

Original Research Paper

Smart Mirror for Home Automation

Muhammad Syazwan Ab. Latif^{1*}, Ammar Asyraf Ismail¹, Asnazulfadhli Zariman¹

¹ Program of Computer Science, Department of Computing, Faculty of FSKIK, Universiti Pendidikan Sultan Idris, Malaysia.

Article History

Received:
11.07.2019

Revised:
23.08.2019

Accepted:
27.09.2019

***Corresponding Author:**
Muhammad Syazwan Ab Latif
Email:
e018828@siswa.upsi.edu.my

This is an open access article,
licensed under: [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/)



Abstract: As technology advances, we continue to find more and more uses for it that would previously be inconceivable. Originally, technology was primarily useful for performing tasks humans struggle with, but today it is used in even the most mundane tasks in an attempt to simplify our lives. With the technological revolution, we have been able to save time in a number of ways; however, as media consumption has increased, we also lose time. Due to this, saving time in our daily routines is always helpful. One way technology has been implemented to save time is by integrating computers into numerous elements in our home, thus creating “Smart Home” devices. The “Smart Mirror” research is based upon this concept. The objective of the research is to develop a system for Smart Mirror, evaluate Smart Mirror system functionality and facilitate and save time of users and employers. Methodology of this research are analysis, design, implementation, test, and renewal. Smart Mirrors are interactive devices that helps everyone check updates easily with voice control. People can do things at the same time in the morning like brushing teeth while catching up on the latest stocks updates, weather or traffic. There is a weakness in Smart Mirror that have to be fixed in the future. One of the weakness is the Smart Mirror cannot run if do not have any connection from the internet. Smart Mirror only can get inform if that have connection to the internet such weather, news and so on..

Keyword: Home Automation, IoT, Magic Mirror, Raspberry Pi, Smart Mirror.



1. Introduction

As technology advances, we continue to find more and more uses for it that would previously be inconceivable. Originally, technology was primarily useful for performing tasks humans struggle with, but today it is used in even the most mundane tasks in an attempt to simplify our lives [1]. With the technological revolution, we have been able to save time in a number of ways; however, as media consumption has increased, we also lose time. Due to this, saving time in our daily routines is always helpful. One way technology has been implemented to save time is by integrating computers into numerous elements in our home, thus creating “Smart Home” devices. The “Smart Mirror” research is based upon this concept.

The Smart Mirror will merge technology with a mirror to provide users information while they use their mirror. The primary motivation behind the smart mirror is to improve quality of life. Providing information to users in the most convenient way possible is a driving motivation behind the majority of technological development for smartphones and tablets. The smart mirror will provide convenient information to users on their mirror every day. Allowing the user to multitask by consuming media while preparing for the day will save people time nationwide. The goal of the mirror is to provide people with information they may require in the morning while getting ready for the day or at night before going to bed. This will save users time every day and help to ensure they are aware of important details for their day. A user will be able to check their calendar for any upcoming events, peek at the weather forecast, and not to mention, consult the mirror for traditional personal appearance adjustments.

Motivation for this research stems from multiple sources. In the Iron Man films, the main character utilizes holographic displays around the home to perform a number of activities. A couple years back, Corning released a video about their product called Glass which is intended to allow a smart surface anywhere in the home. While these examples, and a multitude of others, are well beyond the scope of this mirror, their realization also seems to be well into the future. One benefit to the smart mirror is that, while it does not provide the advanced capabilities of these examples, it is readily feasible. Another driving factor in this research is the fact that smart home technology has been developed for many parts of the home but smart mirrors are lacking. While there are plenty of tinkerers research s posted around the web, no fully realized implementation has been marketed to users thus far.

2. Methodology

2.1. Analysis

The previous scenarios have been focused on use cases that are convenient and directly useful for the user. However the mirror could also play a more indirect role. Such as, peoples receive all the information need to start their day, like weather information/forecast, emails, calendar, estimated traffic to their destination and endless possibilities. Most people uses a mirror at least twice a day: morning and evening.

2.2. Requirement Analysis and Specification

Presentation and punctuality are two of the most valued qualities in modern society. However, it can be difficult to effectively prepare for the day while remaining knowledgeable about current affairs and still maintain a timely schedule. In the morning, it is imperative to prepare for the day in front of a mirror, which is often a slow process. Additionally, factors such as the current weather conditions can influence how a person prepares for the day. Finding an efficient way to check all the factors that can affect how a person prepares for the day while also not adversely affecting the tasks that are performed in front of a mirror can be a challenge.

