

MAGIC MIRROR - A Smart Reflective Interface for a Smart Home

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Technology degree in
Information Technology

By

LAKSHMINARAYANA K (Reg.No - 39120060)
KARTHIKREDDY M (Reg.No – 39120051)



DEPARTMENT OF INFORMATION TECHNOLOGY
SCHOOL OF COMPUTING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A” by NAAC | 12B Status by UGC | Approved by AICTE
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119

MAY- 2023



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF INFORMATION TECHNOLOGY

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **LAKSHMINARAYANA K (Reg.No - 39120060)** and **KARTHIKREDDY M (Reg.No - 39120051)** who carried out the Project Phase-2 entitled "**MAGIC MIRROR - A Smart Reflective Interface for a Smart Home**" under my supervision from January 2023 to May 2023.

Internal Guide

Dr. R.M.GOMATHI, M.Tech., Ph.D

Head of the Department

Dr. R. SUBHASHINI, M.E., Ph.D.

Submitted for Viva voce Examination held on _____

Internal Examiner

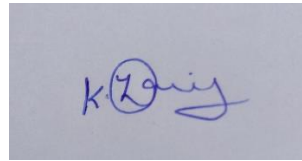
External Examiner

DECLARATION

I, **Lakshminarayana K (Reg.No- 39120060)**, hereby declare that the Project Phase-2 Report entitled “**MAGIC MIRROR- A Smart ReflectiveInterface for a Smart Home**” done by me under the guidance of **Dr. R. M. GOMATHI, M.Tech., Ph.D.** is submitted in partial fulfillment of the requirements for the award Bachelor of Technology degree in Information Technology.

DATE:

PLACE: Chennai



SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. SASIKALA, M.E., Ph. D. Dean**, School of Computing, **Dr. R. SUBHASHINI, M.E., Ph.D.** Head of the Department of Information Technology for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr. R. M. GOMATHI, M.Tech., Ph.D.** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my phase-2 project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Information Technology** who were helpful in many ways for the completion of the project.

ABSTARCT

Our day begins with looking at our reflection in the mirror. This project depicts the design and development of a smart mirror. Our day begins with the reality that we take a look at our image in reflection. Magic mirror or Smart mirror is an application of IOT where in we would be using micro controllers like Raspberry Pi embedded into the mirror that would be displaying required information such as date, time, news, things to do and many more along with acting like a normal mirror. These mirrors are these days programmed using Artificial Intelligence which makes the product expensive. Hence our main goal is to develop a product that is economical and efficient enough to satisfy all the basic needs of the common people. These mirrors are these days programmed using Artificial Intelligence which makes the product expensive. Hence our main goal is to develop a product that is economical and efficient enough to satisfy all the basic needs of the common people.

TABLE OF CONTENT

Chapter No	TITLE	Page No.
	ABSTRACT	v
	LIST OF FIGURES	vii
1	INTRODUCTION	1
2	LITERATURE SURVEY	4
	2.1 Inferences from Literature Survey	
	2.2 Open problems in Existing System	6
3	REQUIREMENTS ANALYSIS	
	3.1 Feasibility Studies	7
	3.2 Software Requirements	11
	3.3 System Use case	11
4	DESCRIPTION OF PROPOSED SYSTEM	
	4.1 Selected Methodology And process model	13
	4.2 Overall Design of Proposed System	15
	4.3 Description of Software for Implementation and Testing plan of the Proposed Model	16
5	IMPLEMENTATION DETAILS	
	5.1 Development and Deployment Setup	17
	5.2 Algorithms	19
6	RESULTS AND DISCUSSION	20
7	CONCLUSION AND FUTURE WORK	
	7.1 Conclusion	23
	7.2 Future work	23
	REFERENCES	24
	APPENDIX	
	A. SOURCE CODE	26
	B. SCREENSHOTS	39
	C. RESEARCH PAPER	41

LIST OF FIGURES

FIGURE NO	FIGURE NAME	Page No.
1.1	Typical structure of a Smart mirror	2
1.2	Smart Mirror	3
3.1	Raspberry pi Symbol	7
3.2	Raspberry pi kit	8
3.3	Web Cam	9
3.4	DTH22 Sensor	10
3.5	Connection Board	11
4.1	System Architecture	13
4.2	Components	14
4.3	Smart Mirror Display Screen	15
6.1	Inner look of magic mirror	20
6.2	Outer look of magic mirror	21
6.3	Smart mirror displays date, time, news	22

CHAPTER 1

INTRODUCTION

In this world everyone needs a comfortable life. Modern man has invented different technology for his purpose. In today's world, people need to be connected and they are willing to access the information easily. Whether it is through the television or internet, people need to be informed and be in touch with current affairs happening around the world. The Internet transformed our lives by connecting us more easily to information and other people in the virtual world. Mobile phones then became smartphones and since then this concept has erupted and morphed into the Internet of Things, things which connect us to everyday objects. There are no end of objects that could be made "smarter", some being more suited to this than others. Mirrors, for example, provide a large surface ideal for displaying information and interacting with. Most people have mirrors at home so the concept of a smart mirror that you can interact with is attractive and has been fantasized in many futuristic movies. Smart mirrors, such as Magic Mirror and Home Mirror have recently started to be developed by people in the Maker community, with varying degrees of interactivity. However, so far, the features of these mirrors have been limited. The innovation and research work in the field of Artificial intelligence, Machine learning, Internet of things has brought a massive change in the technology we use and paved the way for a Smart environment.

Many devices are being developed which use concepts of multimedia communication, artificial intelligence, internet of things (IoT) to revolutionize the way we perform our various day to day tasks in our home, offices or even industries. Most of us use mirrors every day to look at ourselves; we psychologically interact with the mirror every day to check how we look and how our attire is while getting ready for our work or colleges.

The Internet of Things means interconnection via the internet of computing devices embedded in everyday objects, enabling them to send and receive data. The Internet of Things with its enormous growth widens its applications to the living environment of the people by changing a home to a smart home. Home automation has been around for a long time and it is all about turning the house into an intelligent unit with the goal of increasing comfort and efficiency at home. Some of

the typical applications are automatic lights, intelligent thermostats, alarms, window blinds.

Smart home is a connected home that connects all types of digital devices to communicate with each other through the internet. Our lifestyle has evolved in such a way that optimizing time is the most important thing. Our work is based on the idea that we all look at the mirror when we go out, so why wouldn't the mirror become smart and magical.

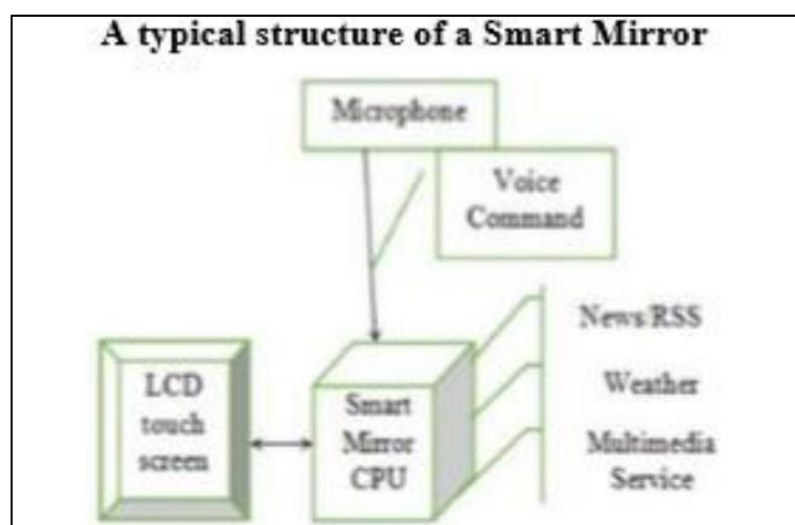


Figure1.1. Typical Structure Of A Smart Mirror

In this project, There is a multipurpose mirror that is meant to serve as both decoration and information source. A common approach for building a magic mirror is to use a high quality two-way glass, a LCD and LED screens, a frame to hold the glass, and sensors to provide us with the accurate information required to make the normal mirror a smart mirror.



Figure 1.2 Smart Mirror

With just one look in the mirror, one will have the basic information on what to wear based on the weather forecast for that day or how much time they have left if they want to arrive on time at the planned destination. This magic mirror is still complex to that of an ordinary mirror, having a display inside a glass that one can interact with, us.

The mirror works with the help of micro controllers such as Arduino UNO and Node MCU. There are several displays or notice boards present that are difficult to operate manually. The problem of finding places in new buildings was too much of a concern. This product acts as a solution for the above problems. It also acts as an innovative and attractive object that can be placed in any surrounding.

