Institute of Engineering and Technology (IET)

Implementing and testing a FSM
Project report

PREPARED BY

Abhishek Arora (2020BTechCSE002)

Mohak Goyal (2020BTechCSE049)

12th January, 2021

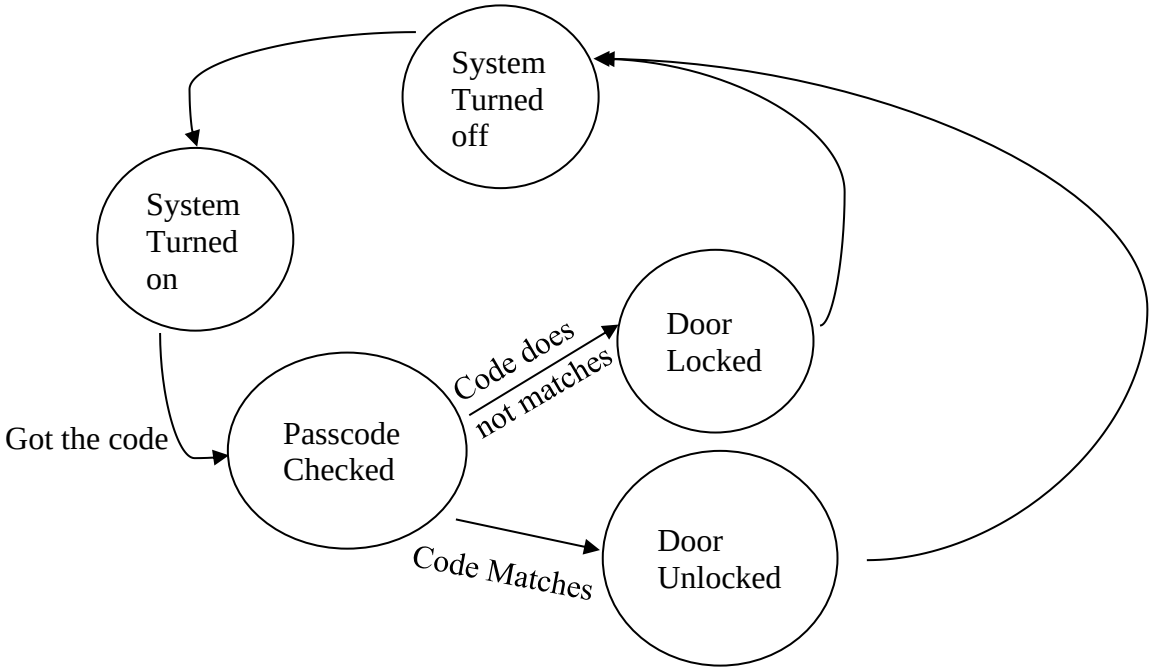
TABLE OF CONTENTS

PROBLEM STATEMENT

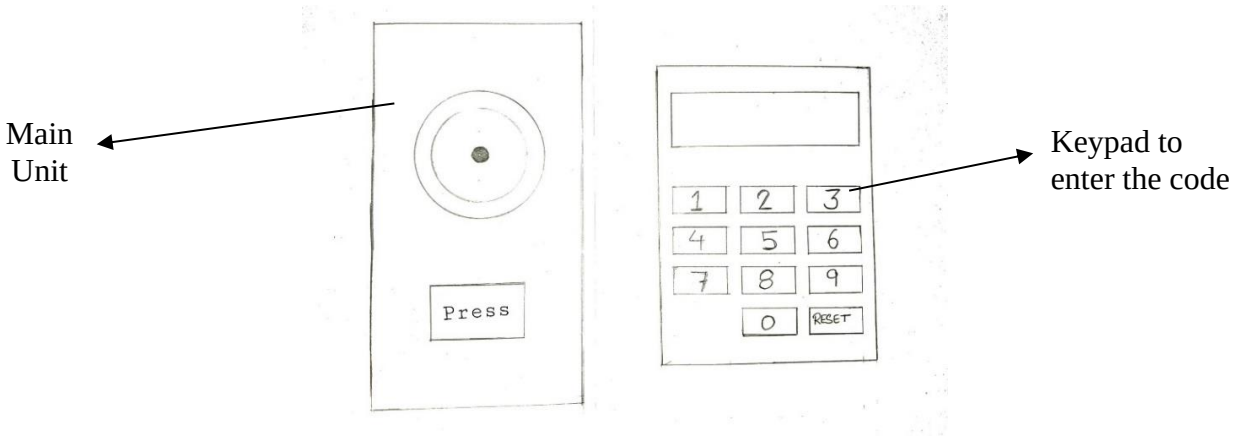
- FSM (Finite State Machine) description