The goal of our project was to create a product that will provide quick and easy access to the time, news, and weather while simultaneously allowing a person to go through their morning routine. Our product should enhance productivity while providing a functional and enjoyable user experience.

2.2.1. Constraints

2.2.2.1. Extensibility

Constarin interm of the extensibility, are:

- 1) Embedded Computer

Since the embedded computer operates on open-source Linux, a large range of programs can be compiled and run on the device. The operating system and powerful CPU and memory allow the embedded computer to perform a wide range of tasks [2].

2) APIs

Each widget was designed to be modular and function independently of other widgets or other Smart Mirror components. This design allows the code from any widget to be used by developers in other websites or web-based applications if they have an API key, where applicable. Additional APIs may be implemented later, or additional functionality may be implemented with the selected APIs.

2.2.2.2. Manufacturability

The reflective film surface introduces more labor costs than other reflective materials. If the Smart Mirror were commercialized the labor process for installing the film would increase the manufacturing time and cost.

2.2.2.3. Reliability

Constarin interm of the reliability, are:

1) Embedded Computer

The overall reliability of the embedded computer is dependent on how resource intensive applications hosted on the system are. The system may experience slower operation if the computer is left on for long durations without a reboot. This could be mitigated by having a scheduled maintenance script.

2) APIs

APIs rely on support from third-party companies, which is somewhat unreliable because support may change or be dropped at any time. If the third-party company servers experience downtime, the widget for that service will not work.

3) Reflective Surface

With the film installed, it may be prone to scratching, which could reduce reliability; however, the film is easy to replace if damage occurs.

2.2.2.4. Economic Context

Constarin interm of the economic context, are:

1) Embedded Computer

Having a cheaper embedded computer would also affect the overall cost of the product, which would influence sales. Minimizing embedded computer costs while still meeting computing power requirements would reduce the system cost.

2) Reflective Surface

The material from which the reflective surface is made can easily affect the overall cost of the system. If a more expensive higher-quality reflective surface was used, the product might be regarded as a high-end luxury product that would not be sold very frequently.

2.2.2. Rational

Rationale for Smart Mirror it is expected that this research will be used as a partner assistant who solves easy problems and can organize daily schedule.

2.2.3. Specification Details

The mirror will be developed with specifications based upon devices people use every day; tablets, smartphones, and PCs. The software implementation specifically will be implemented in a way where specifications for future hardware implementations can be as flexible as possible. However, during the early development phase, strict specifications will be set to ensure all goals and objectives mentioned previously are met.

2.3. Design

The second stage is the Design process. The first and most important of our design phase was to decide the dimensions of our mirror. This ended up being determined by the monitor we chose for our display. Since we ended up using a Samsung 20" monitor, we needed a mirror of a size such that both

the display elements and reflection would not be competing for space. We quickly settled on a 20"x30" dimension as the best-looking and most functional size. Next, it became necessary to design a frame to contain both the mirror and the suite of electronics behind it. A wood frame being standard for most mirrors, we looked to this material for a design solution. We realized that due to weight and bulk of the mirror's contents we'd need sturdy lumber, so while browsing Home Depot we settled on using a 2x3 construction. Screwed together in a box shape, it offers the strength and depth to contain our materials. These consist of the visual applications that we choose to form our display. From the research inception we knew that, at minimum, we wanted to be able to display the date and time. However, we also thought it would be both functional and visually appealing to have the day's weather forecast shown as well. It was the combination of these three that would ultimately make up our design. Deliverable: Date, time, and weather display widgets.

2.4. System Design

Our user interface design will be simple and minimalistic, displaying the various bits of information in an elegant fashion. We will utilize the screen real estate to display the information in an optimal orientation that does not hinder the function of the mirror, such as in the top corners of the display. The font, text size, and color should be appropriate for its intended use location, accounting for the lighting conditions of the room in which it operates. For aesthetic purposes, a monochromatic theme is desirable but could become difficult to read in bright environments.