CHAPTER 2

LITERATURE SURVEY

2.1 Inferences from Literature Survey

Suryansh Chandel, Ashay Mandwarya and S.Ushasukhanya et al. [1] implemented a voice controlled wall mirror. It is a device that can function both as a mirror and an interactive display, displaying multimedia content such as time, date, weather and news simultaneously. It consists of various functionalities like real time data and information updates, voice commands, face detection/recognition using LCD monitor, microphone and webcam. Tathagat Sinha et al. [2] proposed a mirror which works on common architecture and also has all the required functions for the user. Gary Chew and Kyle Stadelmann et al. [3] delivered a working smart mirror from scratch. They built an open source 1-second-a-day module, allowing anybody with a Smart Mirror to record their own 1-second-a-day videos. Third, they also built a patch notes module, allowing Overwatch gamers to stay updated on the game's latest patches.

Neaz Ahsan Chowdhury and Kanchon Sarker et al. [4] created a low cost Smart Mirror device that people could interact with but also to further develop the technology. A few problems arose in the physical construction and software side of the project, such as the glass not being reflective enough but these drawbacks were addressed by doing more tests and trials to further develop the Smart Mirror. Chris Rectenwald, Teague Kohlbeck, Benny Richmond and Mike Schafer et al. [5] provided a single easy to access location for a person to receive all the information that could affect how they prepare for the day. Through the use of LCD displays and a two way mirror it displayed weather, time and date, and news that are available at a glance. Additionally, a user-friendly interface, accessible from any WiFi enabled device, allows the user to easily setup the connection to their home WiFi, change the location from which they receive the weather updates and select a source from which to receive the day's headlines.

Lakshmi N M, Chandana M S, Ishwarya P, Nagarur Meena and Rajendra R Patil et al. [6] depicted the design and development of a smart mirror that represents an elegant interface for glancing information and also used for thief detection in a home environment. Dr. C.K. Gomathy, Mr. R. Venkata Narayana and Mr. T. Giridhar Reddy et al. [7] developed a system that works on the functions of additional capability of displaying date, time, current temperature, weather details. The system is based on the IoT-based intelligence to facilitate and enhance the user's awareness. Sadeta KULOVIC and Belma RAMIC-BRKIC et al. [8] developed a multipurpose mirror used both as a mirror and a device. The displayed information on the mirror is accessible at any point in time. The cost-effective analysis indicated the significant cost savings for produced outcomes compared to available off-the-shelf solutions. Justin Gentry, Michael Trivelli and Hector Zacarias et al. [9] constructed a mirror that will be able to present personalized information to users every morning as they prepare for their day. Ideally it will save user's time by displaying information they would likely check in the morning. This included information such as weather, daily schedule, news and also the time so the user can keep on schedule. The information provided on the mirror in an unobtrusive manner, leaving the majority of the prime mirror real estate unaltered. This allows the user to easily absorb the displayed information while going about their normal routine.

Shreyansh Khale, Aditi Sathe, Rugveda Salunke, Shweta Nathan and Amit Maurya et al. [10] proposed a mirror made smart called Smart mirror which they built using Raspberry Pi. The visual representation of the product is similar to a regular mirror that can display weather details, temperature, time and daily news on voice commands. The smart mirror acts as a personal assistant, an enquiry center and displays important curriculum or college notices.

2.2 OPEN PROBLEMS IN EXISTING SYSTEM

Smart home is a connected home that connects all types of digital devices to communicate with each other through the internet. Many devices are being developed which use concepts of multimedia communication, artificial intelligence, internet of things (IoT) to revolutionize the way we perform our various day to day tasks in our home, offices or even industries.

So, The idea of having an interactive mirror that can respond to your commands can excite anyone. Magic Mirror aims at augmenting the basic reflective mirror with embedded intelligence to combine daily routine tasks like reading newspaper, getting stock updates, weather updates etc. and providing all that data to the user while he/she gets ready. We present a multipurpose mirror that is meant to serve as both decoration and information source. The problem of finding places in new buildings was too much of a concern. This product acts as a solution for the above problems. It also acts as an innovative and attractive object that can be placed in any surrounding.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES

3.1.1. RASPBERRY PI

Raspberry Pi is a series of small, single-board computers developed to teach computer science basics to school students and other people in low-income countries. It became a popular and easy to experiment tool to develop school projects, hardware programming, robotics, basic automated machines, circuits, etc. The Uses of Raspberry Pi is a small, quite affordable, and very much capable hardware device called a credit card size computer. The Raspberry Pi Foundation develops it. Many consider it the most versatile tech ever created because of its simplicity and great usability.

Created by Eben Upton, its goal was to create a low-cost device that could enhance programming skills and understanding of the hardware to students and tech enthusiasts. The Raspberry Pi has a row of GPIO, also called General Purpose input/output pins, that can be used to interact with keyboards, mouse, monitors, etc. and create amazing stuff. Raspberry Pi is used in several applications, from building cools gadgets and games to monitoring and surveillance devices. In this Uses of Raspberry Pi article, we shall focus on the top Uses of the raspberry pi.

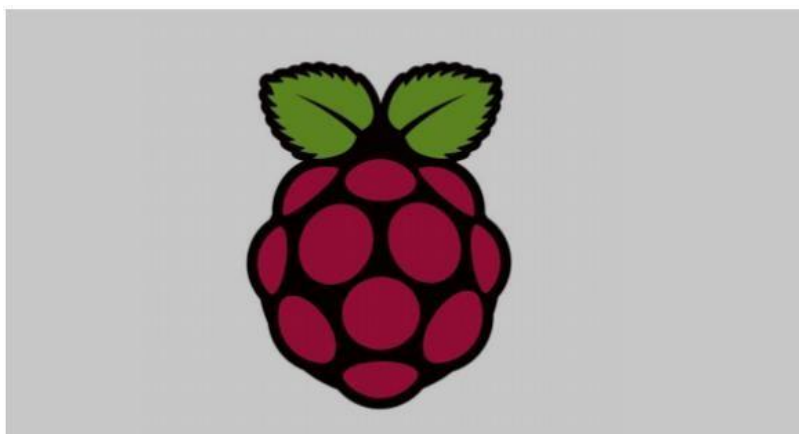


Figure 3.1 Raspberry pi symbol

Raspberry pi is used to implement the project. Raspberry pi contains separate operating system to work. In that software the coding is going to do and using the language is optional .There is no specific language to use .The Raspberry Pi operates in the open source ecosystem: it runs Linux (a variety of distributions), and its main supported operating system, Pi OS, is open source and runs a suite of open source software. The Raspberry Pi Foundation contributes to the Linux kernel and various other open source projects as well as releasing much of its own software as open source. Some people buy a Raspberry Pi to learn to code, and people who can already code use the Pi to learn to code electronics for physical projects. The Raspberry Pi can open opportunities for you to create your own home automation projects, which is popular among people in the open source community because it puts you in control, rather than using a proprietary closed system.

Raspberry Pi OS is a free operating system based on Debian, optimized for the Raspberry Pi hardware, and is the recommended operating system for normal use on a Raspberry Pi. The OS comes with over 35,000 packages: per-compiled software bundled in a nice format for easy installation on your Raspberry Pi. Raspberry Pi OS is under active development, with an emphasis on improving the stability and performance of as many Debian packages as possible on Raspberry Pi.



Figure 3.2 Raspberry pi kit

3.1.2. WEB CAM

A webcam is a digital video device commonly built into a computer. Its main function is to transmit pictures over the Internet. It is popularly used with instant messaging services and for recording images. *Note: The instructions for this webcam are specific to a brand and may vary for other brands and products. They can provide advanced features such as image archiving, motion sensing, custom coding, or even automation. Furthermore, webcams are used for social video recording, video broadcasting, and computer vision and mainly used for security surveillance and in videoconferencing. A webcam is an input device that captures digital images. These are transferred to the computer, which moves them to a server. From there, they can be transmitted to the hosting page. Laptops and desktops are often equipped with a webcam.



Figure 3.3 Webcam

3.1.3 DTH22 SENSOR

DHT22 capacitive humidity sensing digital temperature and humidity module is one that contains the compound has been calibrated digital signal output of the temperature and humidity sensors. Application of a dedicated digital modules collection technology and the temperature and humidity sensing technology, to ensure that the product has high reliability and excellent long-term stability. The sensor includes a capacitive sensor wet components and a high-precision

temperature measurement devices, and connected with a high-performance 8-bit micro controller.

The product has excellent quality, fast response, strong anti-jamming capability, and high cost. Standard single-bus interface, system integration quick and easy. Small size, low power consumption, signal transmission distance up to 20 meters, making it the best choice of all kinds of applications and even the most demanding applications.

DHT22 has higher precision and can replace the expensive imported SHT10 temperature and humidity sensor. It can measure the environment temperature and humidity to meet the high demand. The product has high reliability and good stability. If it's used and combined with special sensor Arduino expansion board, it will be easily implemented the interactive effect which related to the temperature and humidity perception. Caution: DHT22 digital temperature and humidity sensor is designed for analog sensor interfaces. The analog port will be used as the digital which will not occupy the original digital port of the Arduino. The lines of the sensor which can transform the analog function to digital that can be use on digital port.

