music	47
5.2.7	Elements status check	47

5.2.8	Control information feedback	47
5.2.9	Light interface	47
6	Domain Analysis	49
6.1	Domain Model	49
6.1.1	Domain model-General	49
6.1.2	Domain model-Server	50
6.1.3	Domain model-Arduino	51
6.1.4	Domain Model-The communication protocol	53
6.2	System Operation Contracts	53
6.2.1	Light controlling	53
6.2.2	Music controlling	55
6.2.3	Modify the brightness	55
6.2.4	Modify the volume of music	55
6.2.5	Elements status check	55
6.2.6	User login	55
6.2.7	Create new users accounts	56
6.2.8	Remote user	56
6.2.9	User log out	56
6.2.10	Check the authority of user	56
6.2.11	Network error notification	56
6.3	Mathematical Model	57
6.3.1	The fuzzy recognition model	57
7	Project Management	58
7.1	Plan of Work	58
7.2	Current problems and possible solutions	58
7.3	Next steps	59
8	Reference	60

1 CONTRIBUTION BREAKDOWN

Here is the contribution breakdown of the project

Responsibility	Li Xinyu	Gu Yue	Guo Jiaqi	Nayyar Vidur	Shu Chang	Zhang Lufan
Project Management(10')	50%	30%	10%			10%
Section1 (9')			100%			
Section2 (6')		33%		33%		34%
Section3 (30')	7%	23%	22%	48%		
Section4 (15')		20%				80%
Section5 (25')	40%				60%	
Section6 (5')	40%	20%			20%	20%
Total (100')	18.5	16	16.5	16.5	16.5	16

Table 1: CONTRIBUTION BREAKDOWN FOR REPORT 1.

2 Customer Statement of Requirements(CSR)

2.1 Problem Statement

As the modern software system, home automation control system should achieve any requirements from users by its intelligence and generalization. In our home automation system, the requirements of customer can be concluded as below:

2.1.1 Identify the human languages

It is impossible for any users to understand the instructions or algorithm of programming. What they can do is control the elements with human languages and signal of body. So the users require the system to identify the human languages instead of machine instructions or other programming languages.

Human voice instruction Voice is the most directly and simple way for people to convey their requirement. For example, when the sunlight is insufficient inside the room, the user can just send a voice message just like Turn on the light to modify the brightness of the room. This condition requires the excellent flexibility because different users maintain the various ways of speaking and the system should be compatible to cater to a large number of people. And this requirement is the core case of the whole system.

Simple gestures Some users also demand the system to be equipped with the alternative controlling ways. And gesture is a good choice. Gesture s the easiest way of communicating for some one who cannot speak. For example the user can slide one of his fingers to modify the volume of MP3 player. The technology of identifying gestures for system is complex in algorithm and hardware design, but it can really simplify the operation of users.



Figure 1: Model of User Input.

2.1.2 Abundant operations for elements

In general the users have more requirements to control the elements of the system than just turning it on and off. For the lamps, if the users want to

modify the brightness of the room, users can change the brightness of the lamp instead of turning it on or off. Or if the user is using the MP3 player, he can modify the volume or skip to the next or to the last song. This requirement is important because the users in general have their own customs of using the system and the system should make the users comfortable with the abundant operations of the elements.

2.1.3 Application on cellphone platform

When the users are using the system, it is inconvenient and impossible to hold several terminals such as cellphones, iPads and controllers. For users, it is significant to concentrate the functions of different devices into one terminal. In general, the most frequently used device is cellphone for users. So the users require the system to be developed on the cellphone platform such as Android and IOS. As one application of mobile device, the users can log into the home automation system directly with less cost than a whole controller. Android is the most widely used mobile system and it can be compatible with a large number of users if the system developed on Android. Another reason is that the Android system can easily process the voice information because it is a Linux based OS and it can easily access Google Voice. Also Android primarily uses for touchscreen devices [1]. So for users the Android system is the best choice of platform for our home automation system.

2.1.4 Feedback information

At any point of time, the user might want to know the status of the different elements of the house. For this purpose the user would require the system to send a feedback to the user interface. The users should know whether the system recognizes the correct information and execute the right actions. Also if the identification failed, the users should be announced to send the messages again or cancel the action by the feedback information.

2.1.5 Wireless communication and remote controlling

The users prefer controlling the elements at their house remotely using a cell phone. So the system should support wireless communication. Another problem is that WIFI and Bluetooth and other high-speed type of communication cannot be available all the families in general. So the system should use the TCP/IP protocol between the communication of server and the central controller. In addition, the users demand the wireless communication functions of elements of the home automation system because to increase the flexibility of the system by allowing the user to change the position of the various elements of the home automation system. This flexibility is lost in case of wired system because of the constraints due to using wires.

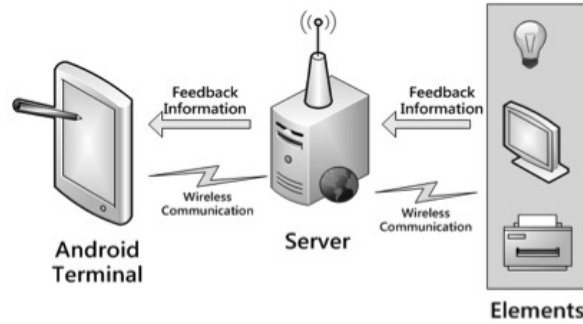


Figure 2: Model of Communication.

2.1.6 User data

The users demand the system to storage the data of different IDs. In general users have two aspects of requirements of users data. First the users have their own customs of living. With the data of history of special users, the system can easily recognized the messages the users send, and create a user friendly interface for different users. Also the system can create several classes for many users. With this function, the system can limit some functions for special classes or groups. For example, if the user A and B have been divided into the class Kids, A and B cannot operate some limited elements such as microwaves. In this way the system can also guarantee the safety of the house and its users.

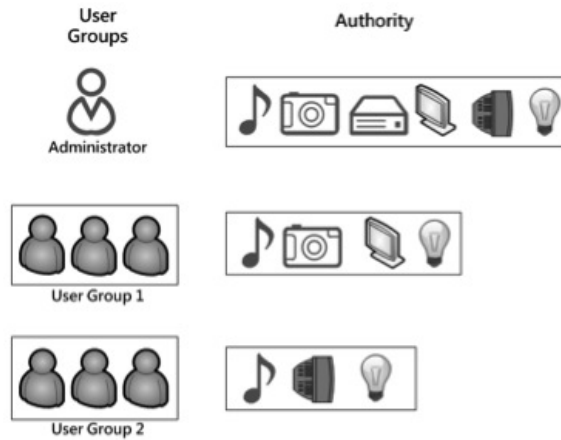


Figure 3: Model of User Group.

2.1.7 Check the status of elements

The users should have the authority to check the status of the element. That requires a new window in UI design to display the status of any elements in the system. And A running log is also needed to assist the people taking care of the maintenance of the system to provide the better user experience.

2.2 Glossary of Terms

1. Network service: a data storage, manipulation, presentation, communication or other shared function provided by one device and consumed by others.

2. Client (computing): software that accesses a remote service on another computer.

3. Machine learning: a branch of artificial intelligence, concerns the construction and study of systems that can learn from data. For example, a machine learning system could be trained on email messages to learn to distinguish between spam and non-spam messages. After learning, it can then be used to classify new email messages into spam and non-spam folders.

4. Decision tree learning: Decision tree learning uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value.

5. Classifier: An algorithm that implements classification, especially in a concrete implementation, is known as a classifier. The term "classifier" sometimes also refers to the mathematical function, implemented by a classification algorithm that maps input data to a category.

6. Thread of execution: In computer science, a thread of execution is the smallest sequence of programmed instructions that can be managed independently by an operating system scheduler. A thread is a light-weight process. The implementation of threads and processes differs from one operating system to another, but in most cases, a thread is contained inside a process.

7. Socket: A network socket is an endpoint of an inter-process communication flow across a computer network. Today, most communication between computers is based on the Internet Protocol; therefore most network sockets are Internet sockets.

8. Packet: A network packet is a formatted unit of data carried by a packet-switched network. Computer communications links that do not support packets, such as traditional point-to-point telecommunications links, simply transmit data as a bit stream. When data is formatted into packets, the bandwidth of the communication medium can be better shared among users than if the network were circuit switched.

9. TCP/IP: The Internet protocol suite is the networking model and a set of communications protocols used for the Internet and similar networks. It is commonly known as TCP/IP, because its most important protocols, the Transmission Control Protocol (TCP) and the Internet Protocol (IP) were the first networking protocols defined in this standard.

10. Android software development: Android software development is the process by which new applications are created for the Android operating system. Applications are usually developed in the Java programming language using the Android Software Development Kit, but other development tools are available.

11. Android SDK: The Android software development kit (SDK) includes a comprehensive set of development tools.[6] These includes a debugger, libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.5.8 or later, Windows XP or later; for the moment one can develop Android software on Android itself by using [AIDE - Android IDE - Java, C++] app and [Android java editor] app.

12. Application programming interface: An application programming interface (API) specifies how some software components should interact with each other. In addition to accessing databases or computer hardware, such as hard disk drives or video cards, an API can be used to ease the work of programming graphical user interface components. In practice, many times an API comes in form of a library that includes specifications for routines, data structures, object classes, and variables. In some other cases, notably for SOAP and REST services, an API comes as just a specification of remote calls exposed to the API consumers.

13. Internet protocol suite: The Internet protocol suite is the networking model and a set of communications protocols used for the Internet and similar networks. It is commonly known as TCP/IP, because its most important protocols, the Transmission Control Protocol (TCP) and the Internet Protocol (IP) were the first networking protocols defined in this standard. It is occasionally known as the DoD model, because the development of the networking model was funded by DARPA, an agency of the United States Department of Defense.

14. Java class file: Java class file is a file (with the .class filename extension) containing a Java bytecode which can be executed on the Java Virtual Machine (JVM). Java class file is produced by Java compiler from Java programming language source files (.java files) containing Java classes. If a source file has more than one class, each class is compiled into a separate class file. JVMs are available for many platforms, and the class file compiled in one platform will execute in a JVM of another platform. This makes Java platform-independent.

15. UI: The user interface, in the industrial design field of humanmachine interaction, is the space where interaction between humans and machines occurs.

16. User experience: involves a person's behaviors, attitudes, and emotions about using a particular product, system or service. User experience includes the practical, experiential, affective, meaningful and valuable aspects of human-computer interaction and product ownership. Additionally, it includes a persons perceptions of system aspects such as utility, ease of use and efficiency.

17. HCI: human computer interaction

18. Adobe Photoshop: Adobe Photoshop is a graphics editing program developed and published by Adobe Systems.

19. Demo: Within the computer subculture known as the demo scene, a non-interactive multimedia presentation is called a demo (or demonstration). Demo groups create demos to demonstrate their abilities in programming, music, drawing, and 3D modeling.

20. HUI: Handset User Interface.

21. Customer The enterprise or department that might want to deploy our system.

22. Arduino Mega IThe Arduino Mega is a microcontroller board based on the ATmega 1280. It has 54 digital input/output pins (of which 14 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. The Mega is compatible with most shields designed for the Arduino Duemilanove or Diecimila.

23. Database A database is a collection of information that is organized so that it can easily be accessed, managed, and updated. In one view, databases can be classified according to types of content: bibliographic, full-text, numeric, and images.

24. Delivery This term can refer (i) to the process of extraction of the package from the box by the receiver or (ii) to the whole process of delivery (from the booking to the actual delivery).

25. Inter-building delivery: A delivery process where both the sender and the receiver are in different buildings.

26. Intra-building delivery: : A delivery process where both the sender and the receiver are in the same building.(may or may not be in the same floor).

27. Mobile interface: A web-based user interface working on Android smart phones, where users can view and execute operations through this application. These operations include: book a delivery, check status of package/mail, personal information management, and system maintenance.

28. Properties of security: Refers to the confidentiality, authenticity, integrity and non-repudiation properties of a secure information transfer.

3 System Requirements

3.1 Enumerated Functional Requirements

In this section, we will extract and analysis the basic and main functional requirements [2] from Customer Statement of Requirements (CSR). By using User Stories [3], we may make the system requirements easier to understand. The table 3.1 is the functional requirements, which are extracted from CSR.

As we know, customers and end users are always interested in the requested functional and costs, which means our groups work should base on the customers requirements, gathering the main requirements from the customers is the first step, however, these requirements cannot be used directly, because customers just present the requirements for us but do not consider if these requirements can be realized or not and do not regard the priorities in their requirements. Hence, we should negotiate with the customers to decide which part is essential, which is desired but not mandatory, which is optional, and which will realize in future.