The general display status and layout of the applications on the mirror display will be controlled by a single class which will hold the current state of the mirror at all times. This class, MirrorState will have a number of strings and booleans to indicate what is currently active as well as where it should be located on the display. It will also include information about the status of each application such as whether or not it is currently focused or running some specific functionality for the user. This class will be the heart of the application and will essentially manage everything going on. All sensor information will be fed into this class as well to be sorted out so action may be taken.

2.4.1. Proposed for Smart Mirror

Smart Mirror been powered by 12 Volt power supply. The power up the motor so it can move. Another 9V Volt power supply will power up the Arduino, ultrasonic, Bluetooth module, led and buzzer. A model proposed for Smart Mirror is shown in Figure 1.

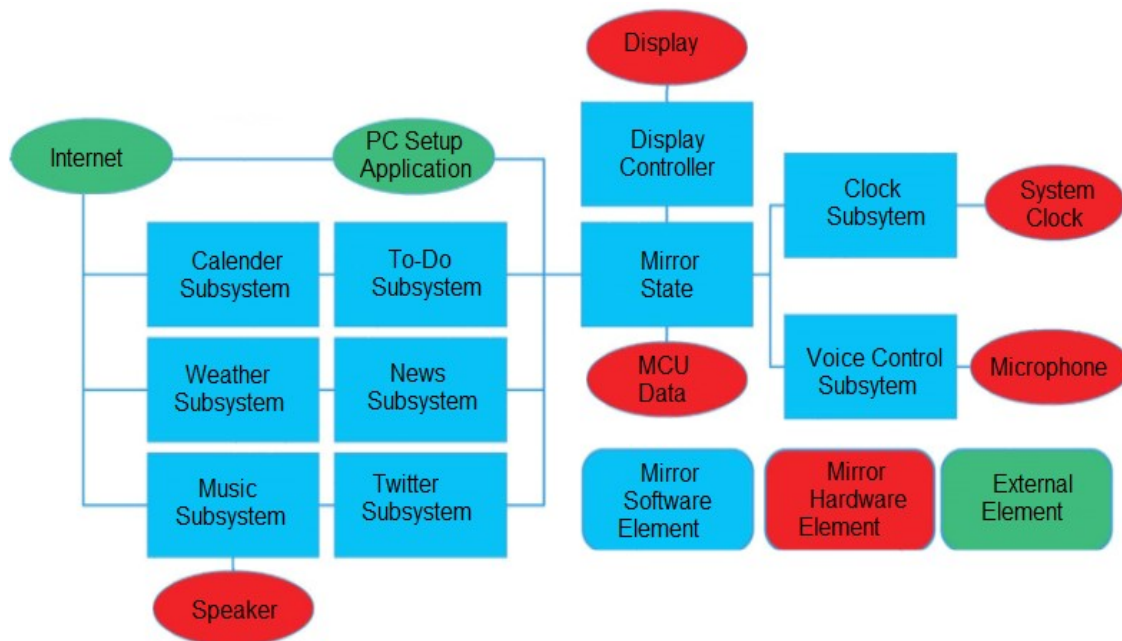


Figure 1. A Model Proposed for Smart Mirror

2.4.2. Schematic

Schematic diagram of Smart Mirror is shown in Figure 2.