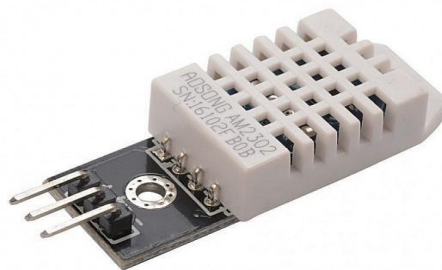


Figure 3.4 DTH22 SENSOR

CABLES:

You will need one or more cables to connect up your RPi system.

- Video cable alternatives: o HDMI-A cable o HDMI-A cable + DVI adapter o Composite video cable o Composite video cable + SCART adaptor
- Audio cable (not needed if you use the HDMI video connection to a TV)
- Ethernet/LAN cable (Model B only)

USB hub:

In order to connect additional devices to the RPi, you may want to obtain a USB hub, which will allow multiple devices to be used. It is recommended that a powered hub is used - this will provide any additional power to the devices without affecting the RPi itself. A USB 2.0 model is recommended. USB 1.1 is fine for keyboards and mice, but may not be fast enough for other accessories.

3.2 Software Requirements Specific Document

HARDWARE CONFIGURATION

Processor: Processor Intel CORE i5 and above

Internet Connection: Existing telephone lines, Fiber nets, Data card

Ram: 8GB

SOFTWARE CONFIGURATION

Operating System: Raspberry pi

Language: python

3.3 System Use Case

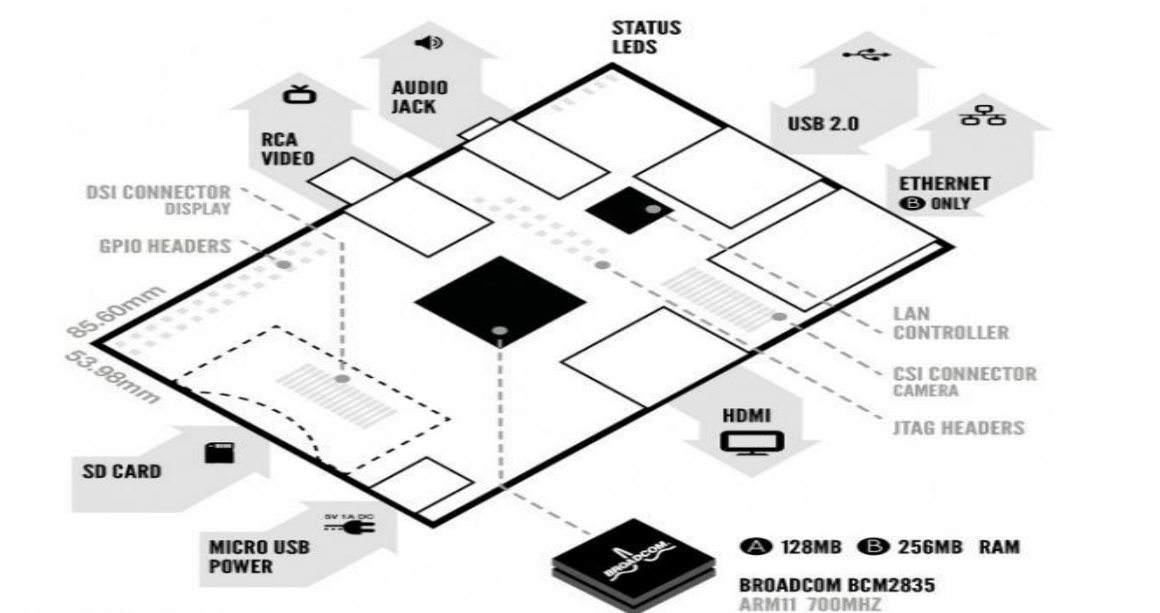


Figure 3.4 Connection Board

The unit is powered via the micro-USB connector (only the power pins are connected, so it will not transfer data over this connection). A standard modern phone charger with a micro-USB connector will do, providing it can supply at least 700mA at +5Vdc. Check your power supply's ratings carefully. Suitable mains adapters will be available from the RPi Shop and are recommended if you are unsure what to use. The individual USB ports on a powered hub or a PC are usually rated to provide 500mA maximum. If you wish to use either of these as a power source then you will need a special cable which plugs into two ports providing a combined current capability of 1000mA.

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

4.1 Selected Methodology And process model

According to research, there are many different and interesting approaches to building a smart mirror. Some of the solutions are available on the market, while others are just prototypes or Do-It-Yourself (DIY) projects developed by enthusiasts for their own use.

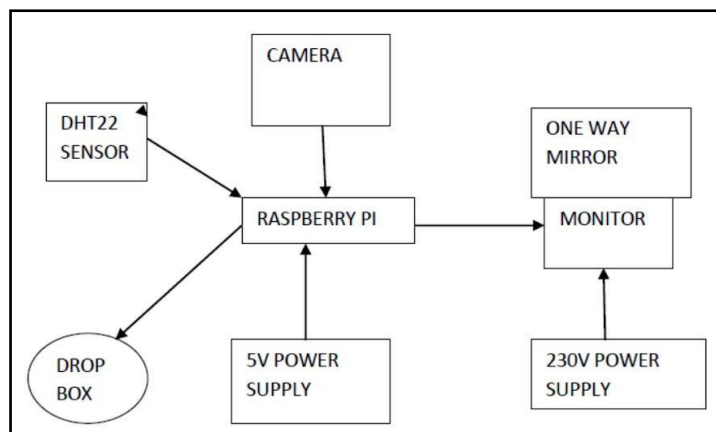


Figure 4.1. System Architecture

PROPOSED MODEL

The Proposed methodologies that were used till now have been developed using a Raspberry Pi microprocessor to display all the multimedia content and act as a normal mirror too. These products have been manufactured by the top most known companies and brands and have been sold into the market long ago, but only very few of the population know about the smart mirrors available and can afford them . A lot of these smart devices have the major requirement of social networking and connectivity that make them smarter. The main features that all of these smart mirrors have in common are the weather report , date and time displays , news reports and audio or music support .

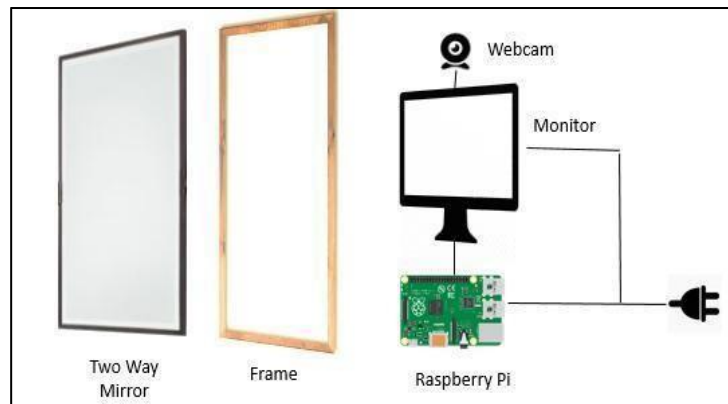


Figure 4.2 Components

FUNCTIONING OF PROPOSED MODEL

The proposed methodology implements the basic amenities required by a mirror to act as a smart mirror. The requirement of this device would be a discovery where the technology and the ordinary elements of human life merge and supply a far better experience. This methodology primarily focuses on building a cost effective and efficient smart device that is feasible.

We even have sensors mounted and connected to the micro controller to get the various inputs such as temperature, air quality etc. that could be displayed too on the screens. We have proposed the comparative study and a design of a futuristic smart mirror which could be a great device for ambient home services. Live animations will make the bathroom more fashionable.

4.2 Overall Design of Proposed System

The idea of having an interactive mirror that can respond to your commands can excite anyone. Magic Mirror aims at augmenting the basic reflective mirror with embedded intelligence to combine daily routine tasks like reading newspaper, getting stock updates, weather updates etc. and providing all that data to the user while he/she gets ready.



Figure 4.3. Smart Mirror Display Screen

The Internet of Things means interconnection via the internet of computing devices embedded in everyday objects, enabling them to send and receive data. The Internet of Things with its enormous growth widens its applications to the living environment of the people by changing a home to a smart home. Home automation has been around for a long time and it is all about turning the house into an intelligent unit with the goal of increasing comfort and efficiency at home. Some of the typical applications are automatic lights, intelligent thermostats, alarms, window blinds.

