

Conference Hall Automation System using Python-kivy Application

M V Rajesh, B. Venkateswara Rao

Abstract— Conference halls contain a variety of technology tools that allows one to host productive meetings. The developed python-kivy based automation system allows all of these technologies to work in unison and be controlled through one Graphical User Interface based device. It can control electrical equipment like lights, fans, sound system and projector, and also controls the presentation slides by taking input from a touch screen. This device makes conferences/seminars more productive and efficient.

Keywords: Automation; Graphical User Interface; Conference Halls

I. INTRODUCTION

Electricity and Time are some of the most important resources in every ones life. With the growing human population, there is a much increase in load which in turn increases the consumption of electricity. Thus, it is essential to save electrical energy [1, 2]. Most of the electricity is wasted due to negligence of people to turn off lights, fans, AC's and other electrical equipment when they leave the room. By adopting the automation both time and electrical energy are saved. Everyday there are many seminars/conferences conducted around the world, in every one of them time is an important factor and also lot of electrical energy is wasted. With the use of modern technology both time and electrical energy are saved [3]. Kivy is one of the latest graphical user interface toolkit, Which permits one to easily make natural interfaces for a wide selection of devices. It has elegant built-in support for multi touch devices [4, 5]. It is the only viable way to code in Python on mobile and low power consuming devices like raspberry pi. It replaces the horrible APIs of past days graphical interfaces like tkinter, HTML & CSS. Conference hall automation system can be developed in raspberry pi by using python and kivy [6, 7]. This system provides very advanced interface to the user to control presentation slides and electrical equipment like lights, fans, sound system and projector.

II. MATHEMATICAL MODEL OF THE SYSTEM

Input Set:

Conference hall automation system takes input via LCD touch screen. So, the set of inputs will be,

$I = \{ \text{All touch inputs from the LCD touch Screen} \}$

Output Set:

The output for the quantified inputs above will be persistent by the raspberry pi conferring to the inputs

Out1 = {speech, display, text}

Out2 = {GUI, application response}

Therefore, $Out = Out1 \cup Out2$

Output Out = {Reply for matching input via GUI, application reaction}

Here is one to many relationships among input & output sets.

III. METHODOLOGY

This project emphases on emerging a control board which has a Graphical User Interface on an LCD touch screen. The main control panel is designed by a Raspberry pi [8, 9] and a 7-inch LCD touch screen. The Raspberry pi takes inputs from GUI on touch screen in order to control the functions of conference hall automation system by using relay circuit and an open source office software called LibreOffice. The raspberry pi is connected to the projector in order to open and control the presentations. The relay control device is designed so that it abides the conference halls electrical criteria. The system features an Arduino Mega board with core AVR microcontroller (AT mega 2560) interfaced to the Bluetooth module (HC-05) and power relay interface in order to receive the commands from raspberry pi and to control all the electrical equipment [10-12].

GUI is created by using kivy, decisions based on the inputs is taken by the python program and sends signals to corresponding devices. The block diagram of the conference hall automation system is displayed in the figure 1.

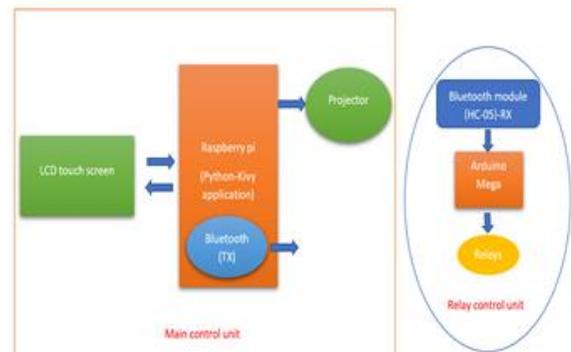


Fig. 1. Block diagram of conference halls automation scheme

Manuscript published on 30 December 2018.

* Correspondence Author (s)

M V Rajesh, UG student, Department of EEE, V R Siddhartha Engineering College, Kanuru, Andhra Pradesh, India. (E-mail: m.vrk@outlook.com)

B. Venkateswara Rao, Associate Professor, Department of EEE, V R Siddhartha Engineering College, Kanuru, Andhra Pradesh, India. (E-mail: drbvrao@vrsiddhartha.ac.in)

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an open access article under the CC-BY-NC-ND license <https://creativecommons.org/licenses/by-nc-nd/4.0/>

IV. SOFTWARE DESIGN

The software design includes

- A. GUI unit
- B. Processing unit
- C. Control unit

A. GUI (Graphical User Interface) unit:

GUI is a program interface that takes advantage of the raspberry pi's graphics capabilities to make the conference hall automation system easier to use. Well-designed graphical user interfaces are easy to understand and use as shown in below figure 2 which is designed by using kivy programming language.