Basing on the principles mentioned above, we issued the requirements from the CSR. Most of the customers consider the basic system should contain some important functions such as turn on or turn off the lamp, locking or opening the door, and start or stop the music system in their home (ST-3), and they prefer all these functions can be control by voice. This requirement is the basic request of most customers and considering the voice control application are widely used in cellphone market, it can be realized, that why we give it 5 PW. Cellphone application is also a basic and essential requirements for the customers, considering more and more people prefer using cellphone in their daily life, the home automation should also be applied in this field, which is the most convenient and comfortable method to control the home devices. ISO [4] and Android are two types application platforms, considering the market information and development cost, our group decide to use Android application platform [5] to develop our voice based home automation system (ST-4). Besides this two parts, using this application in a remote place is also an important points for the customers, they wish to use this system without the limitation of the distance, therefor, applying the WIFI and Bluetooth and other high-speed type of communication is necessary for us (ST-7). These parts are the main points in the customers requirements. There might be some other points which the customers mentioned, considering the some people prefer using button rather than voice or they may be inconvenient to use voice control in some places, button control is also needed in our design (ST-5), and some others informed us that they wish to know the statues of their home devices and encourage us put into this function in the system (ST-10).

Again, the main points of the system is (ST-3) and (ST-4), to realize this two part is the basic goal of the whole application, at the same time, some related parts may be added into our system, such as voice command device[6] by hu-

ID	PW	User Story
ST-1	4	As a user, I can use the control system by human language and receive the feedback information, which are also human language.
ST-2	1	As a user, I can use my body signal or gestures to turn off or on the lamp or the music system in my house.
ST-3	5	As a user, I can turn on or turn off my home lamp and my music stereo by voice control.
ST-4	5	As a user, I can use this system by my cellphone.
ST-5	3	As a user, I wish to use some buttons to control my house devices.
ST-6	3	As a user, I can get the feedback information, such as whether the system recognizes the correct information and executes the right actions, and I also want to announce and send the messages again or cancel the action by the feedback information. Also the system should notify if the state of some element has been altered manually.
ST-7	4	As a user, I can use this system to control my home devices even I am far away from my home.
ST-8	2	As a user, I can set some limited functions for special people.
ST-9	3	As a user, I can add or eliminate new devices into the system.
ST-10	2	As a user, I can check the statues of my home devices by this system.

Table 2: Functional User Stories.

man languages, remote pattern, button control, statues checking, and feedback information. However, even some points of the requirements are given 3 PW and 2 PW, the costs and limitation of technology make them cannot be realized (ST-9, ST-8). As for the body signal or gestures control, only small group customers required it and the technology is very hard for us, for these reasons we define it as 1 point.

3.2 Enumerated Nonfunctional Requirements

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. This should be contrasted with functional requirements that define specific behavior or functions [7]. According to the FURPS [8] definition of non-functional requirements, we would like to divide our non-functional requirements into 5 parts: Accessibility, Usability, Reliability, Performance and supportability. And all the requirements will be

treated as user stories in this section [9].

Accessibility Accessibility is the degree to which a product, device, service, or environment is available to as many people as possible. Accessibility can be viewed as the "ability to access" and benefit from some system or entity. In this case our system should be accessible to different kind of people and different kind of devices. People can use our control system in different kind of devices based on android platform (ST-11).

Usability Usability is the ease of use and learnability of a human-made object. Though we make our system based on voice control there are still some concerns about usability. The voice recognition systems usually have troubles with some voice with accent, however a users requires should be accepted and execute. If the system cannot recognize or fully recognize the users command, it should give user some related options based on the command (ST-12).

Also a system based on voice control doesnt mean the system can only controlled by voice commands. Users should be able to control the system by press the buttons or by typing the settings (ST-13). With some touch based control method people can control the system without talking.

Reliability Reliability emphasizes dependability in the lifecycle management of a product. Dependability, or reliability, describes the ability of a system or component to function under stated conditions for a specified period of time. People usually live in a house for 5 years or longer [10], so the both the software and the hardware should be able to working a longtime without maintenance (ST-14).

The hardware in the system should be safe enough to control the switch of some electronic equipment. If user wants to cut off the power, the user should be able to turn on and turn off the smart element by hand (ST-15).

Also a user should be able to track all the command has been made and the system should keep tracking users commands (ST-16). The log helps user and the system to understand the users habits better and improve the quality of voice recognition by predicate users commands.

Performance Performance is characterized by the amount of useful work accomplished by a system compared to the time and resources used. A user should get feedback in a very short time after he gives the commands (ST-17), just like talking to a real people. The processing time here should be short enough and the system should relay on the server to do the recognition job and complex calculation works.

ID	PW	User Story
ST-11	3	As a user, I should be able to control my system in different kind of devices based on android platform.
ST-12	4	As a user, I should be given some related options based on my commands if my commands are not recognized correctly.
ST-13	4	As a user I should be able to control the system with my android device based on touch screen or input but not only with voice commands.
ST-14	3	As a user, I should be able to use both the software on the android and the hardware for a longtime without maintenance.
ST-15	2	As a user I should be able to track all the commands I have sent.
ST-16	3	As a user, I should be able to turn on and turn off the smart element in my home with my hands.
ST-17	4	As a user, I should receive some feedbacks after I send a voice commands, whether the commands has been recognized correctly or not.
ST-18	3	As a user, I should be able to find some documentation about the home automation system online and fix some little problems with the help of the documentation.
ST-19	2	As a user I should be able to update my software on the android if necessary .

Table 3: Non-functional User Stories.

Supportability Supportability refers to the ability of technical support personnel to install, configure, and monitor computer products, identify exceptions or faults, debug or isolate faults to root cause analysis, and provide hardware or software maintenance in pursuit of solving a problem and restoring the product into service. Incorporating serviceability facilitating features typically results in more efficient product maintenance and reduces operational costs and maintains business continuity. (ST-18) suggests that there should be a set of documentation of the system so that the system can be maintained with some documentation. Also (ST-19) suggests that there should be a software update to keep the software working in the best condition.

3.3 On-Screen Appearance Requirements

Our program is home automation based on voice, that means users can control their home more conveniently, what they need to do is just having a conversation with their mobile phones. This UI design [11] is based on Android system [12].

UI [13] is important, why we say that? As we all know, most users of the prod-

uct dont have the enough knowledge of electronic and software, what they can know about their electronic product always via UI. What they can get about the information of product is through clicking the button or rolling along the screen. Thus, our destination of UI design is making a kind of clear and neat interface, which can help us attract more people to generate interest in our product. To be clear can make sure that users can understand the functions of the device [14] and can easily learn how to use it, and guarantee that they wont be confused by what they see. To be neat is not equal to to be simple, it means that our interface should make users feel comfortable, we try our best to make them achieve their purpose with the least steps. At last, we should make our interface looks harmonious and attractive [15].

We cannot promise that all users are satisfied with our UI, in fact, no designer is able to do it, but if over 80

ID	PW	Requirement
ST-20	5	The interface shall make users be able to input their voice through convenient, clear and easy way.
ST-21	4	The interface shall give users feedback via image or word. The interface should inform users that their voice is received when it is recording.
ST-22	3	The interface shall offer additional button which users can control the system by clicking button in some certain situation.
ST-23	3	The interface shall show users the status of their devices by icon or buttons.
ST-24	2	The interface should support new devices to guarantee that users do not need to abandon their interface when new devices are added.
ST-25	2	The interface shall jump to right desk when it receives certain information and signal.
ST-26	1	The interface should own appropriate overall arrangement and color which can make users feel comfortable when they use it.

Table 4: On-Screen Appearance Requirements .

The pictures below is the preliminary drafts of the interfaces of our system which will be polished and embellished in future, it shows the basic function of our program. Overall, it has 4 separate interfaces: log in interface which is created to guarantee the security and privacy of users [16], main interface through which users can input command, music interface and light interface [17] through which users can check the status of music and light.

4 Functional Requirements Specification

4.1 Stakeholders

In general, the stakeholders of our system can be divided into two different parts: the internal parts and the external parts.

4.1.1 Internal parts

Developers: the developers of our system are a person or a group people who code and design the processes of the whole program, our developers have the right to access all section of the system and interest to realize the function for the system, which required by the end users and customers.

4.1.2 External parts

End users: end users may a person or a group who interest to use and investigate our system.

Customers: homeowners or a group people who share a home are our main customers, they always interest the reliability, value, function, cost performance, and quality of our system.

Investigator: investigator may be a person or a group people who are interested in market, customers, and the user and customer feedback.

4.2 Actors and Goals

Our systems actors and goals are indicated in the table 5.

4.3 Use Cases

4.3.1 Casual Description

UC-1 Light controlling Allow the users to turn on or off the lights by sending the voice message to the server from the Android based cellphone.

UC-2 Music controlling Allow the users to control the music player by their voice. And the server sends the control instruction to the MCU after processing the voice message.

UC-3 Modify the brightness The users can modify the brightness of the light by their voice command. Depending on the voice command, the system will increase or decrease the brightness of the light.

Actor	Category	Goals
Users	Initiating	To turn on or turn off the light and music (by voice), to control the light intensity and music volume (by voice).
Device System (Collector)	Initiating and Participating	To collect and compile the information from the users and send the code information to the central sever.
Device System (Server)	Initiating	To analysis the information from the device collector, sending the feedback to the user interface and signal to the arduino.
Arduino	Initiating and Participating	To collect the signal from the sever and control the light and music switches.
Light Switch	Participating	To turn on or turn off the light and control light intensity.
Music Switch	Participating	To turn on or turn off the music and control music volume.
Communication Network	Participating	To transmit signal by wifi and internet.
Administrator	Initiating	To manage the administrators own authorities and enable administrator increase or decrease their authorities in the management interface.

Table 5: Actors and Goals

UC-4 Modify the volume of music It allows the users to increase or decrease the volume of music player by sending the related voice message to the music system.

UC-5 Elements status check The system allows the users to check the current status of elements in system. It is can either be called upon by the user or if the state of the device is changed manually. The elements can send the status information to the server and the server process this information. The elements can send the status information to the system and the system can relay the status to the user by displaying it on the user interface.

UC-6 User login It allows the authorized users to access into the system.

UC-7 Create new users accounts The system allows the new users to create their own accounts. By following a few easy steps given in the user interface.

UC-8 Remote user It allows the administrator to remote the deserted users. The deserted usernames cannot be used again and their data in server will be deleted.

UC-9 User log out The system allows the administrator to manage the authorities of different users. Based on the requirements, in our system the different users have different authorities to access the elements. And the system administrator can increase or decrease their authorities in the management interface.

UC-10 Authority management The system allows the administrator to manage the authorities of different users. Based on the requirements, in our system the different users have different authorities to access the elements. And the system administrator can increase or decrease their authorities in the management interface.

UC-11 Check the authority of user Administrator can check the authorities of all users in order to management. And the users can check their own permissions to know whether they can do the operations.

UC-12 Control information feedback It allows the users to check the feedback information of voice message in order to check whether the voice have been identified by the server. If the message identification failed, the feedback information can lead the user to send again or cancel the operation.

UC-13 Network error notification The system offers the notification of network error if the signal of network is weak. The signal can be separate into 2 parts: Between cellphone and server; between server and MCU. In such an event, the system will notify the user using the interface device.

4.3.2 Use Case Diagram

All the use case diagrams are showed as follows:

User Case 1 Light Controlling
<p>Related Requirements: ST-3,ST-1,ST-2,ST4,ST20</p> <p>Initiating Actor: User</p> <p>Actors Goal: To turn on or off the light</p> <p>Participating Actor: Light and Arduion</p> <p>Precondition: User is an authorized person</p> <p>Success End Condition: The light is turn on or off by the users instruction</p> <p>Failed End Condition: Server has not identified the message of user</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>← 1. Cellphone displays the main menu of application.</p> <p>→ 2. User selects the light function.</p> <p>← 3. Cellphone displays the light function interface.</p> <p>→ 4. Player sends their voice message to the server.</p> <p>← 5. Server processes the message and sends a signal to the Arduino.</p> <p>← 6. The Arduino receives the signal from server and communicates it to the MCU and invokes UC5(status check).</p> <p>← 7. The MCU receives the signal from the Arduino and turns on or off the switch.</p>

Table 6: User Case 1 Light Controlling .

User Case 2 Music Controlling
Related Requirements: ST-3,ST-1,ST-2,ST4,ST20 Initiating Actor: User Actors Goal: To turn on or off the music player Participating Actor: Music player and Arduion Precondition: User is an authorized person Success End Condition: The music player is turn on or off by the users instruction Failed End Condition: The message of user have not been identified by server Extension Point: Flow of Events for Main Success Scenario: ← 1. Cellphone displays the main menu of application. → 2. User selects the music function. ← 3. Cellphone displays the music function interface. → 4. Player sends their voice message to the server. ← 5. Server processes the message and sends a signal to the Arduino. ← 6. The Arduino receives the signal from server and communicates it to the MCU and invokes UC5(status check).. ← 7. The MCU receives the signal from the Arduino and turns on or off the switch.

Table 7: User Case 2 Music Controlling .

User Case 3 Modify brightness
Related Requirements: ST-3,ST-1,ST-2,ST4,ST20 Initiating Actor: User Actors Goal: To modify the brightness of the light Participating Actor: Light and Arduion Precondition: User is an authorized person Success End Condition: The brightness of the light has been modified follow the users instruction Failed End Condition: Server has not identified the messages of user Extension Point: Flow of Events for Main Success Scenario: ← 1. Cellphone displays the main menu of application. → 2. User selects the light function. ← 3. Cellphone displays the light function interface. → 4. Player sends their voice message to the server. ← 5. Server processes the message and sends a signal to the Arduino. ← 6. The Arduino receives the signal from server and communicates it to the MCU and invokes UC5(status check). ← 7. The MCU receives the signal from the Arduino and modifies the brightness.