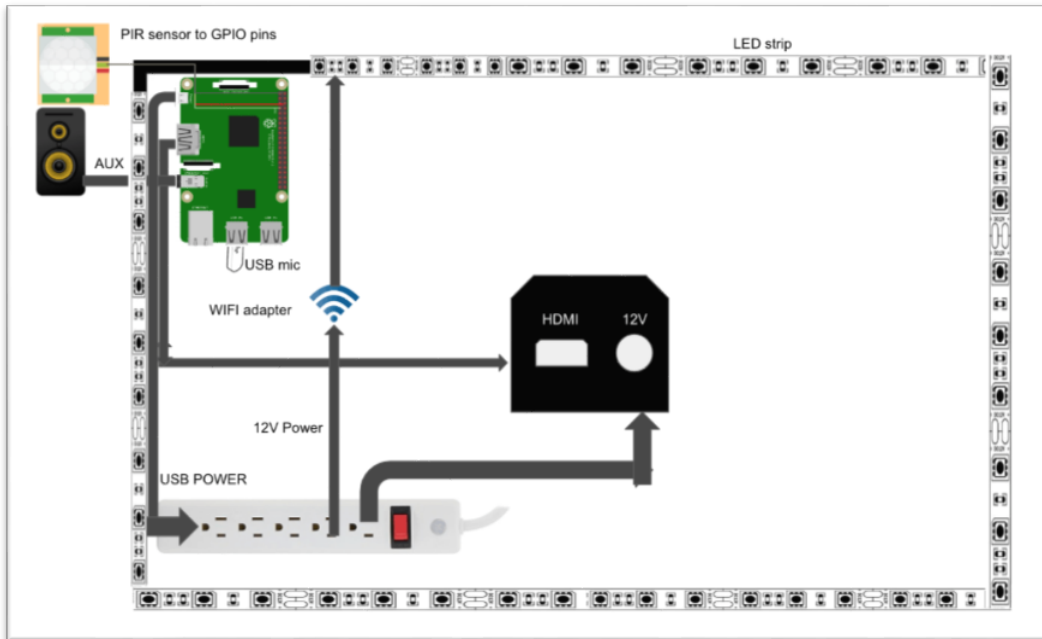


Figure 2. Schematic Diagram of Smart Mirror

Schematic diagram of Smart Home automation is shown in Figure 3.

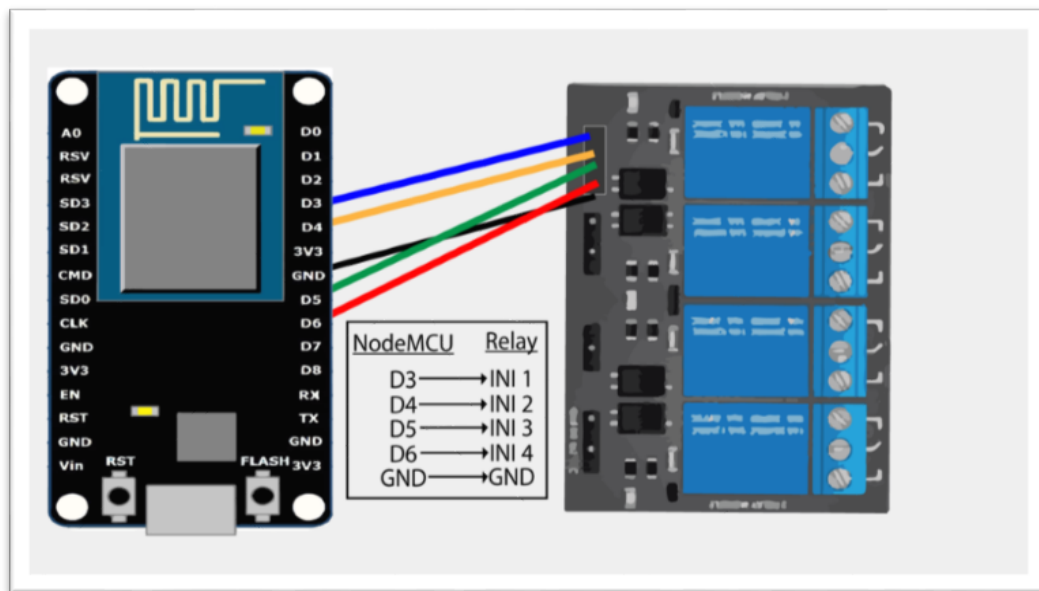


Figure 3. Schematic Diagram of Smart Home Automation

2.4.3. Component

2.4.3.1. Arduino Uno

Microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button [3] [4]. Arduino is shown in Figure 4.

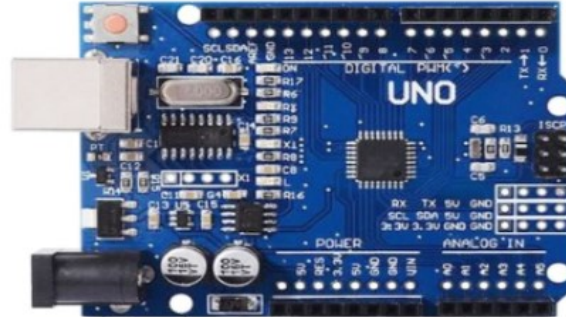


Figure 4. Arduino

2.4.3.2. Raspberry Pi 3

The Raspberry Pi 3 Model B is a tiny credit card size computer. Just add a keyboard, mouse, display, power supply, micro SD card with installed Linux Distribution and everyone will have a fully fledged computer that can run applications from word processors and spreadsheets to games [5] [6]. Raspberry Pi 3 is shown in Figure 5.