4.3 Description of Software for Implementation and Testing plan of the Proposed Model

Raspberry pi is used to implement the project. Raspberry pi contains separate operating system to work. In that software the coding is going to do and using the language is optional .There is no specific language to use .The Raspberry Pi operates in the open source ecosystem: it runs Linux (a variety of distributions), and its main supported operating system, Pi OS, is open source and runs a suite of open source software. The Raspberry Pi Foundation contributes to the Linux kernel and various other open source projects as well as releasing much of its own software as open source.

Some people buy a Raspberry Pi to learn to code, and people who can already code use the Pi to learn to code electronics for physical projects. The Raspberry Pi can open opportunities for you to create your own home automation projects, which is popular among people in the open source community because it puts you in control, rather than using a proprietary closed system.

Raspberry Pi OS is a free operating system based on Debian, optimized for the Raspberry Pi hardware, and is the recommended operating system for normal use on a Raspberry Pi. The OS comes with over 35,000 packages: per-compiled software bundled in a nice format for easy installation on your Raspberry Pi. Raspberry Pi OS is under active development, with an emphasis on improving the stability and performance of as many Debian packages as possible on Raspberry Pi.

CHAPTER 5

IMPLEMENTATION DETAILS

5.1 DEVELOPMENT AND DEPLOYMENT

Raspberry Pi Advanced Setup

You'll need a preload-ed SD card, USB keyboard, TV/Monitor (with HDMI/ DVI/ Composite/SCART input), and power supply (USB charger or a USB port from a powered USB Hub or another computer). You'll likely also want a USB mouse, a case, and a USB hub (a necessity for Model A). A powered USB hub will reduce the demand on the RPi. To connect to the Internet, you'll need either an Ethernet/LAN cable (Model B) or a USB WiFi adaptor (either model). When setting up, it is advisable to connect the power after everything else is ready.

Serial connection :

The Serial Port is a simple and uncomplicated method to connect to the RPi. The communication depends on byte wise data transmission, is easy to setup and is generally available even before boot time.

First interaction with the board:

Connect the serial cable to the COM port in the RPi, and connect the other end to the COM port or USB Serial Adapter in the computer.

Serial Parameters:

The following parameters are needed to connect to the RPi. All parameters except Port_Name and Speed are default values and may not need to be set. Port_Name: Linux automatically assigns different names for different types of serial connectors.

Choose your option:

- USB Serial Port Adapter: ttyUSB0 ... ttyUSBn
- Speed: 115200
- Bits: 8
- Parity: None
- Stop Bits: 1

- Flow Control: None The Serial Port is generally usable by the users in the group dialout. To add oneself to the group dialout the the following command needs to be executed with root privileges:

```
$useradd -G {dialout} your_name
```

Super Easy Way using GNU Screen:

Enter the command below into a terminal window

```
screen Port_Name 115200
```

Super Easy Way using Minicom Run minicom with the following parameters:

```
minicom -b 115200 -o -D Port_Name
```

GUI method with GtkTerm:

Start GtkTerm,

select Configuration->Port and enter the values above in the labelled fields.

Windows Users:

Windows 7 or Vista users must download putty or a comparable terminal program.

Users of XP and below can choose between using putty and Hyper Terminal.

First Dialog:

If you get the prompt below, you are connected to the Raspberry Pi shell!

```
prompt> #
```

First command you might want try is "help":

```
prompt> # help
```

If you get some output, you are correctly connected to the Raspberry Pi!

Congratulations!

SD card setup:

Now we want to install a GNU/Linux distro on an SD card and make space for our stuff. You can use either an SD or SDHC card. In the latter case of course take care that your PC card reader also supports SDHC. Be aware that you are not dealing with an x86 processor, but instead a completely different architecture called ARM, so don't forget to install the ARM port for the distro you are planning to use.

Formatting the SD card via the mkcard.txt script:

1. Download mkcard.txt .
2. \$ chmod +x mkcard.txt
3. \$./mkcard.txt /dev/sdx, where x is the letter of the card.

You can find this by inserting your card and then running `dmesg | tail`. You should see the messages about the device being mounted in the log. Mine mounts as `sd`. Once run, your card should be formatted.

5.2 ALGORITHM

Step1 : Switch on the power supply

Step2 : Get the date,time,and Weather details from predefined URL.

Step3 : Get the news from www.news.com.

Step4 : In code section write down all the compliments to be displayed on mirror.

Step5 : Display it on mirror via LCD monitor.

Step6 : Switch off the power supply when it is of no use.

CHAPTER 6

RESULTS

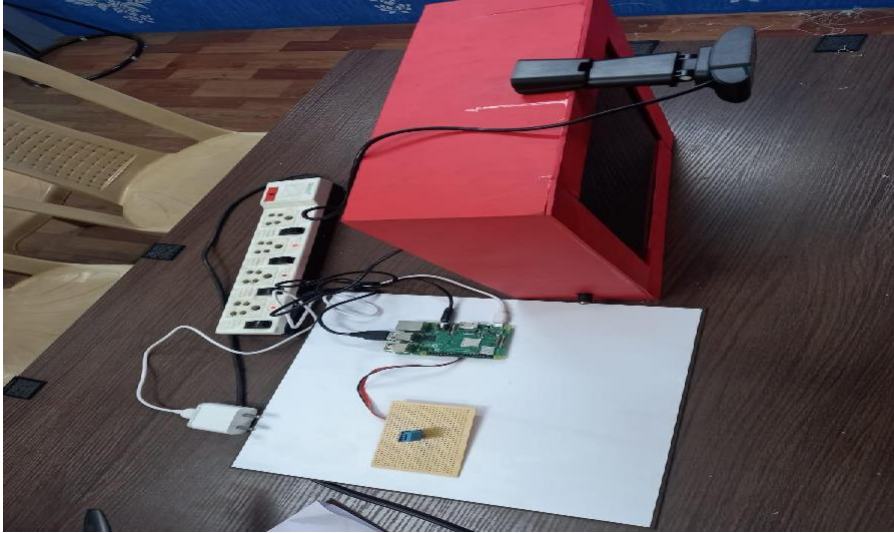


Figure 6.1 Inner look of magic mirror

Above figure shows the inner look and components of the magic mirror and the connectivity of the magic mirror. It shows the camera, two way mirror, raspberry pi, wires, DTH22 sensor, adapter.



Figure 6.2 Out look of the magic mirror

The above figure says that how the magic mirror look after importing all the components and connections. The charging port is more important without that it will not work and WiFi also most important to show the content .



Figure 6.3. Smart mirror displays date,time,news

The above figure shows the content on the screen and the content was the present time date additional title and the news and holidays .This all are operated using WiFi module and this project needed the current supply to run the device.

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1. CONCLUSION

Hence we have come up with the proposed methodology to use a different micro controller and develop the smart mirror instead of using the same Raspberry Pi that every other existing methodologies have already implemented or used. Our main goal is to develop this project using the basic available products that are extremely feasible and efficient other than the commonly used microprocessor Raspberry Pi. We want the device to be developed and made economically efficient so every common man will be able to afford it . During this phase 1, we have learnt a lot of new concepts and also had a detailed study about the available products that have been already existing in the market. We have proposed the comparative study and a design of a futuristic smart mirror which could be a great device for ambient home services. Live animations will make the bathroom more fashionable.

The proposed Magic Mirror can be easily extended for some other functions such as making phone calls. In the future this mirror can be used to build smart home networks with devices such as lights, virtual assistants, TV, music systems, refrigerators, etc. can be integrated together. This would lead to the real smart home.

7.2. FUTURE WORK

In future along with existing one trying to add a Security system. It means when there is nobody in home it can be switched into security by using VNC viewer to detect human presence. When someone enters into room ,PIR sensor will detect the movement of the person when he passes by the mirror and capture the image and stores it in the drop box. It also informs the owner by updating captured image in the drop box, by this way smart mirror system can also be used as security system.