Table 8: User Case 3 Modify brightness.

User Case 4 Modify the Volume of Music
Related Requirements: ST-3, ST-1,ST-2,ST4,ST20 Initiating Actor: User Actors Goal: To modify the volume of the music player Participating Actor: Music player and Arduion Precondition: User is an authorized person Success End Condition: The volume of music player has been modified follow the users instruction Failed End Condition: Server has not identified the messages of user Extension Point: Flow of Events for Main Success Scenario: ← 1. Cellphone displays the main menu of application. → 2. User selects the music function. ← 3. Cellphone displays the music function interface. → 4. Player sends their voice message to the server. ← 5. Server processes the message and sends a signal to the Arduino. ← 6. The Arduino receives the signal from server and communicates it to the MCU and invokes UC5 (status check). ← 7. The MCU receives the signal from the Arduino and modifies the volume.

Table 9: User Case 4 Modify the Volume of Music.

User Case 5 Elements Status Check
Related Requirements: ST-6, ST-10, ST-24 Initiating Actor: User or Arduion Actors Goal: To check the current status of smart elements in system Participating Actor: Light, Music player, Server and micro-controller Precondition: a. User is an authorized person. b. Communication between the server and smart elements is connected Success End Condition: The status of elements display on the interface of cellphone Failed End Condition: The status information conveys failure between server and smart elements Extension Point: Flow of Events for Main Success Scenario: ← 1. Cellphone displays the main menu of application. → 2. User selects the status check function. ← 3. The elements send a status message to the MCU and MCU sends the information to the Arduino. ← 4. Arduino sends message to the server and server sends to the cellphone after processing. ← 5. Cellphone receives the message and displays the information of status on UI.

Table 10: User Case 5 Elements Status Check.

User Case 6 User Login
<p>Related Requirements: ST-8</p> <p>Initiating Actor: User, Administrator</p> <p>Actors Goal: To login the system by the username and password</p> <p>Participating Actor: Server</p> <p>Precondition: Users account has been created in the server</p> <p>Success End Condition: Users account has been created in the server</p> <p>Failed End Condition: The user enters the incorrect username or the password is not corresponding with the username</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>← 1. Cellphone displays the login interface.</p> <p>→ 2. User enters the username and password.</p> <p>← 3. Cellphone send the username and password to the server.</p> <p>← 4. The server checks the username and password in the database and sends the feedback information to the cellphone.</p> <p>← 5. The cellphone display the main menu to the user.</p>

Table 11: User Case 6 User Login.

User Case 7 Create New User Accounts
<p>Related Requirements: ST-8</p> <p>Initiating Actor: User</p> <p>Actors Goal: To create the new accounts for the new users and store in server</p> <p>Participating Actor: Server</p> <p>Precondition: Username of new account has not been occupied and the number of users of system less than the maximum users the server can hold</p> <p>Success End Condition: The user account has been created and the information has been stored in the server</p> <p>Failed End Condition: The username has been occupied or the password is out of requirement</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>Include: Login (UC-6)</p> <p>← 1. Cellphone displays the creating new account interface.</p> <p>→ 2. User enters the personal information.</p> <p>← 3. Cellphone identifies whether the username and password obey the rules.</p> <p>← 4. Cellphone send the information to server and the server stores the information about the new accounts.</p> <p>← 5. Server sends the feedback information to the cellphone and the cellphone display the login interface to the user.</p>

Table 12: User Case 7 Create New User Accounts.

User Case 8 Remove Users
Related Requirements: ST-8 Initiating Actor: Administrator Actors Goal: TTo remove a deserted or illegal user Participating Actor: Server Precondition: a. The information of deserted or illegal user is stored in server. b. Current user must be administrator. Success End Condition: Administrator deletes the information of users in server Failed End Condition: The users information cannot be found in server Extension Point: Flow of Events for Main Success Scenario: Include: Login (UC-6) ← 1. Cellphone displays the user management interface. → 2. Administrator selects the user that will be removed. ← 3. Cellphone send the remove instruction to server. ← 4. Server deleted the related information of user. ← 5. Server sends the feedback information to the cellphone and the cellphone display the current users list.

Table 13: User Case 8 Remove Users.

User Case 9 User Log Out
Related Requirements: ST-8 Initiating Actor: User, Administrator Actors Goal: To log out from the system and protect personal information Participating Actor: Server Precondition: User has already logged into the system Success End Condition: The user has logged out of the system and no operations can be done Failed End Condition: None Extension Point: Flow of Events for Main Success Scenario: Include: Login (UC-6) ← 1. Cellphone displays the main menu of application. → 2. User clicks the log out button. ← 3. Cellphone clear the information of current user and display the login interface.

Table 14: User Case 9 User Log Out.

User Case 10 Authority Management
<p>Related Requirements: ST-8</p> <p>Initiating Actor: Administrator</p> <p>Actors Goal: To manage the authorities of different users</p> <p>Participating Actor: Server</p> <p>Precondition: a. Current user must be the administrator. b. There are 1 user at least in the database.</p> <p>Success End Condition: The authorities of users have been modified after management</p> <p>Failed End Condition: The authorities of users have not be changed</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>Include: Login (UC-6)</p> <p>← 1. Cellphone displays the management interface of administrator.</p> <p>→ 2. Administrator selects the users need to be modified.</p> <p>← 3. Cellphone display the current authorities of this user.</p> <p>→ 4. Administrator modifies the authorities of this user.</p> <p>← 5. Cellphone send the modified information to the server and the server modifies the information in database.</p> <p>← 5. Server sends the feedback information to the cellphone and displays the current authorities of user on interface.</p>

Table 15: User Case 10 Authority Managemen.

User Case 11 Check the Users Authority
<p>Related Requirements: ST-8</p> <p>Initiating Actor: User</p> <p>Actors Goal: To check the current users authority</p> <p>Participating Actor: Server</p> <p>Precondition: a. Authorized user have logged into the system. b. Network between server and cellphone should be connected</p> <p>Success End Condition: The users authority displays on interface</p> <p>Failed End Condition: The communication between server and cellphone has been interrupted</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>Include: Login (UC-6)</p> <p>→ 1. User selects the authority function.</p> <p>← 2. Cellphone sends the check instruction to the server.</p> <p>← 3. Server checks the database and sends the information of authority back to the cellphone.</p> <p>← 4. Cellphone displays the authority on the interface.</p>

Table 16: User Case 11 Check the Users Authority.

User Case 12 Control Information Feedback
<p>Related Requirements: ST-1, ST-12, ST17, ST21</p> <p>Initiating Actor: Server</p> <p>Actors Goal: Send the feedback information to the user interface</p> <p>Participating Actor: User, cellphone</p> <p>Precondition: Network between server and other parts of system should be connected</p> <p>Success End Condition: The feedback information sends to the cellphone and displays on the screen</p> <p>Failed End Condition: The communication between server and other parts has been interrupted</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>→ 1. The server identifies the voice message after user sending the voice messages.</p> <p>← 2. The server processes the feedback information and sends back to the cellphone.</p> <p>← 3. Cellphone receives the feedback and displays on the interface.</p>

Table 17: User Case 12 Control Information Feedback.

User Case 13 Network Error Notification
<p>Related Requirements: ST-8, ST17</p> <p>Initiating Actor: Cellphone</p> <p>Actors Goal: To give a notification of network error to the user</p> <p>Participating Actor: User</p> <p>Precondition: The communication between cellphone and other parts of system has been interrupted</p> <p>Success End Condition: The network error notification has been displayed on the cellphone</p> <p>Failed End Condition: The network has no error currently</p> <p>Extension Point:</p> <p>Flow of Events for Main Success Scenario:</p> <p>Include: Login (UC-6)</p> <p>← 1. Cellphone checks the current status of network.</p> <p>← 2. If the network status is abnormal, the cellphone displays an error notification on the screen.</p>

Table 18: User Case 13 Network Error Notification.

4.3.3 Traceability Matrix

The Traceability Matrix is showed in the Table 19 in next page.

4.3.4 Fully-Dressed Description

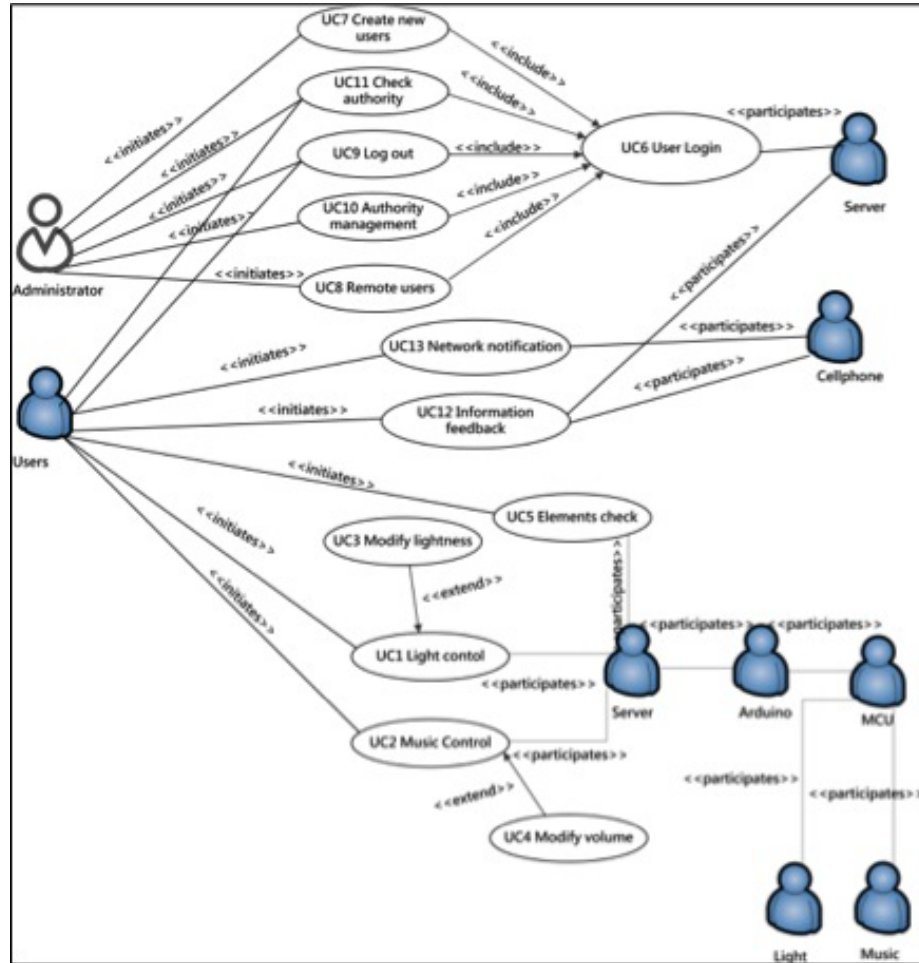


Figure 4: Use Case Diagram.

REQ	PW	UC1	UC2	UC3	UC4	UC5	UC6	UC7	UC8	UC9	UC10	UC11	UC12	UC13
ST1	4	x	x	x	x								x	x
ST2	1	x	x	x	x									
ST3	5	x	x	x	x									
ST4	5	x	x	x	x									
ST6	3					x								
ST8	2						x	x	x	x				
ST10	2					x								
ST12	4												x	
ST17	4												x	x
ST20	5	x	x	x	x								x	x
ST21	4												x	
ST24	2													
Max	PW	5	5	5	5	3	2	2	2	2	2	2	4	4
Total	PW	20	20	20	20	5	2	2	2	2	2	2	16	10

Table 19: Traceability Matrix

4.4 System Sequence Diagrams

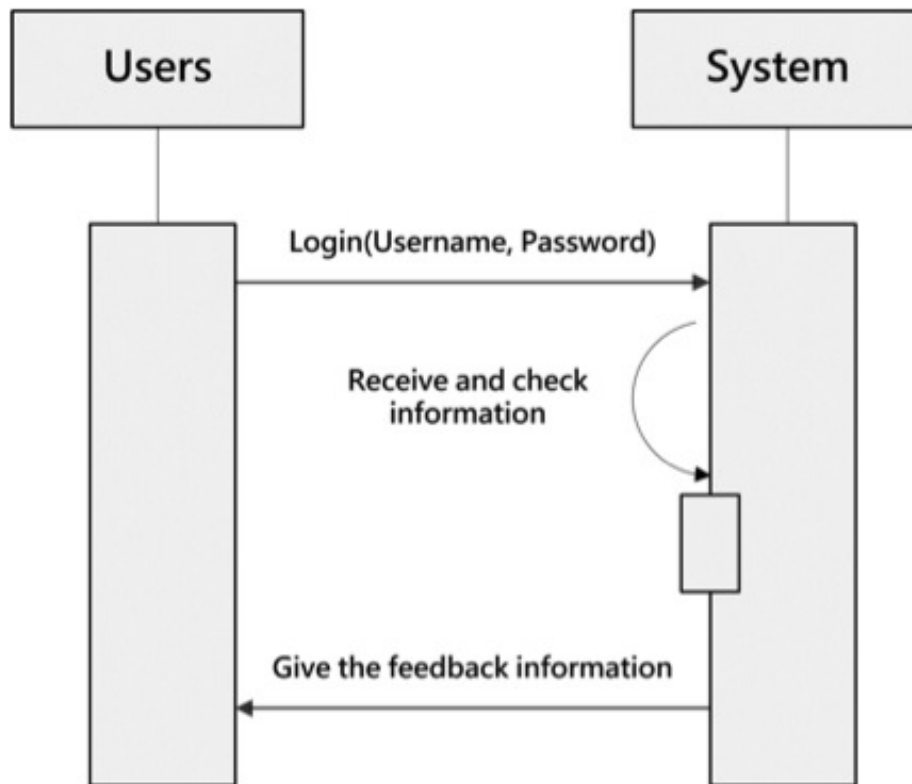


Figure 5: Login sequence diagram UC-6.