This is a Home Security System in which you have to enter the correct code to enter into the house. In this system, first of all the main unit with camera will be displayed and the state will be 'System Turned off' then the user will click the button on it and the state will be changed to 'System Turned on'. The user will be asked for a code after that, when user enters the code the programme will check the code and after that the state will be changed to 'Passcode Checked'. If the code matches, the state will be changed to 'Door Unlocked', the door will allow the user to enter but if code does not match the state will be 'Door Locked' and door will remain locked.

- State diagram



- GUI (Graphic User Interface) Sketch



CODE

```
#importing of all modules needed
import tkinter
import tkinter as tk
from tkinter import *
from tkinter import messagebox
import tkinter.font as font
from tkinter import messagebox
from PIL import Image, ImageTk
from playsound import playsound
from statemachine import StateMachine, State

#defination of statemachine class
class HomeSecuritySystem(StateMachine):
    sysTurnedOff = State('System Turned Off', initial=True)
    sysTurnedOn = State('System Turned On')
    passChecked = State('Passcode Checked')
    locked = State('Door Locked')
    unlocked = State('Door Unlocked')

    turnOn = sysTurnedOff.to(sysTurnedOn)
    checkCode = sysTurnedOn.to(passChecked)
    codeMatched = passChecked.to(unlocked)
    codeNotMatched = passChecked.to(locked)
    sysOff1 = unlocked.to(sysTurnedOff)
    sysOff2 = locked.to(sysTurnedOff)

securitySystem = HomeSecuritySystem()

#Using the TK() class
main_window = Tk()

main_window.geometry("300x400")
main_window.resizable(False,False)
main_window.title("Home Security System")

#creating image label
image1 = Image.open("D:/Users/mohak/Downloads/Home Security System/CAMERA.png")
camera = ImageTk.PhotoImage(image1)

label1 = Label(main_window, image=camera)
label1.image = camera
label1.place(x=50, y=8)

#created function for later use
def message_box():
    message_window = Tk()
    message_window.geometry("250x60")
    message_window.title("Error")
    error = Label(message_window, text="Please use the keypad provided!!", fg = "red
", font = ("Times New Roman", 12))
    error.place(x = 20, y = 15)

    message_window.mainloop()

#command for press button
def press_action():
    playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3')
```

```
57.
58.     #state changed for the first time
59.     securitySystem.run('turnOn')
60.     print(securitySystem.current_state)
61.
62.     #using the same class Tk() for creating another window
63.     keypad = Tk()
64.     keypad.geometry("300x320")
65.     keypad.resizable(False, False)
66.     keypad.title("Enter Code")
67.
68.     #command for button1 in keypad window
69.     def button1_action():
70.         playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3')
71.         try:
72.             code = int(code_text.get())
73.             code_text.delete(0, END)
74.             number = (code*10)+1
75.             code_text.insert(tk.END, number)
76.         except:
77.             code_text.insert(tk.END, 1)
78.
79.     #command for button2 in keypad window
80.     def button2_action():
81.         playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3')
82.         try:
83.             code = int(code_text.get())
84.             code_text.delete(0, END)
85.             number = (code*10)+2
86.             code_text.insert(tk.END, number)
87.         except:
88.             code_text.insert(tk.END, 2)
89.
90.     #command for button3 in keypad window
91.     def button3_action():
92.         playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3')
93.         try:
94.             code = int(code_text.get())
95.             code_text.delete(0, END)
96.             number = (code*10)+3
97.             code_text.insert(tk.END, number)
98.         except:
99.             code_text.insert(tk.END, 3)
100.
101.        #command for button4 in keypad window
102.        def button4_action():
103.            playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
104.            try:
105.                code = int(code_text.get())
106.                code_text.delete(0, END)
107.                number = (code*10)+4
108.                code_text.insert(tk.END, number)
109.            except:
110.                code_text.insert(tk.END, 4)
111.
112.        #command for button5 in keypad window
113.        def button5_action():
```

```