Figure 5. Raspberry Pi 3 B+

2.4.3.3. AOC LCD Monitor 22"

Connect the raspberry pi to the monitor for any display appearance. Aoc LCD Monitor 22" is shown in Figure 6.



Figure 6. Aoc LCD Monitor 22"

2.4.3.4. NodeMCU

NodeMCU is the WiFi equivalent of ethernet module [7] [8]. NodeMcu is shown in Figure 7.

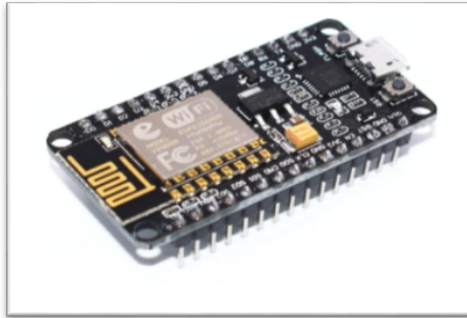


Figure 7. NodeMCU

2.4.3.5. Jumper Wires

Used to interconnect the components of a breadboard Male to male, female to female and male to female wires. Jumper wires male to male is shown in Figure 8.

2.4.3.6. Relay

This is a 5V 4-Channel Relay interfaceboard, be able to control various appliances [9]. Relay 4-Channel is shown in Figure 8.



Figure 8. Relay 4-Channel

2.4.3.7. HDMI Cable

HDMI stands for High Definition Multimedia Interface and is the most frequently used HD signal for transferring both high definition video and audio over a single cable [10]. HDMI cable is shown in Figure 9.

2.5. Implementation

The implementation of the voice recognition software will utilize the Google Assistant API for all voice command processing. The object that listens for voice commands is an implementation of the Speech Recognizer class. This class is run in a separate thread from the main program utilizing the C# async keyword. The object raises interrupt events when speech begins and it listens to the speech then determines if the commands that were spoken fit one of the given rules in any of the grammar constraint files currently loaded. If the command matches any of the rules in the constraint file then the Speech Recognizer object raises an interrupt and passes arguments of what words were spoken in the parameter list to a method specified at the Speech Recognizer initialization. It is then expected for

the developer to write code to handle the interrupt and determine what speech command was given and execute the appropriate code.

2.6. Testing

2.6.1. Hardware Testing

Testing the hardware is a crucial phase of research development that allows everyone to assess what they have accomplished and see what areas they need to improve upon. The testing process is what ensures that the final product performs to the specifications and requirements as they were drawn up before the design phase. It is critical to test the hardware because there are so many different components that must be compatible with each other and sometimes this is not apparent in the research. Each of the different hardware component systems will be subjected to different test criteria in order to determine that they perform their intended function and achieve a passing status.

2.6.2. Software Testing

Throughout the development and testing phases, the smart mirror functionality was constantly being tested using the Debug features within Visual Studio in order to build the application for local execution on our machines. This was made possible by the fact that our smart mirror software is built on the Universal Windows Platform (UWP), allowing us to build a native version that is identical to the version loaded onto the mirror, apart from its processor architecture. A majority of the testing was performed on our local machines because compiling and building the research for the local machine is far quicker than remotely deploying it to the mirror; however, try deployed the research to the mirror when necessary in order to evaluate how the UI was displayed and to make adjustments. Once development was complete, all of the test cases were finally performed on the actual mirror in order to confirm proper functionality for the end user.

2.6.3. Graphical User Interface

The graphical user interface displays all of the various software features that are built into the smart mirror and must conform to a set of requirements on both a functional and nonfunctional level. The functional requirements are comprised of the specific feature set that we implemented into the mirror. Each of the requirements is expected pass an objectively determined goal in order to be considered functional. Conversely, the nonfunctional requirements demonstrate a level of quality assurance that must be subjectively assessed. The expected outcome of these requirements was the culmination of our discussions and experiences while using the smart mirror.

2.6.4. Functional

The functional tests of the graphical user interface encompasses all of the requirements that can be objectively determined as a success or failure. These tests cover proper mirror boot behavior, general user interface function as well as voice recognition behavior. The tests have been divided into the two tables below. Assorted functional Smart Mirror test is shown in Table 1 and speech recognition Smart Mirror tests is shown in Table 2.