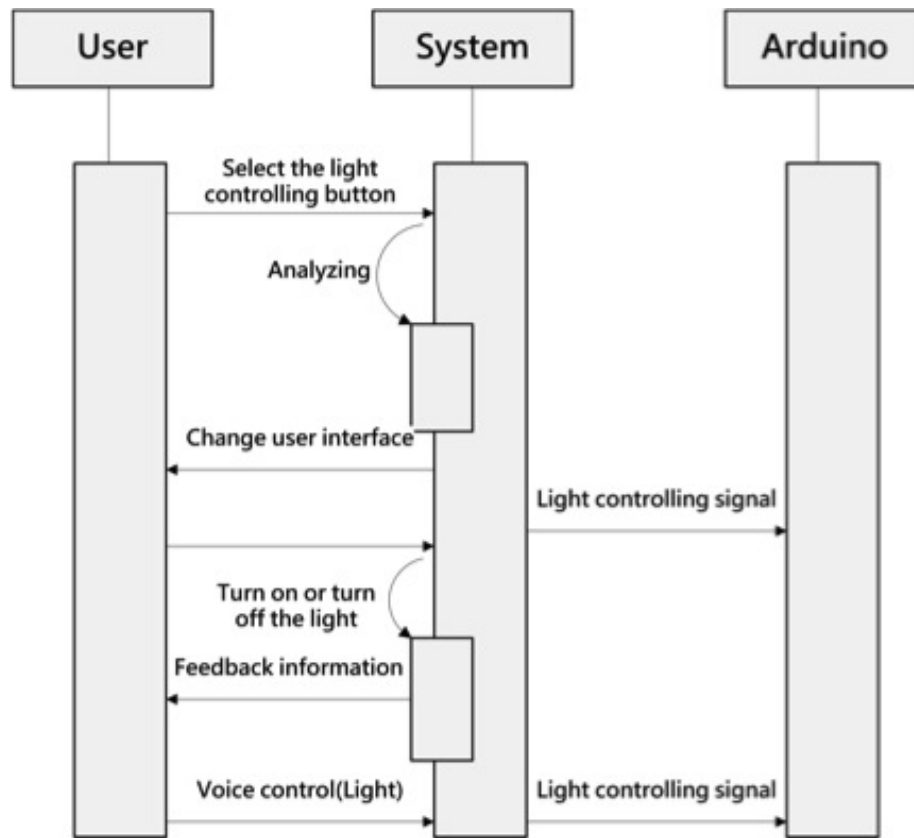


Figure 6: Light controlling sequence diagram UC-1.

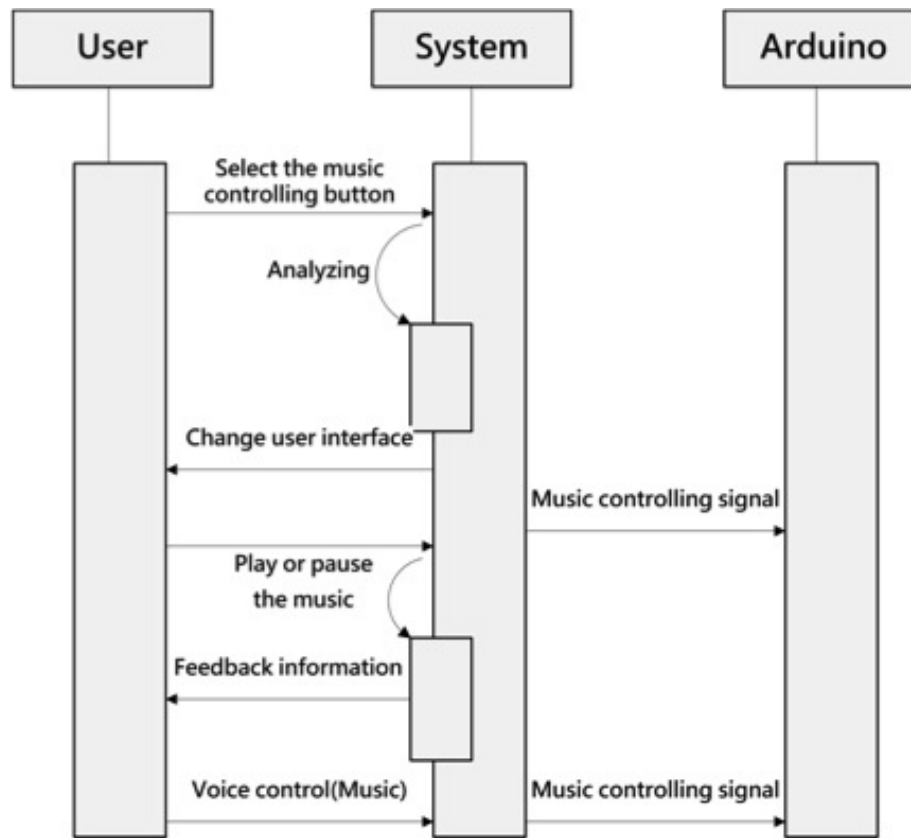


Figure 7: Music controlling sequence diagram UC-2.

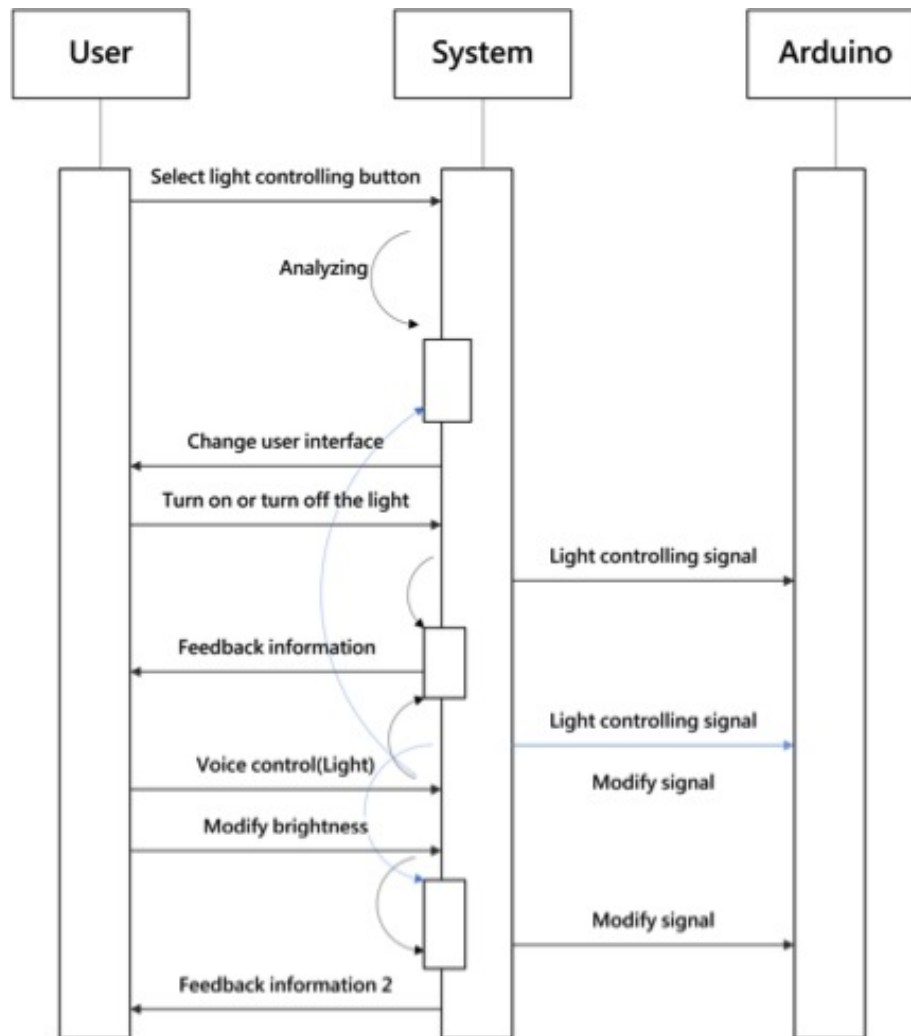


Figure 8: Modify the volume of music sequence diagram UC-4.

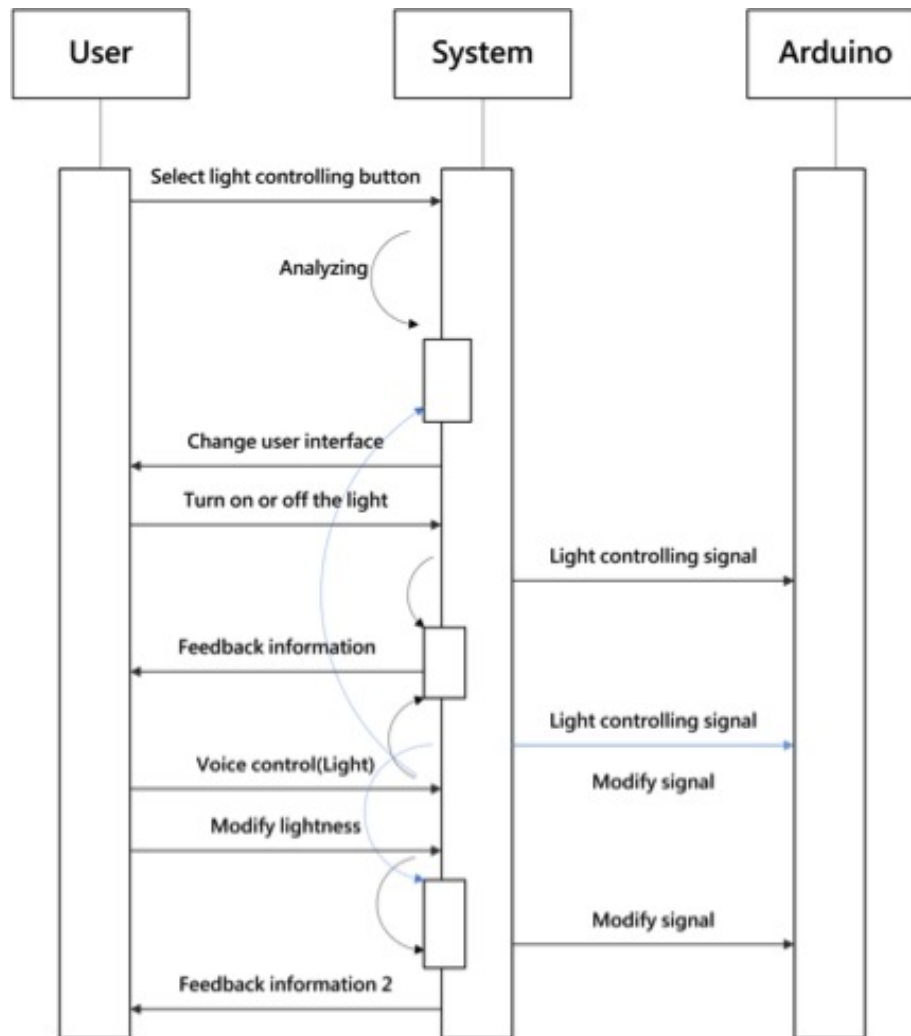


Figure 9: Modify the brightness sequence diagram UC-3.