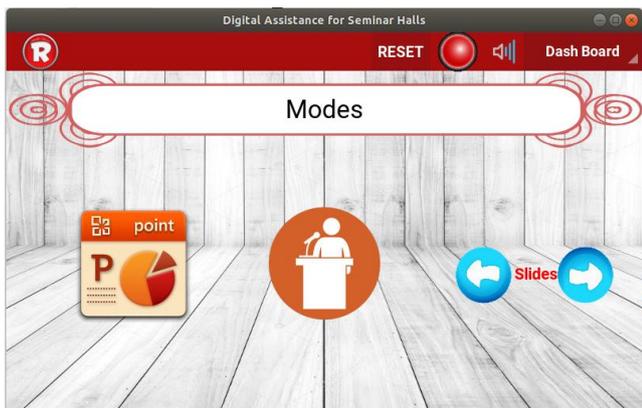


Fig 2. Graphical User Interface of conference halls Automation system

There are three steps in creating the above application in kivy, they are

1. sub-classing the App class
2. applying its RootWidget() method so it returns a Widget instance (the root of the GUI widget tree)
3. Instantiating this class, and calling its run () method.

All the three steps mentioned above are implemented as shown in the below figure 3.

```
class MainApp(App):
    def build(self):
        rw=RootWidget()
        Clock.schedule_interval(rw.update_GUI,0.5)
        return rw

if __name__ == '__main__':
    MainApp().run()
```

Fig 3. Screenshot of code

The RootWidget () method is implemented with various functions in order to get different graphical components like toggle buttons, switches, sliders etc as shown in below figures.



Fig 4. Welcome Message

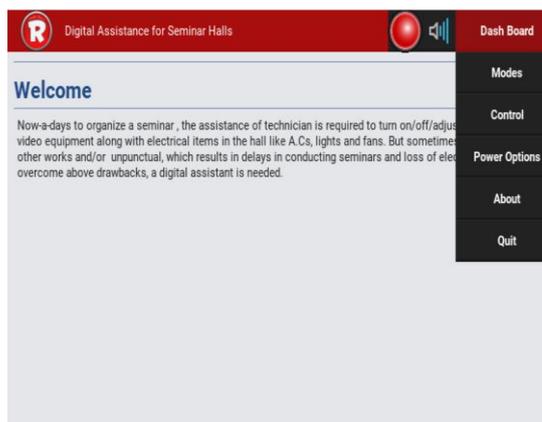


Fig .5 Various options of dash board to control



Fig. 6. Control panel options to control the hardware and PPT

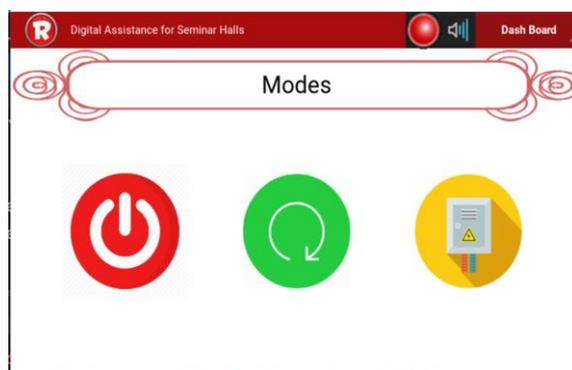


Fig 7. Various power options

The GUI will be updated for every 0.5sec. fig 8 shows that If there is an event of touch i.e. if a button is pressed then it will call the switch() function which is implemented in python program, then the processing unit will take the decisions.

```
Switch:
id:s11
size_hint_y: None
size_hint_x: None
pos:Window.size[0]*.05, Window.size[1]*.43
on_active:rw.switch(self,self.active,'s11')
Switch:
id:s12
size_hint_y: None
size_hint_x: None
pos:Window.size[0]*.05, Window.size[1]*.30
on_active:rw.switch(self,self.active,'s12')
```

Fig 8. Code for displaying two switches on LCD touch screen

The flow chart of the GUI unit is shown below figure 9.

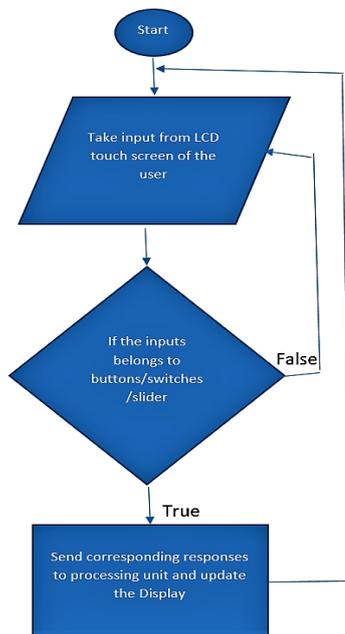


Fig 9. Flowchart of GUI

GUI consists of different modes, control options and power options. They are Presentation mode, lecture mode Normal mode, different switches for different electrical equipment and shutdown, restart, mains OFF buttons as shown in figures 2, 6, 7 respectively.