REFERENCES

- [1] SuryanshChandel, Ashay Mandwarya and S.Ushasukhanya, "IMPLEMENTATION OF MAGIC MIRROR USING RASPBERRY PI 3" International Journal of Pure and Applied Mathematics, All Rights ReservedPage | ISSN: 1314-3395 (on-line version). Volume 118 No. 22 2018, 451-455, July 09, 2020.
- [2] Tathagat Sinha, " SMART MIRROR USING RASPBERRY PI " . May, 2020. http://103.47.12.35/bitstream/handle/1/1165/project_report_15scse101346_tathagat_sinha_pdf%20-%20Tathagat%20Sinha.pdf?sequence=1&isAllowed=y.
- [3] Tathagat Sinha, " SMART MIRROR USING RASPBERRY PI " . May, 2020. http://103.47.12.35/bitstream/handle/1/1165/project_report_15scse101346_tathagat_sinha_pdf%20-%20Tathagat%20Sinha.pdf?sequence=1&isAllowed=y.
- [4] Md. Neaz Ahsan Chowdhury and Kanchon Sarker , "SMART MIRROR". <http://182.160.117.219:8080/xmlui/bitstream/handle/123456789/203/9.03.17%20%20Project%20%20CSE05006459%20%20CSE05006444%20%20Mohammad%20Manzurul%20Islam.pdf?sequence=1&isAllowed=y>.
- [5] Chris Rectenwald, Teague Kohlbeck, and Benny Richmond,"The SmartMirror"<http://seniordesign.ee.nd.edu/2017/Design%20Teams/smartmir/SmartMirrorFinalReport.pdf>
- [6] Lakshmi N M, Chandana M S, Ishwarya P, Nagarur Meena and Rajendra R Patil , " IoT based Smart Mirror using Raspberry Pi" , International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org NCESC - 2018 Conference Proceedings , Volume 6, Issue 13 , 2018
- [7] Dr. C.K. Gomathy, Mr. R. Venkata Narayana and Mr. T.Giridhar Reddy, "A SMART MIRROR USING RASPBERRY PI BASED ON IOT" ,International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072 , Volume: 08 Issue: 10 ,Oct 2021 .
- [8] Shreyansh Khale, Aditi Sathe, Rugveda Salunke, Shweta Nathan, Amit Maurya , "Smart Mirror",International Journal of Recent Technology and Engineering (IJRTE) ISSN: 2277-3878, Volume-8, Issue-2S11, September 2019.

- [9] P.Selvaraj , Lakshmi Kanthan Narayanan , “ IoT Enhanced Smart Mirror for Personal and Commercial Applications”, International Journal of Advanced Trends in Computer Science and Engineering, ISSN 2278-3091, Volume. 9 , No: 4 ,July - August , 2020.
- [10]Piyush Maheshwari , Maninder Jeet Kaur and Sarthak Anand , “Smart Mirror: A Reflective Interface to Maximize Productivity”, International Journal of Computer Applications (0975 – 8887) Volume 166 – No.9 , May 2017.

APPENDIX

A.SOURCE CODE

```
import datetime
import json
import os
import time
import tkinter.font
import traceback
from tkinter import *
from requests import get

import feedparser
import httpplib2
import requests
from PIL import ImageTk, Image
from apiclient import discovery
from forecastiopy import *
from oauth2client import client
from oauth2client import tools
from oauth2client.file import Storage

try:
    import argparse

    flags = argparse.ArgumentParser(parents=[tools.argparser]).parse_args()
except ImportError:
    flags = None

# Text size
x_large_text_size = 94
large_text_size = 48
medium_text_size = 28
small_text_size = 18
```

```

x_small_text_size = 14

# tk objects
root = Tk()
root.title('Smart Mirror')
root.configure(background='black')
screen_width = root.winfo_screenwidth()
screen_height = root.winfo_screenheight()

# Frames
frame_top = Frame(root, background='black')
frame_bottom = Frame(root, background='black')

frame_t_left = Frame(frame_top, background='black')
frame_calendar = Frame(frame_t_left, background='black')
frame_calendar_image = Frame(frame_calendar, background='black')
frame_calendar_events = Frame(frame_calendar, background='black')

frame_t_right = Frame(frame_top, background='black')
frame_weather = Frame(frame_t_right, background='black')
frame_current_high_low = Frame(frame_weather, background='black')

frame_forecast = Frame(frame_t_right, background='black')
frame_days = Frame(frame_forecast, background='black')
frame_days_icon = Frame(frame_forecast, background='black')
frame_temp_high = Frame(frame_forecast, background='black')
frame_temp_low = Frame(frame_forecast, background='black')

frame_b_left = Frame(frame_bottom, background='black')
frame_news = Frame(frame_b_left, background='black')
frame_newspaper = Frame(frame_b_left, background='black')

frame_b_right = Frame(frame_bottom, background='black')

```

```

root.geometry('{}x{}'.format(screen_width, screen_height))

# Fonts
font_time = tkinter.font.Font(family='Helvetica', size=x_large_text_size)
font_date = tkinter.font.Font(family='Helvetica', size=medium_text_size)
font_location = tkinter.font.Font(family='Helvetica', size=medium_text_size)
font_temperature = tkinter.font.Font(
    family='Helvetica', size=x_large_text_size)
font_quote = tkinter.font.Font(family='Helvetica', size=medium_text_size)
font_holiday = tkinter.font.Font(family='Helvetica', size=small_text_size)
font_weather = tkinter.font.Font(family='Helvetica', size=small_text_size)
font_news = tkinter.font.Font(family='Helvetica', size=x_small_text_size)
font_news_headlines = tkinter.font.Font(
    family='Helvetica', size=medium_text_size)

# Weather

# file_object.close()
units = ForecastIO.ForecastIO.UNITS_US
lang = ForecastIO.ForecastIO.LANG_ENGLISH

time_format = '%l:%M'
date_format = '%A, %B %d, %Y'

CLIENT_SECRET_FILE = 'client_secret.json'
APPLICATION_NAME = 'Google Calendar API Python Quickstart'

# Asset images with launch word
weatherIcons = {"cloudy": 'assets/Cloud.png',
                 "hail": 'assets/Hail.png',
                 "fog": 'assets/Haze.png',
                 "clear-night": 'assets/Moon.png',

```

```

        "partly-cloudy-night": 'assets/PartlyMoon.png',
        "partly-cloudy-day": 'assets/PartlySunny.png',
        "rain": 'assets/Rain.png',
        "snow": 'assets/Snow.png',
        "storm": 'assets/Storm.png',
        "clear-day": 'assets/Sun.png',
        "sunrise": 'assets/Sunrise.png',
        "tornado": 'assets/Tornado.png',
        "wind": 'assets/Wind.png'
    }
weather_data = []

# Images
image = Image.open("assets/Newspaper.png")
image = image.resize((25, 25), Image.ANTIALIAS)
image = image.convert('RGB')
photo = ImageTk.PhotoImage(image)

image_calendar = Image.open("assets/Calendar.png")
image_calendar = image_calendar.resize((25, 25), Image.ANTIALIAS)
image_calendar = image_calendar.convert('RGB')
photo_calendar = ImageTk.PhotoImage(image_calendar)

# Labels
weather_image_lg = Label(frame_weather, bg='black', fg='white')

label_temperature = Label(frame_weather, font=font_temperature,
                           bg='black',
                           fg='white')
label_location = Label(frame_weather, font=font_location,
                       bg='black',
                       fg='white')
label_current_temp_high = Label(

```

```
frame_current_high_low, bg='black', fg='white', font=font_holiday)
label_current_temp_low = Label(
    frame_current_high_low, bg='black', fg='white', font=font_holiday)
```

```
label_news_title = Label(frame_b_left, font=font_news_headlines,
    text="Headlines",
    bg='black',
    fg='white')
```

```
label_date = Label(frame_t_left, font=font_date,
    bg='black',
    fg='white')
label_clock = Label(frame_t_left, font=font_time,
    bg='black',
    fg='white')
```

```
# Layout Top Left
```

```
label_date.pack(side=TOP, anchor=W)
label_clock.pack(side=TOP, anchor=W)
```

```
# Layout Top Right
```

```
label_location.pack(side=TOP)
weather_image_lg.pack(side=LEFT, anchor=E)
label_temperature.pack(side=LEFT, anchor=E)
```

```
label_current_temp_high.pack(side=LEFT, anchor=E)
label_current_temp_low.pack(side=RIGHT, anchor=E)
```

```
# Layout Bottom Left
```

```
label_news_title.pack(side=TOP, anchor=W)
```

```
# Clock function
```

```

def tick():
    s = time.strftime(time_format)
    d = time.strftime(date_format)
    if s != label_clock["text"]:
        label_clock["text"] = s
    if d != label_date["text"]:
        label_date["text"] = d
    label_clock.after(200, tick)

# Displays Location name and weather data for location
def current_weather():
    for w in frame_temp_high.winfo_children():
        w.destroy()
    for t in frame_temp_low.winfo_children():
        t.destroy()
    for q in frame_days_icon.winfo_children():
        q.destroy()
    for y in frame_days.winfo_children():
        y.destroy()
    for items in weather_data:
        weather_data.remove(items)

    # Location request
    location_req_url = "http://api.ipstack.com/%s?access_key=%s" % (get_ip(),
ip_api_key)
    t = requests.get(location_req_url)
    location_obj = json.loads(t.text)
    latitude = location_obj['latitude']
    longitude = location_obj['longitude']
    weather_url = "https://api.darksky.net/forecast/%s/%s,%s?lang=%s&units=%s" %
(weather_api_key, latitude, longitude,
                                lang, units)