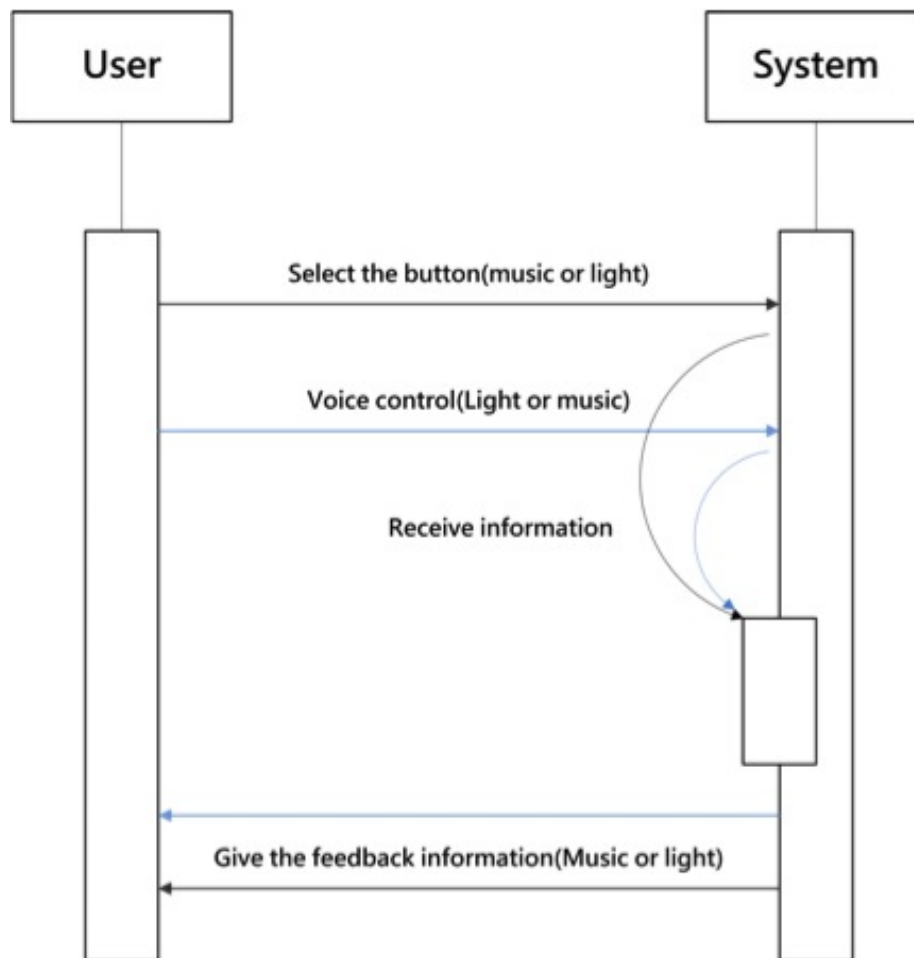


Figure 10: Create new users accounts sequence diagram UC-7.

The figures show some important use case system sequence diagram. In each diagram, we provide two methods to help user to control the system, the black arrow means the user completes the processes by button and the blue arrow means the users completes the processes by voice control.

5 User Interface Specification

5.1 Preliminary Design

From this section, we will give you a detail explanation about our interface design, and show you step by step that how users can run their house with minimized approach. We do effort to make our interface be friendly and clear.

Users can try input passport at most 3 times. If they type in wrong number three times succession, the screen would locked for about 5 minutes which is designed in order to protect users privacy.

5.1.1 Log In

When users enter our system, they are prompt with log in interface (shown in figure 11), which can guarantee their security, authorization and privacy. What the users need to do is inputting their password through the number keyboard showing on the desk. If they key in a wrong number accidentally, instead of keying in the entire password, they can modify it by clicking on the button at bottom-right to delete only the wrong one, which can minimize their effort effectively. Then users can tab the blue circle button which with an arrow on it to log in. When they fill in the right password, they will come to the main interface, otherwise, the interface will show users that the password is incorrectly, thus they can try it again (shown in figure 12). However, if they input wrong numbers more than 3 times, the system will be locked automatically in case of malevolent act of intruding.

The image shows a log-in interface on a dark background. At the top center is a circular icon of a person's silhouette. Below it are two input fields: the first is labeled 'user name:' and contains three asterisks '***'; the second is labeled 'password:' and contains seven asterisks '*****'. To the right of these fields is a blue circular button with a white right-pointing arrow. Below the input fields is a numeric keypad with a 4x3 grid of buttons. The buttons are arranged as follows:

1	2	3
4	5	6
7	8	9
exit	0	<

Figure 11: Log in interface; users can enter the main interface after keying in correct passport.

The image shows a user login interface. At the top, there is a circular icon of a person's head and shoulders. Below this, there are two input fields: "user name:" followed by three asterisks (***) and "password:" followed by seven asterisks (*****). To the right of these fields is a blue circular button with a white right-pointing arrow. Below the input fields, a red error message reads "incorrect input, please try again...". At the bottom of the interface is a numeric keypad with four rows of three buttons each. The buttons are labeled as follows:

1	2	3
4	5	6
7	8	9
exit	0	<

Figure 12: users cannot enter our system without authorization.

5.1.2 Main Interface

With the authorization, the user is presented with main interface (shown in figure 9-3), it means user can start control their house by voice now. On the top of the screen, he or she can see the detailed status of information of his/her specific devices. Knowing the status of devices, users can make command better and more efficiently. For example, if the status of light is on, they do not need to turn on it again. In other words, users can make better judgment when they know more.

Apart from the list on top, users can see that the main interface includes other 4 parts:

Microphone Microphone is the essential and indispensable part to a voice control device, when users want to output command, they just need to click on this button, then the circle background of microphone change its color from red to gray (shown in figure9-4). Meanwhile, the phone will start to record what they say, and then send it to Arduino over Wi-Fi. Users can be informed that their phones with that change are receiving their voice. When they finish the command, they need to click the Microphones button again, the circle background is back to red. Finally, it will receive the feedback and transfer it to users via voice system. (The rest three buttons are just needed when users cannot control by voice in some special situation. Otherwise, users can approach to their destination via voice. To sum up, the buttons shown on interface provide users with more options so that they can choose the way they prefer.)

Music One click on this button, they will come into the music interface, and then they can do more detailed things.

Light Through click on this button, the system would redirect users into the light interface.

Exit This button simply exits from system.

Our target is to make main interface as simple as possible in order to make users would not be bothered with too many useless option and messy buttons. Meanwhile, our interface is designed to satisfy the most principal functions.

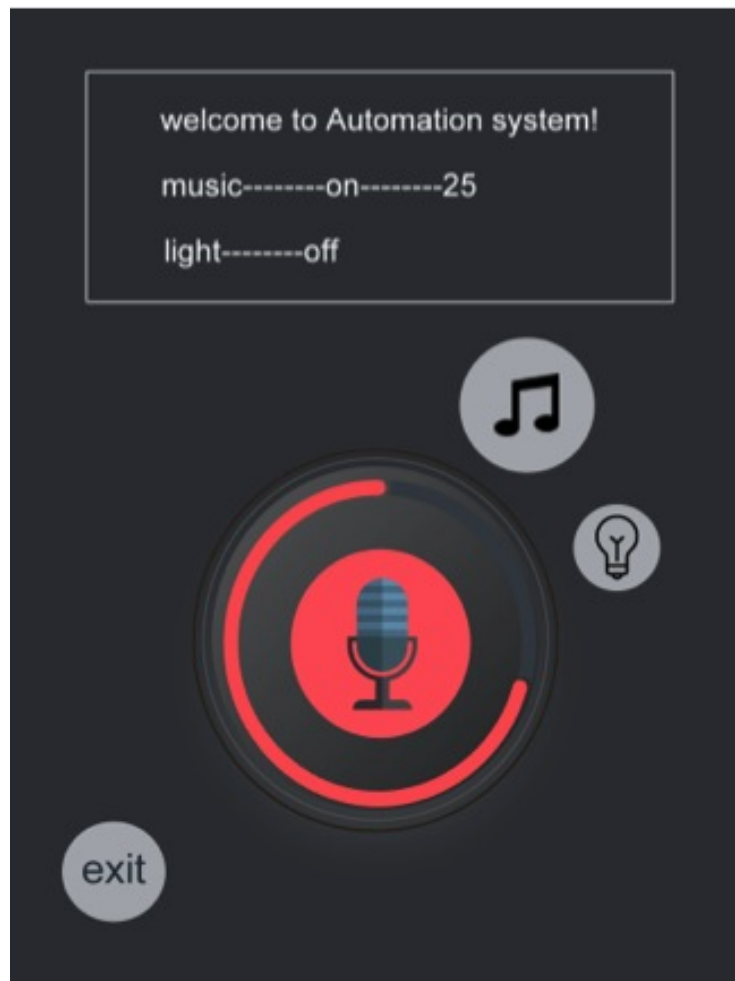


Figure 13: Main interface which users can click the Microphone button to make command.

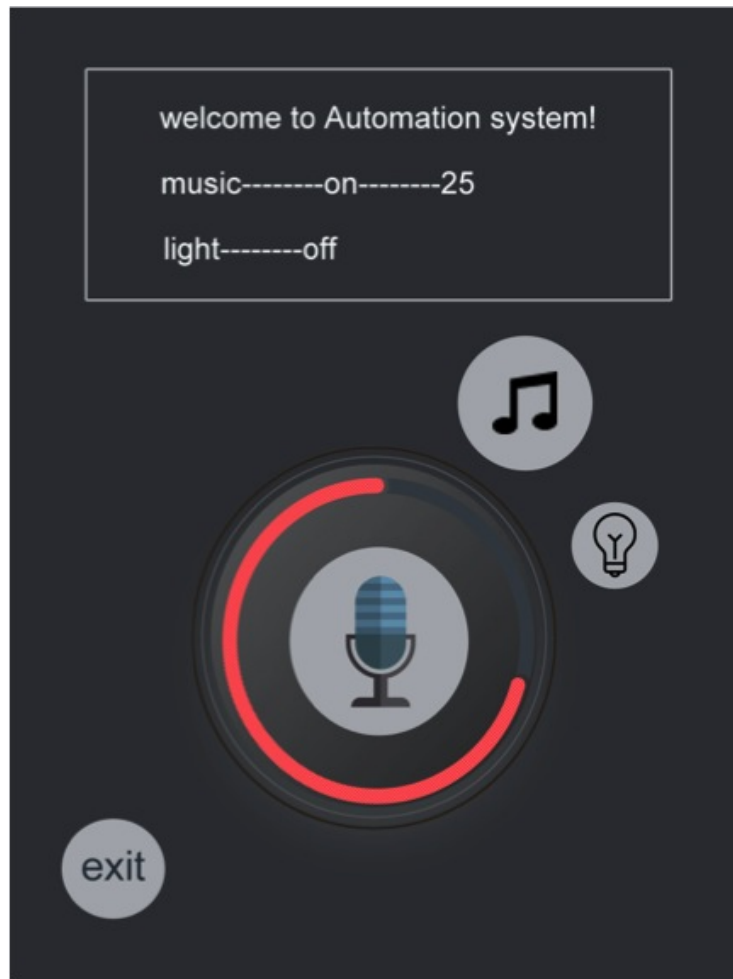


Figure 14: Main interface when record users voice.

5.1.3 Music Interface

When the mobile phone receive the command about the music, it will jumps into music interface. In this interface, users can command the music and check the status of music more clearly and more detailed. Users can see a button line, it includes play(shown in figure 9-5), pause(shown in figure 9-6), and volume (shown in figure 9-7 and figure 9-8) button, and the number shown under the button line is the degree of volume. If users want to change the status of music, they can click on the microphone button and speak, and then click the button again after finishing their order. If they want to go back to main interface, they can say back to main interface or just click the back button; they can choose what you want.

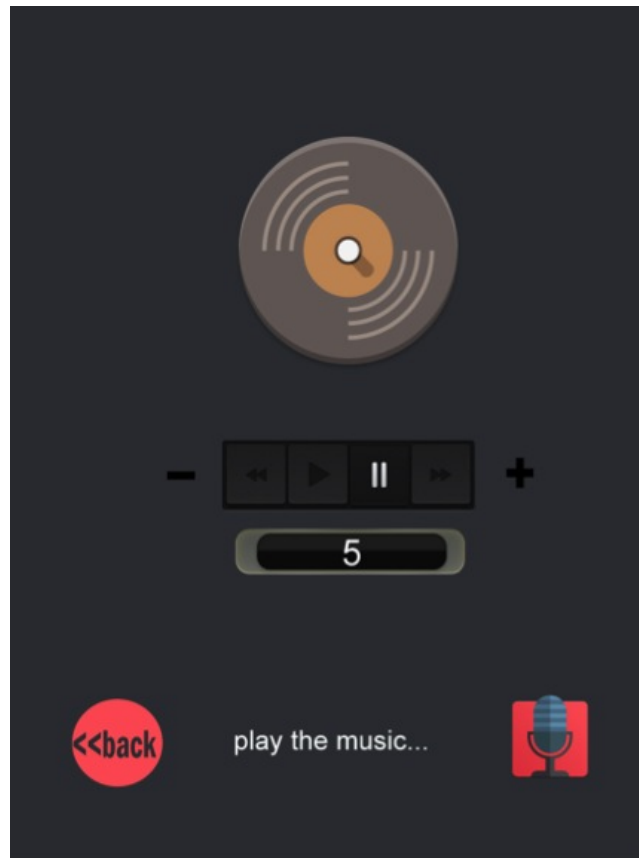


Figure 15: Music interface play the music.