2.6.5. Non- Functional

The non-functional tests of the graphical user interface were determined in order to test the requirements that cannot be strictly determined as a success. These subjective tests were determined with our combined thoughts and discussions pertaining to our desired smart mirror behavior. Table 4.5 outlines all of these non-functional tests including the procedure for each as well as the “passing” criteria. Non-functional Smart Mirror test is shown in Table 3.

2.6.6. Non-Functional

The non-functional tests of the graphical user interface were determined in order to test the requirements that cannot be strictly determined as a success. These subjective tests were determined with our combined thoughts and discussions pertaining to our desired smart mirror behavior. Table 4.5 outlines all of these non-functional tests including the procedure for each as well as the “passing” criteria. Non-functional Smart Mirror test is shown in Table 3.

2.7. Renewal

This is the final stage in implementing the Smart Mirror. At this stage, Smart Mirror has been restored based on observations and tests conducted on UPSI students and surrounding residents in Tanjung Malim.

Table 1. Assorted Functional Smart Mirror Test

TESTING	PROCEDURE	EXPECTED OUTCOME
Smart Mirror GUI on boot	Power up the mirror by plugging in the power cable.	The Smart Mirror app should load upon boot rather than the Windows 10 IoT Core Dashboard utility.
Initial application data fetch	Boot up the mirror and ensure that the Smart Mirror app is booted.	The various software features should fetch their respective data such as the weather conditions and news headlines.
Application data refresh	Observe whether the various software features update their content according to their set refresh cycles. Reduce the refresh intervals in the code to expedite the testing.	Each software feature should update within 5 seconds after their specified refresh intervals have passed (given that the data from the API is different from the existing data).
Excessive Temperature/Humidity Warning	Manually initiate the code path that will trigger the warning associated with exceeding favorable operating threshold.	The mirror will display a warning message to user to indicate that the humidity/temperature sensor has detected unfavorable operating conditions.
GUI becomes hidden automatically	With the GUI visible, step away from the motion sensors and wait a moment.	Once the mirror has not detected any motion after 60 seconds, the UI elements will become hidden.
Loss of Internet Connectivity Test	Remove the USB WiFi dongle to force a loss of internet connectivity.	In the event of internet connectivity disruption, the software features that rely on the internet connection should respond as programmed without compromising total system functionality.

Table 2. Speech Recognition Smart Mirror Tests

TESTING	PROCEDURE	EXPECTED OUTCOME
Different weather views	Use speech recognition to access the different weather view options. Available voice prompts: <ul style="list-style-type: none"> • Show/hide tomorrow's weather. • Show/hide this week's weather. • Show today's weather. 	The voice commands should successfully display the requested weather conditions in at least 75% of all attempts.
Show/Hide GUI	Use speech recognition to show or hide all of the software feature elements by saying "Mirror On" or "Mirror off" respectively.	The voice commands should show or hide the GUI successfully in at least 75% of all attempts.
Music Control	Use speech recognition to initiate music searches and control playback. Available voice prompts: <ul style="list-style-type: none"> • Search Pandora for 'Song, Artist, Genre' • Pause music and Next track 	The voice commands should successfully perform the music-related functions in at least 75% of all attempts.

Table 3. Non-Functional Smart Mirror Test

TESTING	PROCEDURE	EXPECTED OUTCOME
Boot-up time	Boot up the mirror by plugging in the power cable.	The Smart Mirror app should boot within 30 seconds of providing power.
Shutdown time	Shut down the mirror properly by initiating a shutdown with the companion app.	The Smart Mirror should shut down completely within 20 seconds of issuing the command.
Speaker Volume Level	Initiate Pandora music playback using speech recognition and assess the volume level.	All group members should agree that the volume level is appropriate and audible for the mirror.
Speech Recognition Consistency	Have all group members perform the multitude of speech recognition prompts, and variations of each, and observe the mirror's response.	The mirror should appropriately respond to the voice commands and variations with 75% success rate.
Display Brightness	Place the mirror in a bright environment and power it up.	The user interface elements should still be visible despite the unfavorable lighting conditions. The rest of the screen real estate should retain its mirror-like finish as provided by the display's reflective tint.
Motion Detector Reaction	Put the mirror into an inactive state with the user interface hidden and then trigger the motion sensor by walking in front of its line of sight. Repeat this process 10 times.	The mirror should become active with its user interface elements visible within 5 seconds of stepping into its line of sight.