```

```

location = "%s, %s" % (location_obj['city'], location_obj['region_code'])

# Sets Weather Object
r = requests.get(weather_url)
weather_obj = json.loads(r.text)
weather_data.append(weather_obj)
for data in weather_data:

    fahrenheit = int(data['currently']['temperature'])
    icon_id = data['currently']['icon']

    current_data = data['daily']['data']

    for today in current_data[0:1]:

        temperature_high_today = int(today['temperatureHigh'])
        temperature_low_today = int(today['temperatureLow'])

        if temperature_high_today != label_current_temp_high and
temperature_low_today != label_current_temp_low:
            label_current_temp_high['text'] = format(
                temperature_high_today, '.0f') + "°" + "/"
            label_current_temp_low['text'] = format(
                temperature_low_today, '.0f') + "°"

    if icon_id in weatherIcons and weather_image_lg['image'] != icon_id:
        icon2 = weatherIcons[icon_id]
        w_icon = Image.open(icon2)
        w_icon = w_icon.resize((100, 100), Image.ANTIALIAS)
        w_icon = w_icon.convert('RGB')
        w_photo = ImageTk.PhotoImage(w_icon)

        weather_image_lg.configure(image=w_photo)

```

```

weather_image_lg.icon = w_photo

format(fahrenheit, '.0f')
if fahrenheit != label_temperature["text"]:
    label_temperature["text"] = format(fahrenheit, '.0f') + "°"

if location != label_location["text"]:
    label_location["text"] = location

week_days = data['daily']['data']
for day in week_days[1:6]:

    label_days = Label(frame_days, bg='black',
                       fg='white', font=font_holiday)
    label_days_icon = Label(frame_days_icon, bg='black', fg='white')
    label_temp_high = Label(
        frame_temp_high, bg='black', fg='white', font=font_holiday)
    label_temp_low = Label(frame_temp_low, bg='black',
                           fg='white', font=font_holiday)

    daily_icon_id = day['icon']

    if daily_icon_id in weatherIcons and label_days_icon['image'] != daily_icon_id:
        icon3 = weatherIcons[daily_icon_id]
        daily_icon = Image.open(icon3)
        daily_icon = daily_icon.resize((30, 30), Image.ANTIALIAS)
        daily_icon = daily_icon.convert('RGB')
        daily_photo = ImageTk.PhotoImage(daily_icon)

        label_days_icon.configure(image=daily_photo)
        label_days_icon.icon = daily_photo

```

```

days_time_conversion = day['time']
time_object = datetime.datetime.fromtimestamp(days_time_conversion)
time_object_format = time_object.strftime('%a')

```

```

label_days['text'] = time_object_format

```

```

daily_temperature_high = int(day['temperatureHigh'])
daily_temperature_low = int(day['temperatureLow'])
if label_current_temp_high['text'] != daily_temperature_high \
    and label_current_temp_low['text'] != daily_temperature_low:
    label_temp_high['text'] = format(
        daily_temperature_high, '.0f') + "°" + "/"
    label_temp_low['text'] = format(daily_temperature_low, '.0f') + "°"

```

```

label_days.pack(side=TOP, anchor=W)
label_days_icon.pack(side=TOP, anchor=W)
label_temp_high.pack(side=TOP, anchor=W)
label_temp_low.pack(side=TOP, anchor=W)

```

```

label_temperature.after(600000, current_weather)

```

```

def get_news():
    for widget in frame_news.winfo_children():
        widget.destroy()
    for i in frame_newspaper.winfo_children():
        i.destroy()
    try:
        feed = feedparser.parse(google_news_url)

        for post in feed.entries[0:5]:
            newspaper_image = Label(frame_newspaper, bg='black', fg='white')
            newspaper_image.configure(image=photo)

```

```

newspaper_image.icon = photo

label_news = Label(frame_news, bg='black',
                    fg='white', font=font_news)
label_news['text'] = post['title']
newspaper_image.pack(side=TOP, anchor=W)
label_news.pack(side=TOP, anchor=W)
frame_news.after(600000, get_news)

except Exception as e:
    traceback.print_exc()
    print("Error: %s. Cannot get news." % e)

def get_ip():
    try:
        ip = get('https://api.ipify.org').text
        return ip
    except Exception as e:
        traceback.print_exc()
        return "Error: %s. Cannot get ip. " % e

def get_credentials():
    """Gets valid user credentials from storage.
    If nothing has been stored, or if the stored credentials are invalid,
    the OAuth2 flow is completed to obtain the new credentials.
    Returns:
        Credentials, the obtained credential.
    """
    home_dir = os.path.expanduser('~')
    credential_dir = os.path.join(home_dir, '.credentials')
    if not os.path.exists(credential_dir):

```

```

    os.makedirs(credential_dir)
    credential_path = os.path.join(credential_dir,
                                   'calendar-python-quickstart.json')

    store = Storage(credential_path)
    credentials = store.get()
    if not credentials or credentials.invalid:
        flow = client.flow_from_clientsecrets(CLIENT_SECRET_FILE, SCOPES)
        flow.user_agent = APPLICATION_NAME
        if flags:
            credentials = tools.run_flow(flow, store, flags)
        else: # Needed only for compatibility with Python 2.6
            credentials = tools.run(flow, store)
        print('Storing credentials to ' + credential_path)
    return credentials

def get_calendar():
    """Shows basic usage of the Google Calendar API.
    Creates a Google Calendar API service object and outputs a list of the next
    5 events on the user's calendar.
    """
    for widget in frame_calendar_events.wininfo_children():
        widget.destroy()
    for i in frame_calendar_image.wininfo_children():
        i.destroy()
    credentials = get_credentials()
    http = credentials.authorize(httplib2.Http())
    service = discovery.build('calendar', 'v3', http=http)

    now = datetime.datetime.utcnow().isoformat() + 'Z' # 'Z' indicates UTC time

    eventsResult = service.events().list(

```

```

calendarId='primary', timeMin=now, maxResults=10, singleEvents=True,
orderBy='startTime').execute()
events = eventsResult.get('items', [])

if not events:
    print('No upcoming events found.')
for event in events[0:5]:
    label_calender_image = Label(
        frame_calendar_image, bg='black', fg='white')
    label_calender = Label(frame_calendar_events,
        font=font_news, bg='black', fg='white')

    label_calender_image.configure(image=photo_calendar)
    label_calender_image.icon = photo_calendar

    if event['start'].get('dateTime') is None and event['end'].get('dateTime') is None:
        start_date = event['start'].get('date')
        start_date_format = datetime.datetime.strptime(
            start_date, "%Y-%m-%d")
        start = datetime.datetime.strptime(start_date_format, "%b, %d")
        label_calender['text'] = str(start) + ": " + event['summary']
    else:
        end_date_time = event['end'].get('dateTime')
        end_date_time_obj = datetime.datetime.strptime(
            ".join(end_date_time.rsplit(':', 1)), "%Y-%m-%dT%H:%M:%S%z")
        end_date_time_format = datetime.datetime.strptime(
            end_date_time_obj, "%l:%M%p")

        start_date_time = event['start'].get('dateTime')
        start_date_time_obj = datetime.datetime.strptime(
            ".join(start_date_time.rsplit(':', 1)), "%Y-%m-%dT%H:%M:%S%z")
        start_date_time_format = datetime.datetime.strptime(
            start_date_time_obj, "%b, %d %l:%M%p")