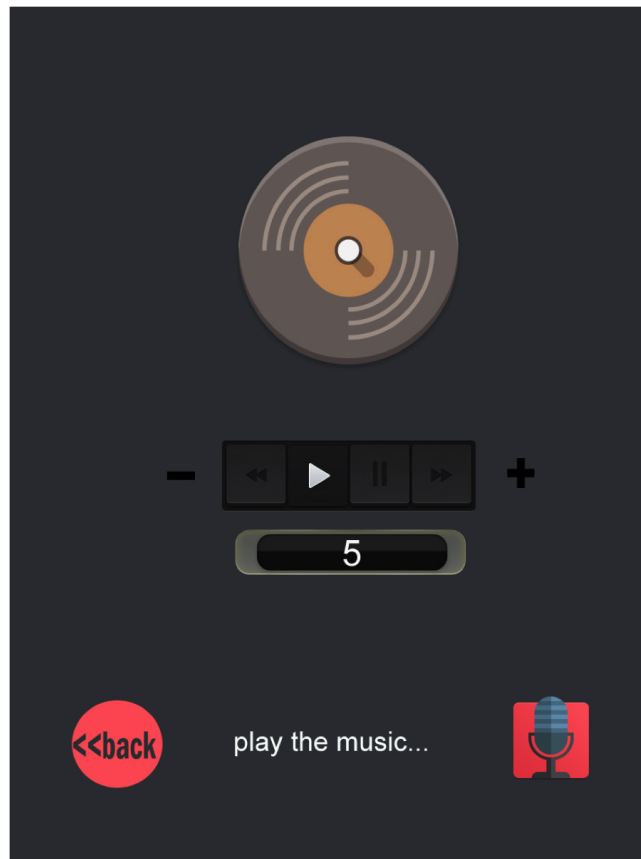


Figure 16: Music interface turn off the music.

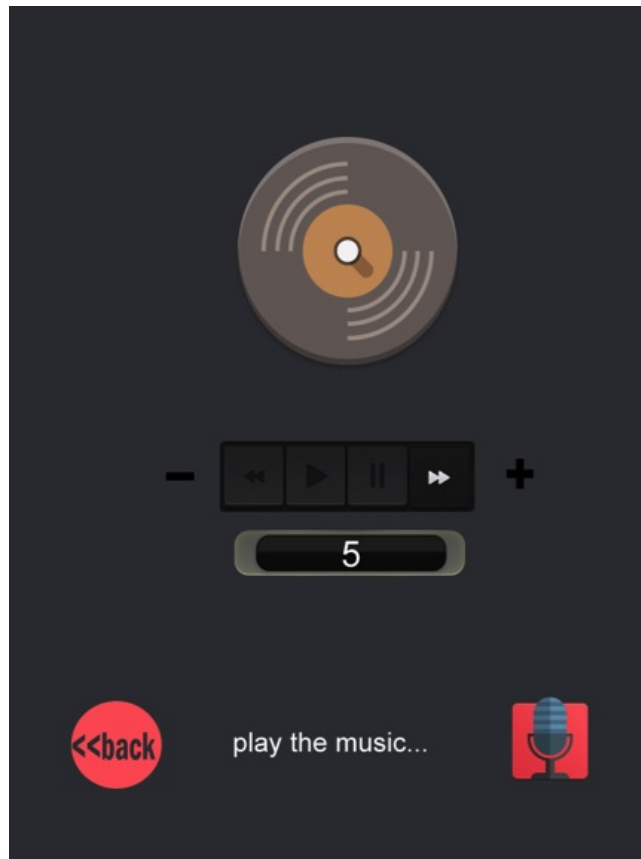


Figure 17: Turn up the music.

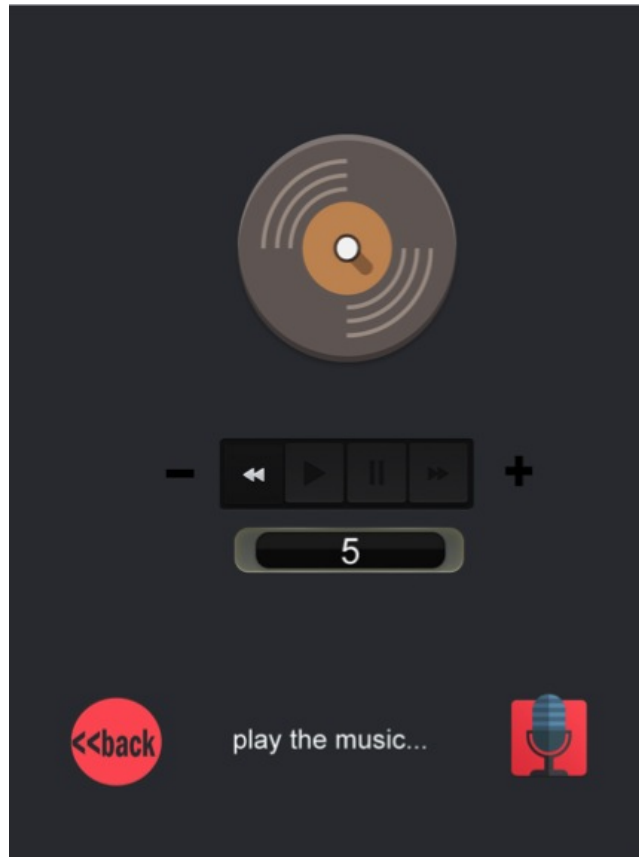


Figure 18: Turn down the music.

5.1.4 Light interface

After getting the signal about light, users are prompted with light interface. The picture of light can basically show whether the light is on or not, users can know the status of their lights simply through the dynamic picture or the button at the bottom of the screen. User can also control the brightness of the light through the button in the middle of screen, the number near the button can show the user the intensity of their light, we consider that the number can offer the user more direct and clear feelings which make them control their light more appropriately. Again, when the user wants to back to main interface, they can click the back button or back via voice.

Finally, as we can see from the whole interfaces, which are shown above, what user needs to do is very casual and simple. To log in, user needs type their

passport and click one button. To input their command, user needs to click the Microphone button two times. Generally, after these two steps, users can control their devices by voice all the time. As a result, our interface design can basically satisfy our requirement demand.

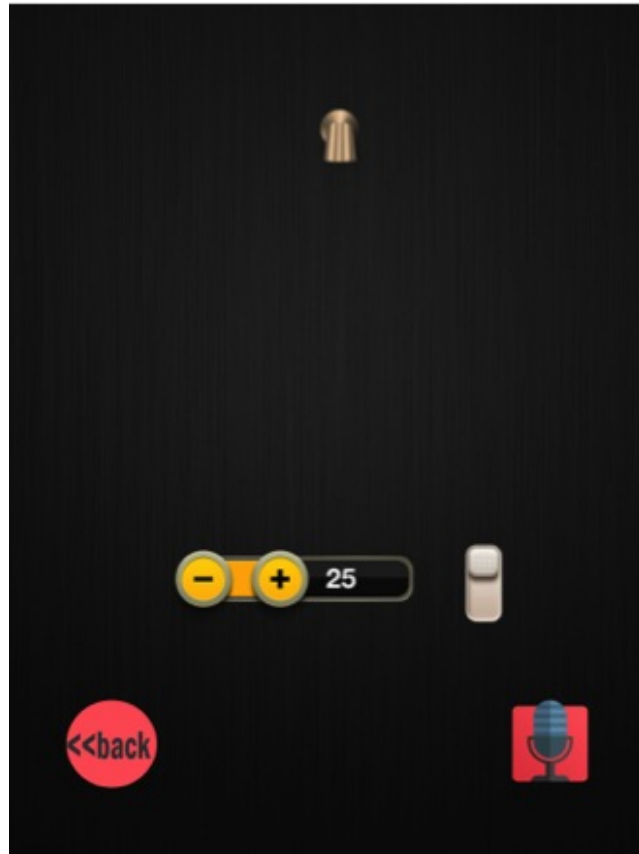


Figure 19: Turn up the music.

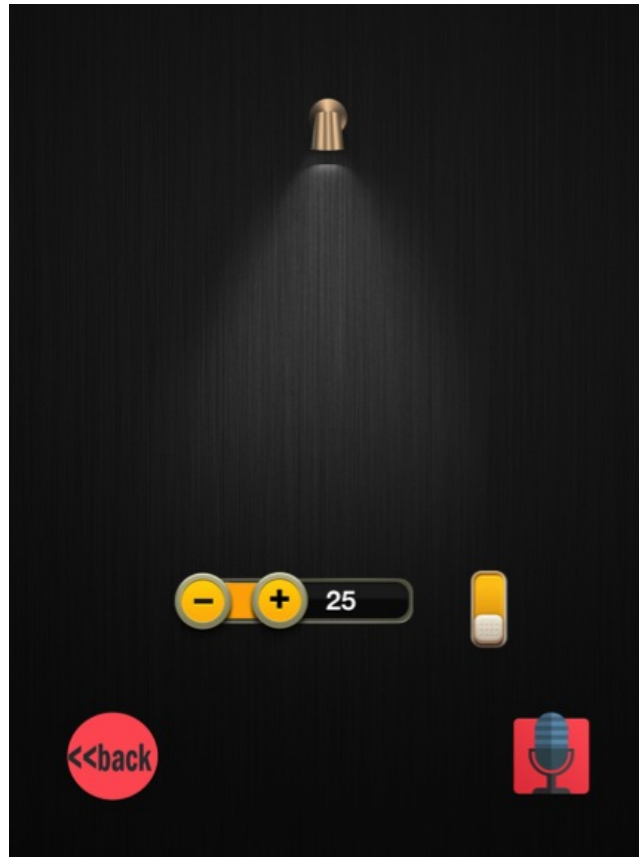


Figure 20: Turn down the music.

5.2 User Effort Estimation

5.2.1 User login

Respective user cases: UC-6

Navigation and Data Entry: 2 clicks and 10 20 keystrokes

1 Click auto-home to enter into the system.

2 Enter the username.

3 Enter the password.

4 Click the login button to check the username and password. If the word is correct, the user enable log in.

5 If the password is incorrect, user can enter the password again.

6 If the password is incorrect, user can click on return button to exit the system.

5.2.2 User logout

Respective user cases: UC-9

Navigation and Data Entry: 1 2 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes)

- 1 Click exit button to return to the login interface.
- 2 Click the return button to the cellphone main interface.
- 3 If the users do not want to return the login interface, click on return button to logout directly.

5.2.3 Light controlling

Respective user cases: UC-1

Navigation and Data Entry: 1 3 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes)

- 1 Click the light figure button to enter the light-controlling interface.
- 2 Push down the light button to turn on the light.
- 3 Push on the light button to turn off the light.
- 4 User enables use voice to control the light overall process click on the click on the microphone graphic and then use the voice to turn on or turn off the light.

5.2.4 Music controlling

Respective user cases: UC-2

Navigation and Data Entry: 1 3 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes)

- 1 Click the music figure button to enter the music-controlling interface.
- 2 Click the music play button to turn on the music.
- 3 Click pause to pause the music.
- 4 User enable use voice to control the music overall process click on the click on the microphone graphic and then use the voice to play or pause the music.

5.2.5 Modify the brightness

Respective user cases: UC-3

Navigation and Data Entry: 1 3 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes); 3. Light controlling - the light should be turn on (0 3 clicks)

- 1 Press on + and then drag the button to modify the brightness
- 2 User enable use voice to modify the brightness overall process click the micro-

phone graphic and then use voice to modify the brightness



5.2.6 Modify the volume of music

Respective user cases: UC-4

Navigation and Data Entry: 1 3 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes);

4. Music controlling (0 3 clicks)

1 Click  and  to modify the volume of the music

2 User uses voice to modify the volume of the music click the microphone graphic and then use voice to modify the volume of the music.

5.2.7 Elements status check

Respective user cases: UC-5

Navigation and Data Entry: 0 clicks

Prerequisite: 1. User login (3 clicks and 8 keystrokes)

1 When the use login the system the music and light status will show on the screen, user enable check these information directly.

5.2.8 Control information feedback

Respective user cases: UC-12

Navigation and Data Entry:

Prerequisite: 1. User login (3 clicks and 8 keystrokes)

This section will be divided into several parts to discuss

1 User uses button to control the system The feedback information will show on the screen directly, hence the user effort estimation depend on what operation the user chooses refer to the 1, 2, 3, 4, 5, 6

2 User uses voice to control system.

1: If the voice control fails, the feedback information will show on the screen directly.

2: If the voice controls successes, the feedback information will show on the next screen directly.

5.2.9 Light interface

Respective user cases: UC-12

Navigation and Data Entry: 2 clicks and 50 keystrokes

- 1 Click sign in button to enter the create new accounts interface
- 2 Enter the new username
- 3 Enter the new password
- 4 Enter the new email address
- 5 Click finish button to complete the sign in process.