B. Processing unit:

The processing unit performs different operations based on different inputs from LCD touch screen they are:

1. Controls slides, sound and electrical equipment's like lights, fans, AC etc. by taking input from touch screen and sends commands to relay unit.
2. Automatically adjusts the lights and fans according to modes.
3. Automatically setups the projector and sound system.
4. Automatically searches for the PPT file and puts it in full screen mode on the projector screen.

V. RESULTS & PROTOTYPE

Figures 10, 11 and 12 are the pictures of developed model.



Fig. 10 LCD touch screen and main control unit placed on the podium front view



Fig. 11 LCD touch screen and main control unit placed on the podium back view

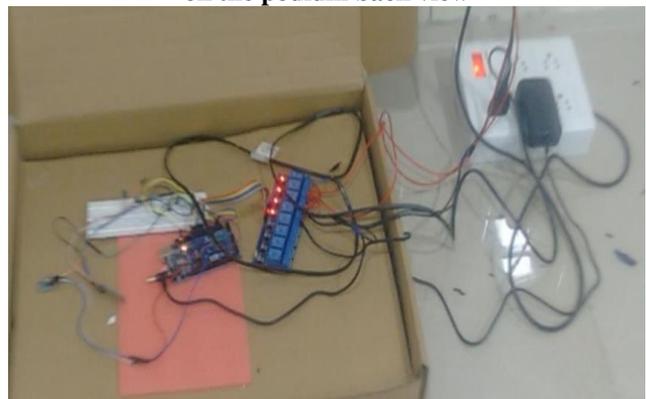


Fig. 12 relay controlling unit having Arduino mega, Bluetooth module and relay module.

VI. CONCLUSION

The Developed device is a low cost, reliable and scalable control device it can replaces computers, laptops and security systems in seminar halls. It saves lot of time and electrical energy.

ACKNOWLEDGMENT

“The author/s thankfully acknowledge(s) the financial support provided by The Institution of Engineers (India) and V R Siddhartha Engineering College for carrying out Research & Development work in this subject”.

REFERENCES

1. Elaine Rich, Kelvin Knight, Shiva Shankar B-Nair “Artificial Intelligence”, MC Graw Hill Publication, 2013.
2. Raspberry Pi [online], Available at <http://www.raspberrypi.org/>.
3. M.N. Chowdhury, M.S. Nooman, S. Sarker ‘Access Control of Door and Home Security by Raspberry Pi Through Internet’, Int. J. Sci. Eng. Res. 2013, 4, 550–558.
4. M V Rajesh, B. Venkateswararao, “Artificial Intelligence Based Machine Learning Assistance for Self-Driving Car using Raspberry Pi”, IJARSE, Vol. 6, Iss 11, November 2017, pp. 1718-1724.
5. Vidyasagar K, Balaji G, Narendra R K, “Android Phone Enabled Home Automation”, JAIR, ISSN: 2278-5213, Vol. 4, Iss 2 July 2015.
6. N Radhamani , “Intelligent Home Automation & Security System”, IJISSET, Vol.2 no. 6, pp. 639-644, June 2015.
7. Babu Sooraj S., “Intelligent Home Automation”, IJRET, Vol.4 no.3, pp. 32-34, April 2015.
8. Panth Sharon, “Designing Home Automation System using Java ME for Mobile Phone”, IJECSE, Vol.2 No.2, pp. 798-807, April 2011
9. Arduino, Available at <http://www.arduino.cc>
10. Praveen Kumar S, Rajesh K, Subrahmanya S , Kantharaju A G, Vanishree Moji , “ Seminar Hall Automation Using Raspberry Pi 3 ” International Journal of Technological Research & Innovative Solutions (IJTRIS) , 2017
11. M. Hariprasath, P.K. Karthick “Intelligent Conference Hall Automation System” IJARECE, Vol 5, Iss.12 2016, pp.2543-2547.
12. M V Rajesh, B Venkateswara Rao, P Sai Vamsi Krishna and S Pavan Kumar, “Raspberry PI based Digital Assistant for Seminar Halls (D.A.S.H)”, International Conference on New Trends in Engineering & Technology (ICNTET)-2018, Organized by GRT Institute of Engineering and Technology, Chennai, during September 07th & 08th 2018.