```

```

label_calender['text'] = start_date_time_format + \
    "-" + end_date_time_format + ": " + event['summary']

label_calender.pack(side=TOP, anchor=W)
label_calender_image.pack(side=TOP, anchor=W)
label_calender.after(600000, get_calendar)

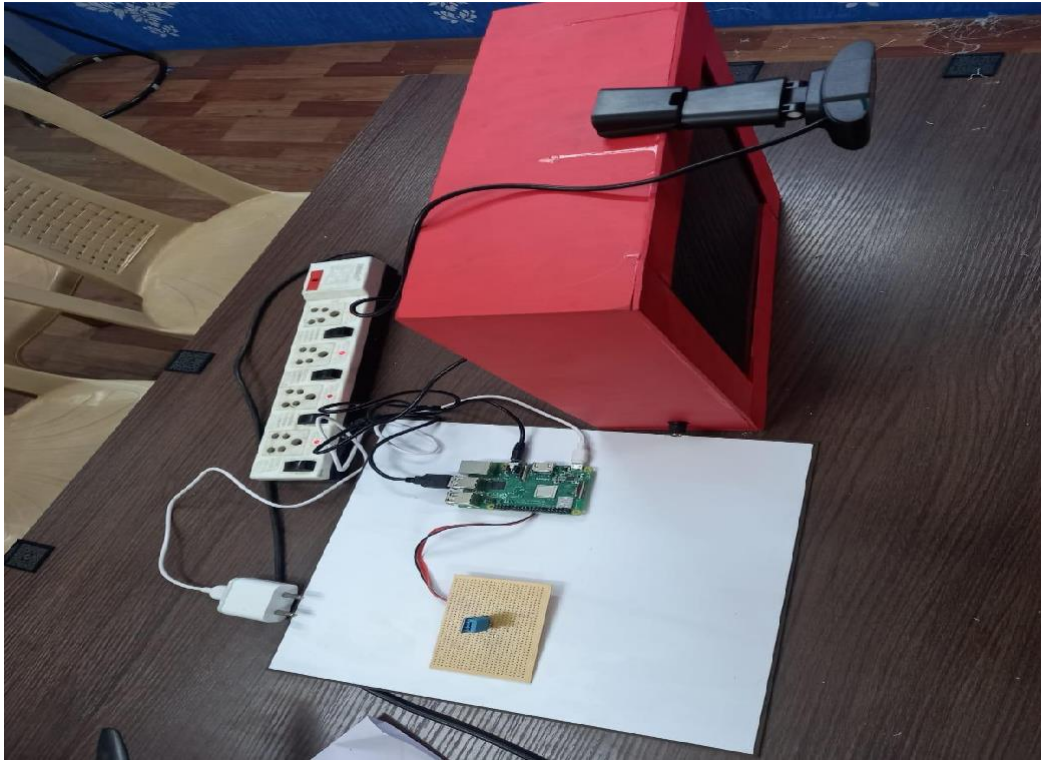
tick()
current_weather()
get_news()
get_calendar()

frame_t_left.pack(side=LEFT, anchor=N, padx=40, pady=40)
frame_calendar.pack(side=TOP, anchor=W)
frame_calendar_events.pack(side=RIGHT, anchor=N)
frame_calendar_image.pack(side=LEFT, anchor=N)

frame_t_right.pack(side=RIGHT, anchor=N, padx=40, pady=40)
frame_weather.pack(side=TOP, anchor=N)
frame_current_high_low.pack(side=LEFT, anchor=W)
frame_forecast.pack()
frame_days.pack(side=LEFT, anchor=S)
frame_days_icon.pack(side=LEFT, anchor=N)
frame_temp_high.pack(side=LEFT, anchor=S)
frame_temp_low.pack(side=LEFT, anchor=S)
frame_b_left.pack(side=BOTTOM, anchor=W, padx=40, pady=40)
frame_news.pack(side=RIGHT, anchor=W)
frame_newspaper.pack(side=LEFT, anchor=W)
frame_top.pack(expand=TRUE, fill=BOTH, side=TOP)
frame_bottom.pack(expand=TRUE, fill=BOTH, side=BOTTOM)
root.mainloop()

```

B. SCREENSHOTS



Inner look of magic mirror



Out look of the magic mirror



Smart mirror displays date,time,news

C. RESEARCH PAPER

MAGIC MIRROR - A Smart Reflective Interface for a Smart Home

R. M. Gomathi
Associate Professor
Department of IT
Sathyabama Institute of
Science and Technology
Chennai, India
gomathi.it@sathyabama.ac.in

Kante Lakshminarayana
UG Scholar
Department of IT
Sathyabama Institute of
Science and Technology
Chennai, India
kantelakshminarayana21@gmail.com

Maligi Reddy Karthik Reddy
UG Scholar
Department of IT
Sathyabama Institute of
Science and Technology
Chennai, India
karthikreddymaligireddy437@gmail.com

Abstract—Our day begins with the reality that we study our picture inside the reflection. This project depicts the layout and improvement of the application. Our day begins with the reality that we take a look at our image in reflection. Magic Mirror or Smart Mirror is an IoT utility in which we are able to use micro-controllers embedded inside the Raspberry Pi replicate, so one can show the essential facts along with time, date, news, events and more, and act as a regular reflect. . These glass have now programmed the usage of artificial intelligence which makes the work treasured. The primary goal is therefore to expand a product this is cost effective and efficient sufficient to meet all of the basic wishes of regular people. These glasses are actually programmed using synthetic intelligence which makes the paintings treasured. The essential goal is consequently to broaden a product that is reasonably-priced and green sufficient to meet all the primary desires of ordinary human beings.

Keywords—Smart Mirror, Raspberry Pi, IOT

I. INTRODUCTION

In this world absolutely everyone desires a secure lifestyles. Modern guy has invented various technology for his cause. In today's global, human beings need to be connected and need clean get admission to records. Whether through television or the Internet, humans need to be knowledgeable and up to date on modern occasions round the sector. That lifestyles itself is a lot of time is made. Our work is

based at the idea that we all appearance inside the mirror whilst we depart the house, so the replicate isn't always clever and magical. The Internet of Things refers to the interconnection through the Internet of computing gadgets embedded in common gadgets, allowing them to ship and get hold of statistics. The Internet of Things, with its titanic boom, expands its software in the human surroundings, turning the house right into a smart domestic. Home automation has been around for a long term, and it's turning the home right into a clever unit with the goal of creating the home extra cozy and green. Some common packages are lighting automation, smart thermostats, alarms, and window blinds (Home Automation, 2013). A smart domestic is a connected domestic that connects all varieties of digital gadgets to communicate with every different through the Internet.

II. LITERATURE SURVEY

Suryansh Chandel, Ashai Mandwarya and S. Ushasuhanya et al [1] The voice performance of the reflect of the continent wall. This is a tool which can have a reflect and an interactive display on the equal time, even as showing the multimedia content material of time, climate, day, weather and news. It includes diverse features including real records and facts replace, voice commands, face detection/reputation using LCD display, microphone

and webcam.

Tathagat Sinha et al.[2] A reflect that runs on a common structure and has all of the features required by using the user. Gary Chu and Kyle Stadelmann et al.[3] It changed into established as a hard work replicate from the whole. They have created an open source 1 and consistent with day module permitting every person with a Smart Mirror to file their own 1 second videos per day. Thirdly, they have got additionally created mod patch notes to maintain Over watch gamers updated with the cutting-edge sport updates.

Neaz Ahsan Chowdhury and Kanchon Sarker et al [4] created an less expensive Smart Mirror tool that human beings should interact with and even in addition boost the era. There were several problems with the design and software of the physical a part of the task, such as the reflect not being reflective sufficient, but these failures have been resolved thru in addition assessments and trials within the in addition improvement of the Smart Mirror.

Chris Rektenwald, Teague Colbeck, Benny Richmond, and Mike Schafer et al, [5] have given someone easy access to the web page to get all the records which could have an effect on the practise for the brand new day. Through using an LCD display and a two-manner reflect, it indicates the climate, time and date and shows the news. In addition, the consumer-friendly interface reachable from any Wi-Fi-enabled tool lets in the user to easily installation a domestic Wi-Fi connection, alternate in which to acquire climate reviews, and pick out the source from which to obtain information. .

Lakshmi N.M., Chandana M.S., Ishwarya P., Nagarur Meena and Rajendra R. Patil et al. [6] described the layout and improvement of a smart mirror, which is an stylish interface used to view statistics and also to stumble on burglars at home. Dr. S.K. Gomati, Mr. R. Venkata Narayana and Mr. T. Giridhar Reddy et al. [7] advanced a machine to

paintings on extra functions to broaden time, weather, temperature, and weather information. The system is primarily based on IoT-primarily based intelligence to facilitate and boom consumer recognition.

Sadeta KULOVIC and Belma RAMIK-BRKICH et al, proposed a common mirror turned into advanced, which is used as a mirror and a tool. The statistics within the mirror is to be had at any time. Cost analyzes benefit from good sized cost financial savings throughout consequences compared to to be had off-the-shelf answers [8].

III. EXISTING SYSTEM

A smart home is a related home that connects all forms of digital devices to communicate with every other through the Internet. Many gadgets had been evolved with a view to revolutionize the manner we paintings with the concepts of multimedia communicate, artificial intelligence, and the Internet of Things (IoT). Our numerous day by day activities at home, in the office, or even in production. The concept of having an interactive replicate which could respond on your instructions is thrilling to anyone. The Magic Mirror aims to supplement the simple reflexive with built-in intelligence to perform commonplace each day tasks consisting of reading the newspaper, family tree updates, climate forecasting, and so on., and provide all this information to the person while getting ready. The multifunctional mirror that can be both a decoration and a source of facts. The hassle of finding places for new homes was too critical. This product gives a technique to the aforementioned troubles. It additionally acts as an progressive and appealing item that can't be located everywhere. One of the hazards of Magic Mirror technology is the shortage of human interaction. Of path, customers spend a lot time on-line these days that they can keep on my own.

IV. PROPOSED MODEL

According to analyze, there are many specific and thrilling tactics to constructing a pointy reflect. Other solutions are available in the market place, others are just prototypes or do-it-yourself (DIY) projects developed through fanatics for their very own use.