6 Domain Analysis

6.1 Domain Model

Voice based home motivation has many parts. Devices controlled or monitored may include electronic ovens, motion sensors, music players. Most, if not all, of the useful sensors that this project plans to utilize will send or process a single data value. This makes it very convenient to lay out a generic way to process the inputs from, or outputs to these devices. Each device can then be reduced to an object bearing a single data value to report to the system, or an object receiving a single data value from the system to govern the device's actions, or both. These standardized objects will be the device nodes as shown in the diagram. The data values obtained from these devices can then be categorized into analog, digital, or none (if the device hasn't a value to report, or cannot be controlled). We give our solution in three major fields, the application in the android platform, the server and related algorithm, the intelligent elements at home and the communication through different network

6.1.1 Domain model-General

The general domain model is showed in figure 21, also the object lifelines is showed in figure 22.

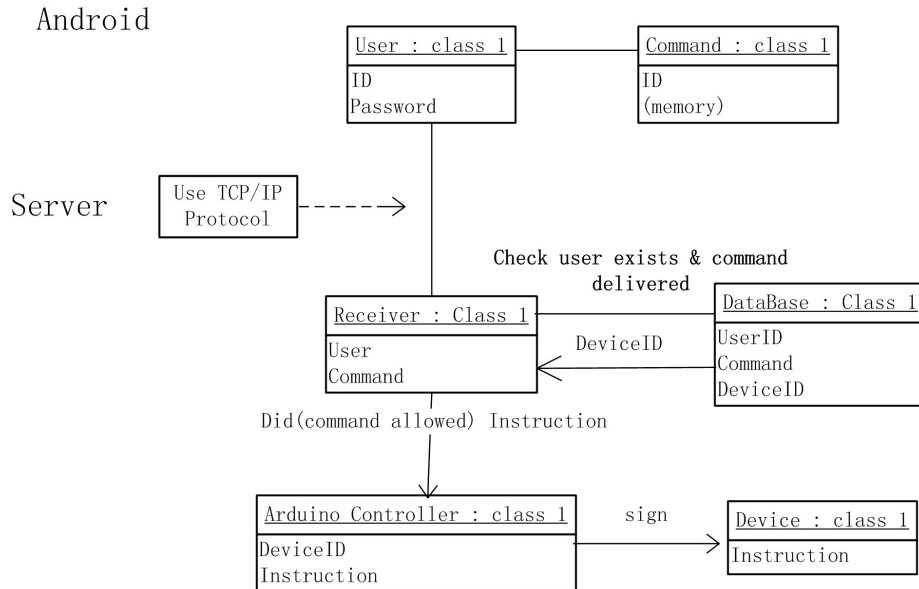


Figure 21: General domain model.

We have 3 layers. One is the Android layer which stores the information the user sent in and pass requirement of the user or the Admin to the server through the Internet. And the second layer, the Server layer, which receives the requirement the Android layer gets, and compute it to signals easy to understand by the Arduino Module by Wi-Fi or Internet. The last layer, which constructed with an Arduino Module and a MCU, is for receiving and control the electronic devices in home, the music player and the lights,etc.

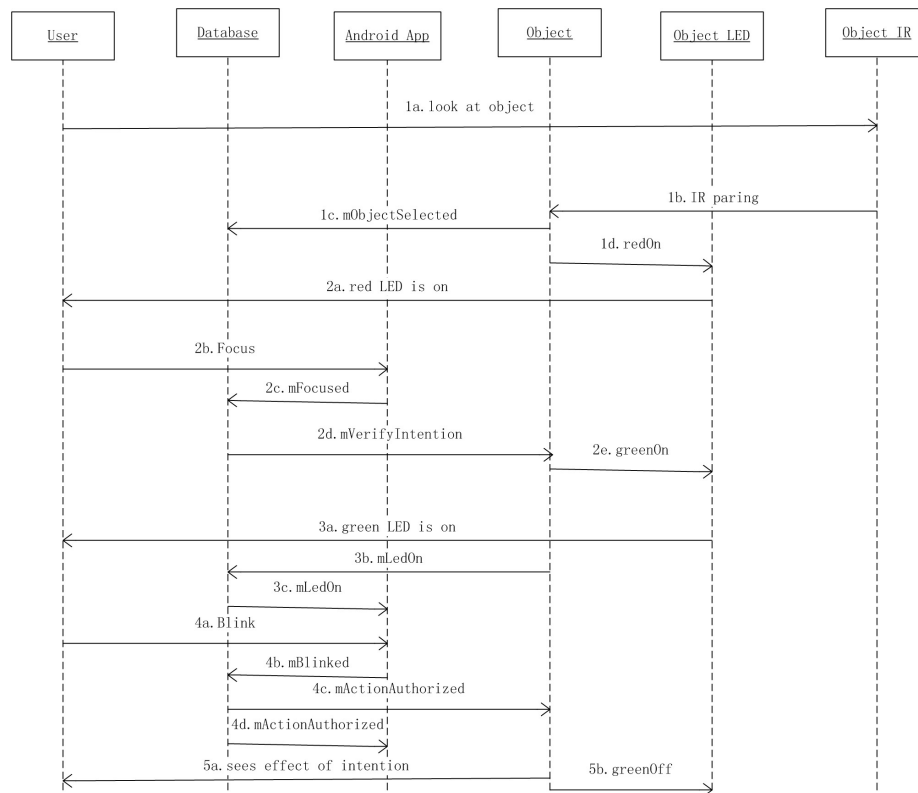


Figure 22: Object lifelines.

This is the object lifeline of the whole project, which illustrate how the user controlled the electronic devices in home by the Voice Based Home Automation.

6.1.2 Domain model-Server

The server is the brain of the whole home automation system. The cell phone or some other android platforms do not understand the users requirements and these devices just do some dictation works. The server gets the users words by communication with devices through networks; and then, the server analysis

the words and sentences people talk.

The analysis work can be extremely difficult. Our solution is starting with locate some key words in the sentences and try to figure out peoples requirements with those key words. Thats probably the natural way we use to understand each others talking.

We try to understand peoples requirements with some machine learn method. For example, if you want to turn on the light, you may say turn on the light, if you want turn off the light, the sentence should be turn off the light. We use a filter to pick out some top level keywords like Light, then we pick up some lower level key words like turn on, turn off, and then we can pick up some lowest level key words like 50 percent.

The server should also have the ability to communicate with the intelligent elements at home and the devices. When the server complete the analysis, the server should send some commands to the Smart elements to control them work in right way. Meanwhile the server should also send some feedback to the user to remind them the requirements have been accomplished.

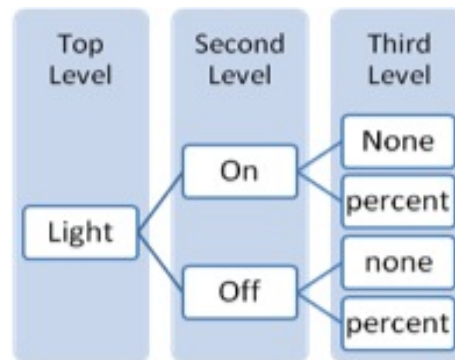


Figure 23: Server Model.

6.1.3 Domain model-Arduino

The android platform should have ability to communicate with server via LTE network or Wi-Fi. Anyway, the communication should base on the Socket protocol. Socket communication is widely used nearly everywhere and Google also gives the developer some free APIs to use.

After the command given by the user is sent to the server, in the server it is required to generate a specific code for each command. A serial code is generated

and communicated with the Arduino over the Wi-Fi.

For example:

If the code is 001501-02-01-1000-150111

We can split it up as:

001501: start of code

02: user code (for e.g. 02- Vidur, 03- Chang)

01: device code (for e.g. 01- light, 02- fan)

1000: action code (for e.g. 1000- ON, 1001- OFF, 1010- status)

150111: end of code

The Arduino will read this serial code and relay it to the MCU over the Infra-red bandwidth to perform the desired action. This process can be broken up into smaller modules to make it easier for the reader to understand.

The First model can be considered the work done by the Arduino. It is required to do the following jobs:

- Receive the signal sent by the server over the Wi-Fi.
- Decode the received signal and recognize what work and by which device is required to be done.
- Generate a binary code to be transmitted by the required infra - Red transmitter towards the device on the job needs to be performed (For Example a light which needs to be turned ON).
- Wait for the feedback from the Micro Controller Unit (MCU) regarding the status of the device on which the job needs to be done.
- Generate an alarm and reset the MCU, in case no feedback is received in the required amount of time.
- In case the feedback is received, send a confirmation signal to the MCU and relay the status of the MCU to the server.
- In case if the state of the device is changed manually, send the status to the server through Wi-Fi.
- If authentication of the user fails a number of times, Lockdown the whole house and sound the alarm.

The Second model can be considered the work done by the Micro Controller Unit (MCU). It is required to do the following jobs:

- Receive the signal sent by the Arduino over the Infrared bandwidth.
- Check the state of the device it is connected to.
- In case the device is already in the state, in which the user wants it to be, send a feedback to the Arduino communicating the same, so that the server can tell the user that the device is already in the required state.(For example if the user wants to turn ON the light and the light is already in the ON state.)
- In case the device is not in the state, in which the user wants it to be in. Then change the state of the device through the electronic circuit and send a feedback to the Arduino regarding the same.
- Keep on sending the feedback code until it receives a confirmatory signal from

the Arduino.

-In case if the state of the device is changed manually, send the status to the Arduino.

The arduino is based on MCU and with that micro-controller we can use AD/DA to control the luminance of the light, we can play or stop the music. With thousands of open source projects in the Internet, we can do a lot of things with MCU based elements at home.

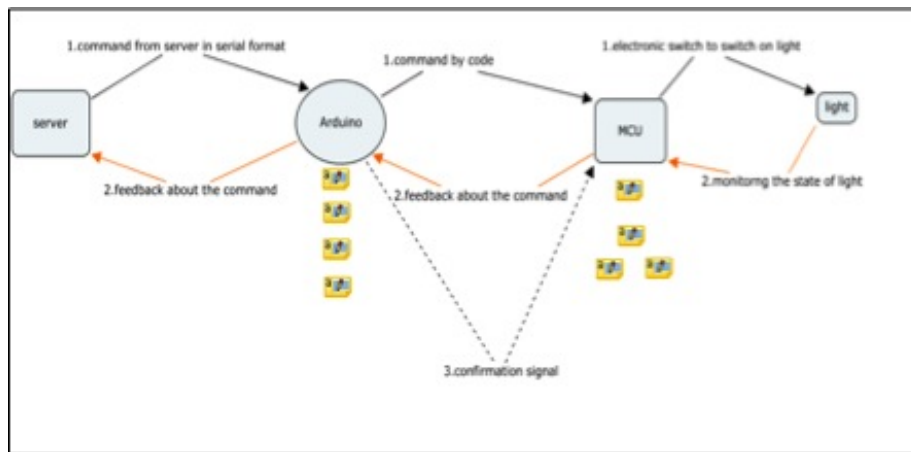


Figure 24: General description diagram.

6.1.4 Domain Model-The communication protocol

The communication is the chain to connect the server the devices and the smart elements at home together as a system. It is a challenge for us to build a reliable communication between different devices and different networks.

However, though devices and network changes, the IP address and the mac address stay the same (in general cases). The communication based on Socket is suitable for our apply situation. However we still need to build a communication protocol in application layer. The protocol should include some key elements like the time, the event, the key command, and some labels to label the state like in processing, done and failed.

6.2 System Operation Contracts

6.2.1 Light controlling

Preconditions:

The Light Smart Element has an Internet connection

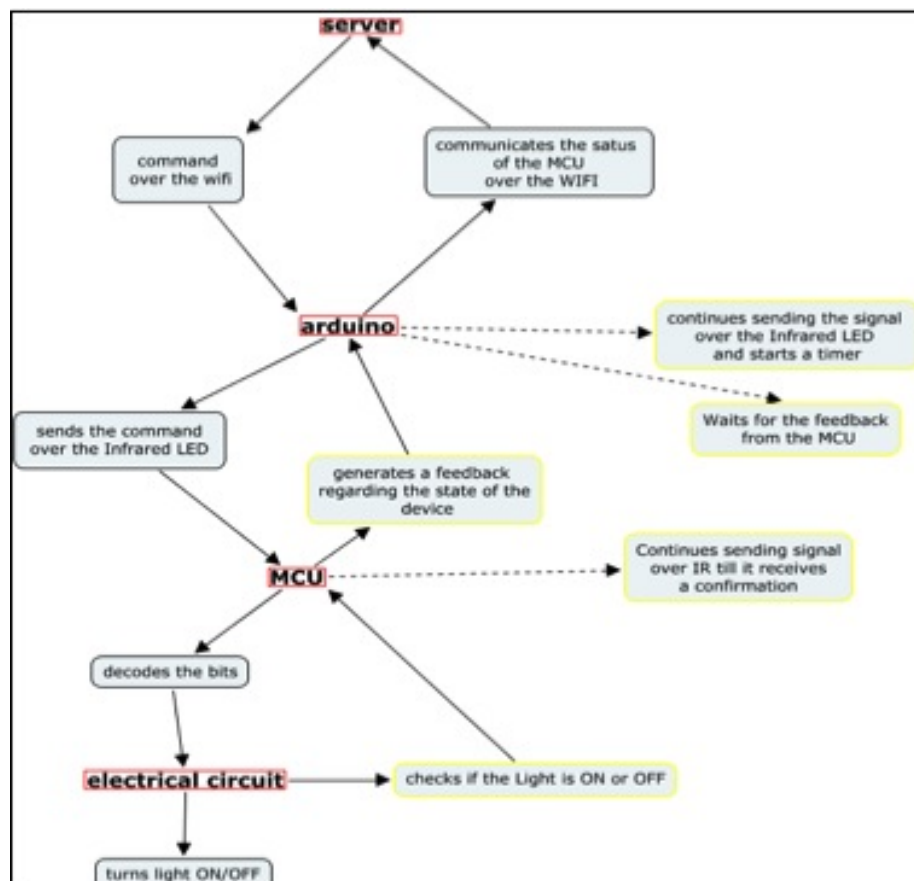


Figure 25: The Camp for the two modules.