3. Result

Physical of Smart Mirror is shown in Figure 10.



Figure 10. Physical of Smart Mirror

4. Discussion

4.1. Strengthen

Smart Mirrors are interactive devices that helps everyone check updates easily with voice control. People can do things at the same time in the morning like brushing teeth while catching up on the latest stocks updates, weather or traffic. People can watch shows while preparing going to work. Some smart mirrors has true light for vanity purposes or touchless on and off sensor. Smart mirrors can help everyone to save time and just multitask.

4.2. Weakness

There is a weakness in Smart Mirror that have to be fixed in the future. One of the weakness is the Smart Mirror cannot run if do not have any connection from the internet. Smart Mirror only can get inform if that have connection to the internet such weather, news and so on. If Smart Mirror run without any connection from the internet that screen will blank and not appear any information. Second weakness in the Smart Mirror is the power, without any power from the socket plug smart mirror can run. The point here is user cannot save the power electric. So, maybe electric bill's user go up.

5. Conclusion

At minimum, the research set out to have a mirror that could display a dynamic date and time. This was our minimum criteria for success, and we are happy with the display that resulted. Researcher also able to include a researcher ather display, though unfortunately that particular element is not as visible as the other display items. Researcher also disappointed at not being able to incorporate an infrared sensor into our mirror, which would have let us program a motion-activated start. Unfortunately, time and the difficulty of programming the other elements forced us to shelve this particular feature. Future versions would need to work on reducing light and slimming down the product.

References

- [1] M. M. Elsobeihi, and S. A. N. Samy, "Effects of Mobile Technology on Human Relationships," *International Journal of Engineering and Information Systems (IJEAIS)*, vol. 1, no. 5, pp. 110-125, 2017.
- [2] R. Cohen, and W. Tao, *Overview of Embedded Application Development for Intel Architecture*. Brekeley: Apress, 2014.
- [3] Mridul0001, "Control House Lights with Google Assistant Using Arduino," *Instructable*, 2018. [Online]. Available: <https://www.instructables.com/id/Control-House-Lights-With-Google-Assistant-Using-A/>. [Accessed: May 28, 2019].
- [4] I. Y., Panessai, M. M.Lakulu, , S. Subramaniam, , A. F.,Saad, M. I. Damanhuri, and N. I. Yusuf, "Developing a Prototype for Sun Tracker System Based on IoT: Controlled by Mobile App and Online Database Monitoring," *American Journal of Applied Sciences*, vol. 16, no. 1, 2019. doi: 10.3844/ajassp.2019.11.25.
- [5] I. Buckley, "How to Build a DIY Google Home Assistant with Raspberry Pi. *Make Use of*. [Online]. Available: <https://www.makeuseof.com/tag/diy-google-home-assistant-raspberry-pi/>. Retrieved: 2018, 22th October. (2018).
- [6] (2018). Zach. Build a Voice-Controlled DIY Raspberry Pi Smart Mirror with Jasper," *Howchoo*, 2018. [Online]. Available: <https://howchoo.com/project/mzu3ndm2otu/building-a-voice-controlled-smart-mirror-with-raspberry-pi-and-jasper>. [Accessed: March 10, 2019].
- [7] M. Fezari, N. Zakaria, and A. Al Dahoud, *Comparative Study between Two Powerfull Nodemcu Circuits: ESP32 And ESP8266*. Jordan: Al-Zaytoonah University, 2019.
- [8] A. Hingorani. Home Automation Using NodeMCU and Google Assistant in Under \$20. *Codeometry*, 2018. [Online]. Available: <https://codeometry.in/home-automation-using-nodemcu-and-google-assistant/>. [Accessed: March 2019].
- [9] I. Irvawansyah, and A. Rahmansyah, "Prototype of Monitoring and Control System of SCADA-based Water Tank Level," *JTT (Jurnal Teknologi Terapan)*, vol. 4, no. 3, 2018. [Online]. Available: 31884/jtt.v4i1.88. [Accessed: April 2019].
- [10] V. Chayapathy, "Iot Based Home Automation by Using Personal Assistant," *International Conference on Smart Technology for Smart Nation*. Bengaluru, India, 2018.