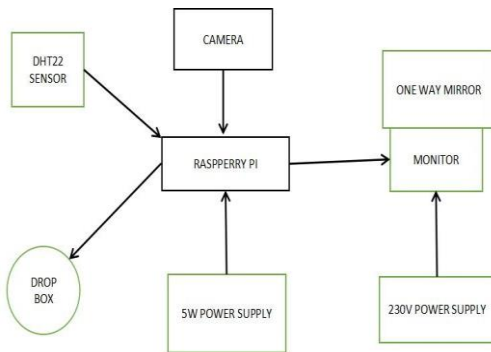


Fig. 1. Working of Components

From the above fig.1. all the components are inter linked with each other and each and every component will play a major roll while implementing. The proposed methodologies which have been used so far, the usage of the Raspberry Pi microprocessor to show all multimedia content material and as a everyday reflect. This merchandise are made with the aid of well-known companies and brands and had been within the marketplace for a long term, but only a few humans are acquainted with clever mirrors and might afford them. Many of those smart devices have the simple requirement of social media and connectivity to render them. The main functions commonplace to these kind of clever mirrors are climate document, date and time display, document reports, sound or track guide.

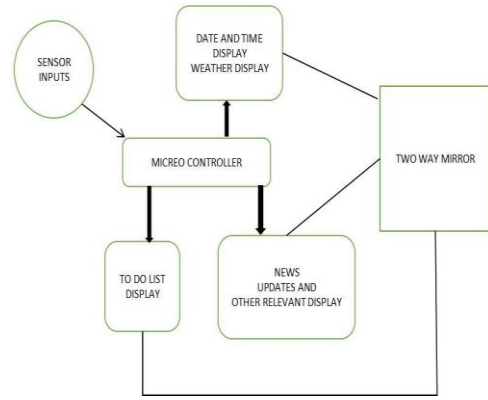


Fig. 2. Block Diagram of Magic Mirror

From the above Fig.2. the mirror displays all the content like Date and Time, Display, Weather Display, News Update and Other Relevant Display.

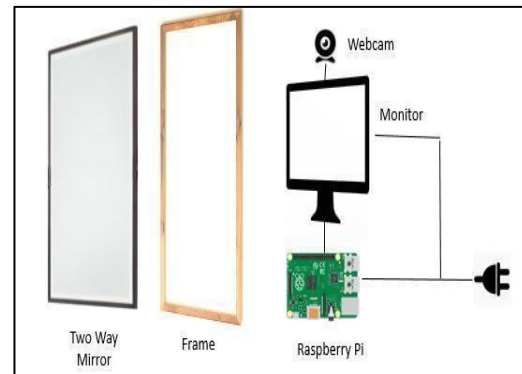


Fig. 3. Outlook of Smart Mirror

V. FUNCTIONING OF PROPOSED METHODOLOGY

A proposed approach of handing over the principle features required for a mirror characteristic for a painful replicate. The need for this gadget might be an opening in which technology and the normal factors of human life would converge and provide a miles higher experience. This technique more often than not goals to create a cheap and efficient smart tool that is feasible. There are aaditional sensors established and connected with a micro controller to obtain numerous inputs which includes temperature, air excellent, and so forth, which also can be displayed at the shelves.

VI. ADVANTAGES OF PROPOSED SYSTEM

- The proposed method is simple to enforce and

almost the whole thing is available in present methodologies.

- Micro controllers are incredibly reasonably-priced to apply, in contrast to the Raspberry Pi used in maximum methods.
- The circuit connections are simple and also can be finished as a DIY assignment, although most of the connections encompass micro-controllers and show monitors.

VII. RASPBERRY PI CONTROLLER



Fig. 4. Raspberry pi

The Raspberry Pi is a cheaper, small card-sized laptop as shown in above fig.4 that plugs into a laptop screen or TV and uses a trendy keyboard and mouse. It's a small application gadget that lets in human beings of every age to explore computing and learn to application in languages like Scratch and Python. It can do everything you'll assume from a computer laptop, from surfing the internet and playing HD video to developing spreadsheets, word processing and video games.

VIII. DHT22 SENSOR

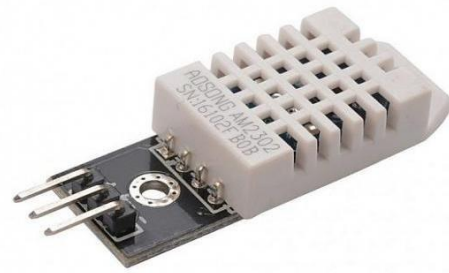


Fig. 5.DHT22 sensor

The DHT22 virtual temperature and humidity sensor module is a fundamental low value virtual temperature and humidity sensor. It makes use of a capacitive humidity sensor and a thermistor to measure the ambient air and output a digital sign on a facts pin (no longer requiring analog input pins).It is simple to apply, however requires care for facts series. The only real downside to this sensor is that you may handiest get new information from it as soon as each 2 seconds, so when the usage of our library, sensor readings can be up to two seconds obsolete, the second one pin to the records input pin, and the proper pin to ground.

IX. WEB CAMERA



Fig. 6. web cam

- Built-in macro sensitivity and first-class CMOS photo sensor.
- Interpolated image decision up to 25 megapixels with 6 light sensors; Image resolution

sixteen MP; USB interface; Nightvision ; USB cable period: 1m; Focus range: 4 cm to infinity

- Image controls for color saturation, brightness, sharpness and clarity are adjustable
- Snapshot switch to take pictures. Focus range from four cm to infinity
- Anti-flicker 50Hz, 60Hz or extra
- Image nice: RGB24 or I420

X. RESULT

Raspberry Pi, DHT22 Sensor, Web camera, Two way mirror, Connectors were used in the development of the Magic Mirror. When an person looks in to a mirror ,the magic mirror will display the Date, Time, News and Weather. The mirror will get respond by using the web camera and send request to the raspberry pi and the content will get displays on the mirror.



Fig. 7. Mirror with hardware components

From the above fig.7 the components which was used for preparing the magic mirror we can see. The connections are visible how they are connected through each other.



Fig. 8. Magic Mirror attached with Webcam

From the figure 8 the outlook of the magic mirror can we see and how it looks to see from outside.



Fig. 9. Magic Mirror showing News, Date and Time

XI. CONCLUSION

The proposed methodology for the use of a unique micro controller and growing a pointy reflect for the usage of the same Raspberry Pi, which has already been implemented or utilized by all other existing methodologies. Our major intention is to expand this assignment the use of primary products available which can be extremely realistic and green, in addition to the vast use of the Raspberry Pi microprocessor. We need the idea to be designed and made value effective so that every ordinary person can have enough money it. The clever mirror venture may be easily extended to a few other environments like telephones. In the destiny, this mirror can be used as a smart home network with lighting fixtures gadgets, virtual assistants, TV, tune systems, lions, etc., which may be integrated collectively. This is a actual pain at home.

REFERENCES

- [1] Suryansh Chandel, Ashay Mandwarya and S.Ushasukhanya, "IMPLEMENTATION OF MAGIC MIRROR USING RASPBERRY PI 3", International Journal of Pure and Applied Mathematics , ISSN: 1314-3395, Volume 118, No. 22 2018, 451-455, July 09, 2020.
- [2] Tathagat Sinha, " SMART MIRROR USING RASPBERRY PI " . May , 2020.
- [3] Gary Chew and Kyle Stadelmann , "Smart Mirror" <https://kastner.ucsd.edu/ryan/wp-content/uploads/sites/5/2020/06/admin/smartmirror.pdf>.
- [4] Md. Neaz Ahsan Chowdhury and Kanchon Sarker , "SMART MIRROR". <http://182.160.117.219:8080/xmlui/bitstream/handle/123456789/203/9.03.17%20-%20Project%20-%20CSE05006459%20-%20CSE05006444%20-%20Mohammad%20Manzurul%20Islam.pdf?sequence=1&isAllowed=y>
- [5] Chris Rectenwald, Teague Kohlbeck, and Benny Richmond , " The SmartMirror". [http://seniordesign.ee.nd.edu/2017/Design%20Teams/smartmir/SmartMirrorFinal Report.pdf](http://seniordesign.ee.nd.edu/2017/Design%20Teams/smartmir/SmartMirrorFinal%20Report.pdf)
- [6] Lakshmi N M, Chandana M S, Ishwarya P, Nagarur Meena and Rajendra R Patil , " IoT based Smart Mirror using Raspberry Pi" , International Journal of Engineering Research & Technology (IJERT) ISSN: 2278-0181 Published by, www.ijert.org NCESC - 2018 Conference Proceedings , Volume 6, Issue 13 , 2018
- [7] Dr. C.K. Gomathy, Mr. R. Venkata Narayana and Mr. T.Giridhar Reddy, "A SMART MIRROR USING RASPBERRY PI BASED ON IOT" ,International Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395-0056, p-ISSN: 2395-0072 , Volume: 08 Issue: 10 ,Oct 2021 .
- [8] Sadeta KULOVIC and Belma RAMIC-BRKIC , " DIY Smart Mirror" DOI: 10.1007/978-3-319-71321-2_31.