The Light Smart Element can update it condition

Post conditions:

None.

6.2.2 Music controlling

Preconditions:

The Music Smart Element has an Internet connection

The Music Smart Element can update it condition

Post conditions:

The music Smart element can send updates of volume and music information

The Music Smart element can play music without Internet connection.

6.2.3 Modify the brightness

Preconditions:

The Light Smart Element has an Internet connection

The Light Smart Element has DAC to change the voltage

Post conditions:

None

6.2.4 Modify the volume of music

Preconditions:

The Music Smart Element has an Internet connection

The Music Smart Element can update it condition

The Music Smart Element has DAC to change the voltage

Post conditions:

The music Smart element can send updates of volume and music information

The Music Smart element can play music without Internet connection.

6.2.5 Elements status check

Preconditions:

The Smart Element has an Internet connection

The Smart Element can update it condition

Post conditions:

The Smart element can notify the changes when status changes.

6.2.6 User login

Preconditions:

The User has an account

The Server has the users information

Post conditions:

The server can record the users login history.

6.2.7 Create new users accounts

Preconditions:

The User does not have an account

The Server does not have the users information

Post conditions:

The server can record the users login history.

6.2.8 Remote user

Preconditions:

The User has an account

Post conditions:

None

6.2.9 User log out

Preconditions:

The User does not have an account

The User is logged in

Post conditions:

The server can record the users log out history.

6.2.10 Check the authority of user

Preconditions:

The User does not have an account

The Server have the users information

Post conditions:

The server can update the users information via Internet.

6.2.11 Network error notification

Preconditions: The Smart element has Internet connection

The server has Internet connection

The smart Elements at home have Internet connection

Post conditions:

The system is able to rebuild the connection of the whole system.

6.3 Mathematical Model

In this section we are going to talk about the mathematical models in our system.

6.3.1 The fuzzy recognition model

Though we use Google's voice dictation APIs to do the voice recognition work, and Google did a good job in voice recognition, the recognition results are still affected by multiple factors. The noise in the environment, the accent and even the talking speed can have some bad influence on the accuracy of recognition.

However, we want to build a system that even the server only recognize a part of the command the system is still able to give user some feedbacks and options.

In the natural command sentences there are usually 4 types of words: nouns, verbs, adjectives and some other words. By detecting these words we can know or partly know the user's idea. For example if a user wants to turn on the light he will say, Turn on the light. Here Turn on is the verb and it tells us the action need to be done. Light here is the noun, which shows us the subject the user wants to control.

In our server algorithm, we classify the input command in 4 levels: Subject, actions, degrees, and other information. The server first segments the every single words in the command and then put the words in different levels by cross compare the words in different word pools.

Then we can know exactly what is the user's requirement. If the recognition of some part of the command is failed we can still provide user some action options by the information from other level. The level model and description is showed in table 20.

Level	Description	Addition
1	Nouns	If missing, system provide some information about the types of Smart Elements based on Level 2 words.
2	Verbs	If missing, System give some optional actions based on level 1 words.
3	Words about degrees	If missing the system do the action without degree information and send user some feedbacks.
4	Others	No use.

Table 20: Different levels of words.

7 Project Management

7.1 Plan of Work

In this section we are going to talk about our tasks and plans. We have two kinds of jobs in general, the system design related jobs and paper works.

We treat each week as a building circle so that our plan circle is a week. Table 8 shows the works and the related due time. Blue part means the tasks has already been finished; Green part represents the jobs we are working on; the yellow parts mean the tasks will be done in the future.

Task	SEP21	OCT4	OCT11	OCT18	OCT25	NOV1	NOV8	NOV15	NOV22	NOV29	DEC6	DEC13
Website Maintenance												
App UI Design												
App Frame Design												
Server Frame Design												
Smart Element Design												
Communication Design												
System De-Bugging I												
Problem Fix												
System De-Bugging II												
Problem Fix												
Report 1												
Report 2												
Report 3												
Demo 1												
Demo 2												
Archive Documentation												

7.2 Current problems and possible solutions

In this section we are going to talk about our current job some major problems we are facing.

Currently we are working mainly on the UI design and the system logic design. A system like this is a very complicated system; multiple parts are involved in a single command. So far, we have successfully designed the UI and we are now working on the Smart Element design, Server algorithm design and Android based application design. However, while doing our job there is some serious problems we are facing and we are try to get over them.

The most important purpose of our project is that we want to make our system based on voice control. However, voice recognition and processing is an extremely difficult job. Though we use some APIs provided by Google and Android, we still have to provide some original voice information to the APIs to accomplish the dictation work. But the environment do have a great influence

on our system, some times in the noisy environment, the system can hardly recognize any single word. So may be a kind of filter should be added to the system to control the influence from noise.

Furthermore, because we know that even the best the voice recognition system cannot fully understand every command. People can be confused if the system cannot understand their commands for some times. Focus on such situation, we are try to build a Server algorithm that even the system only recognize a part of the users command, it would still be able to give user some control feedbacks or control suggestions.

Also we have some concerns about the Smart Elements at home. We want to make them energy saving which means those smart elements will shutdown themselves automatically at night and start up in the morning. However, such design may face the problem that what if people want to use the smart elements at night. We are still trying to find a better design logic to deal with it.

7.3 Next steps

In the future we will focus on two type of things based on our home automation system.

One type of things is that we will try to build as many kinds of Smart Elements as possible. Different Elements can do different jobs as home and all the Smart Elements share a same communication protocol and design logic, which mean all the new and old Smart Elements can working together in a same system environment. We can add a bunch of new features into our systems like control your washing machine with your voice or control your coffee machine or TV with voice commands.

Also we would like to bring some new control method to our system. Voice control is a really good method to communicate with system, but there are some situations people cannot talk to their cellphones or people are not willing to talk to their cellphones. In this case we would like to build a pattern based control system. In general people can control the Smart elements at home b drawing some patterns in the cellphones.

8 Reference

- [1]**ANDROID (OS)**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/ANDROID_\(OPERATING_SYSTEM\)](http://en.wikipedia.org/wiki/Android_(operating_system))
- [2]**FUNCTIONAL REQUIREMENT**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/FUNCTIONAL_REQUIREMENT](http://en.wikipedia.org/wiki/Functional_requirement)
- [3]**USER STORIES**. [HTTP://WWW.ECE.RUTGERS.EDU/~MARSIC/BOOKS/SE/](http://www.ece.rutgers.edu/~marsic/books/se/)
- [4]**ISO**. [HTTPS://DEVELOPER.APPLE.COM](https://developer.apple.com)
- [5]**ANDROID APPLICATION PLATFORM**. [WWW.MANTRAIS.COM/MOBILE/ANDROID-APPLICATION-DEVELOPMENT/](http://www.mantrais.com/mobile/android-application-development/)
- [6]**VOICE COMMAND DEVICE**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/VOICE_COMMAND_DEVICE](http://en.wikipedia.org/wiki/Voice_command_device)
- [7]**NON-FUNCTIONAL REQUIREMENT**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/NON-FUNCTIONAL_REQUIREMENT](http://en.wikipedia.org/wiki/Non-functional_requirement)
- [8]**FURPS**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/FURPS](http://en.wikipedia.org/wiki/Furps)
- [9]**NON-FUNCTIONAL REQUIREMENT**. [HTTP://WWW.MOUNTANGOATSOFTWARE.COM/BLOG/NON-FUNCTIONAL-REQUIREMENTS-AS-USER-STORIES](http://www.mountangoatssoftware.com/blog/non-functional-requirements-as-user-stories)
- [10]**HOUSING**. [HTTP://MONEYNING.COM/HOUSING/THE-FIVE-YEAR-RULE-FOR-BUYING-A-HOUSE/](http://moneyning.com/housing/the-five-year-rule-for-buying-a-house/)
- [11]**UI-OVERVIEW**. [HTTP://DEVELOPER.ANDROID.COM/DESIGN/GET-STARTED/UI-OVERVIEW.HTML](http://developer.android.com/design/get-started/ui-overview.html)
- [12]**ANDROID**. [HTTP://WWW.ANDROID.COM/ABOUT/](http://www.android.com/about/)
- [13]**MOBILEPHONE UI DESIGN**. [HTTPS://WWW.GOOGLE.COM.HK/SEARCH?Q=MOBILE+UI+DESIGN&SAFE=STRICT&SA=X&TBM=ISCH&TBO=U&SOURCE=UNIV&EI=30LMUOSCLIHQ8QTBNOG4AG&VED=OCduQSAQ&BIW=1280&BIH=650&DPR=1](https://www.google.com.hk/search?q=mobile+ui+design&safe=strict&sa=x&tbm=isch&tbo=u&source=univ&ei=30LMUOSCLIHQ8QTBNOG4AG&ved=OCduQSAQ&biw=1280&bih=650&dpr=1)
- [14]**MOBILE-PATTERNS**. [HTTP://WWW.MOBILE-PATTERNS.COM/](http://www.mobile-patterns.com/)
- [15]**MOBILE APP DESIGN**. [HTTP://WWW.1STWEBDESIGNER.COM/DESIGN/MOBILE-APPS-DESIGNS/](http://www.1stwebdesigner.com/design/mobile-apps-designs/)
- [16]**GLADE**. [HTTPS://GLADE.GNOME.ORG/](https://glade.gnome.org/)
- [17]**HOME AUTOMATION AND CONTROL**. [HTTP://WWW.SAVANTSYSTEMS.COM/HOME_AUTOMATION_AND_CONTROL.ASPX](http://www.savantsystems.com/home_automation_and_control.aspx)
- [18]**SOFTWARE ENGINEERING**. [HTTP://WWW.ECE.RUTGERS.EDU/~MARSIC/BOOKS/SE/](http://www.ece.rutgers.edu/~marsic/books/se/)
- [19]**STAKEHOLDERS**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/STAKEHOLDERS](http://en.wikipedia.org/wiki/Stakeholders)
- [20]**USE CASE**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/USE_CASE#ACTORS](http://en.wikipedia.org/wiki/Use_case#actors)
- [21]**AUTOHOME**. [HTTP://ECEWEB1.RUTGERS.EDU/~MARSIC/BOOKS/SE/PROJECTS/AUTOHOME/2012-G1-REPORT3.PDF](http://eceweb1.rutgers.edu/~marsic/books/se/projects/autohome/2012-g1-report3.pdf)
- [22]**SE PROJECT REPORT3**. [HTTP://ECEWEB1.RUTGERS.EDU/~MARSIC/BOOKS/SE/PROJECTS/OTHER/2012-G4-REPORT3.PDF](http://eceweb1.rutgers.edu/~marsic/books/se/projects/other/2012-g4-report3.pdf)
- [23]**SOFTWARE ENGINEERING**. [HTTP://WWW.ECE.RUTGERS.EDU/~MARSIC/BOOKS/SE/](http://www.ece.rutgers.edu/~marsic/books/se/)
- [24]**SYSTEM SEQUENCE DIAGRAM**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/SYSTEM_SEQUENCE_DIAGRAM](http://en.wikipedia.org/wiki/System_sequence_diagram)
- [25]**SOFTWARE ENGINEERING**. [HTTP://WWW.ECE.RUTGERS.EDU/~MARSIC/BOOKS/SE/](http://www.ece.rutgers.edu/~marsic/books/se/)
- [26]**SE PROJECT REPORT3**. [HTTP://ECEWEB1.RUTGERS.EDU/~MARSIC/BOOKS/SE/PROJECTS/AUTOHOME/2012-G1-REPORT3.PDF](http://eceweb1.rutgers.edu/~marsic/books/se/projects/autohome/2012-g1-report3.pdf)

- [27]**SE PROJECT REPORT3**. [HTTP://ECEWEB1.RUTGERS.EDU/~MARSIC/BOOKS/SE/PROJECTS/OTHER/2012-G4-REPORT3.PDF](http://ECEWEB1.RUTGERS.EDU/~MARSIC/BOOKS/SE/PROJECTS/OTHER/2012-G4-REPORT3.PDF)
- [28]**EFFORT ESTIMATION**. [HTTP://EN.WIKIPEDIA.ORG/WIKI/SOFTWARE_DEVELOPMENT_EFFORT_ESTIMATION](http://EN.WIKIPEDIA.ORG/WIKI/SOFTWARE_DEVELOPMENT_EFFORT_ESTIMATION)