114.             playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
115.             try:
116.                 code = int(code_text.get())
117.                 code_text.delete(0, END)
118.                 number = (code*10)+5
119.                 code_text.insert(tk.END, number)
120.             except:
121.                 code_text.insert(tk.END, 5)
122.
123.         #command for button6 in keypad window
124.         def button6_action():
125.             playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
126.             try:
127.                 code = int(code_text.get())
128.                 code_text.delete(0, END)
129.                 number = (code*10)+6
130.                 code_text.insert(tk.END, number)
131.             except:
132.                 code_text.insert(tk.END, 6)
133.
134.         #command for button7 in keypad window
135.         def button7_action():
136.             playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
137.             try:
138.                 code = int(code_text.get())
139.                 code_text.delete(0, END)
140.                 number = (code*10)+7
141.                 code_text.insert(tk.END, number)
142.             except:
143.                 code_text.insert(tk.END, 7)
144.
145.         #command for button8 in keypad window
146.         def button8_action():
147.             playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
148.             try:
149.                 code = int(code_text.get())
150.                 code_text.delete(0, END)
151.                 number = (code*10)+8
152.                 code_text.insert(tk.END, number)
153.             except:
154.                 code_text.insert(tk.END, 8)
155.
156.         #command for button9 in keypad window
157.         def button9_action():
158.             playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
159.             try:
160.                 code = int(code_text.get())
161.                 code_text.delete(0, END)
162.                 number = (code*10)+9
163.                 code_text.insert(tk.END, number)
164.             except:
165.                 code_text.insert(tk.END, 9)
166.
167.         #command for button0 in keypad window
```

```
168.        def button0_action():
169.            playsound('D:/Users/mohak/Downloads/Home Security System/Beep Sound1.mp3'
    )
170.            try:
171.                code = int(code_text.get())
172.                code_text.delete(0, END)
173.                number = (code*10)+0
174.                code_text.insert(tk.END, number)
175.            except:
176.                code_text.insert(tk.END, 0)
177.
178.        #command for reset button in keypad window
179.        def reset_action():
180.            code_text.delete(0, END)
181.
182.        #command for submit button in keypad window
183.        def submit_action():
184.
185.            #changing state 2nd time
186.            securitySystem.run('checkCode')
187.            print(securitySystem.current_state)
188.
189.            #to get the code entered in int form and remove all the errors
190.            try:
191.                x = int(code_text.get())
192.            except:
193.                #using the function created above
194.                message_box()
195.
196.            #condition for checking the input
197.            if(x==8921):
198.
199.                #changing state 3rd time
200.                securitySystem.run('codeMatched')
201.                print(securitySystem.current_state)
202.
203.                #using the class Tk() for crating new window
204.                result_window = Tk()
205.
206.                result_window.geometry("300x400")
207.                result_window.resizable(False,False)
208.                result_window.title("Home Security System")
209.
210.                #creating labels to place in resullt winodw
211.                image1 = Image.open("D:/Users/mohak/Downloads/Home Security System/CA
    MERA.png")
212.                camera = ImageTk.PhotoImage(image1)
213.
214.                label1 = Label(result_window, image=camera)
215.                label1.image = camera
216.                label1.place(x=50, y=8)
217.
218.                result = Label(result_window, text = "Access Granted", fg = "white",
    bg = 'green')
219.                result.configure(font=("Bookman Old Style", 20))
220.                result.place(x = 50, y = 275)
221.
222.                #closing of keypad and opening another window
223.                keypad.destroy()
```

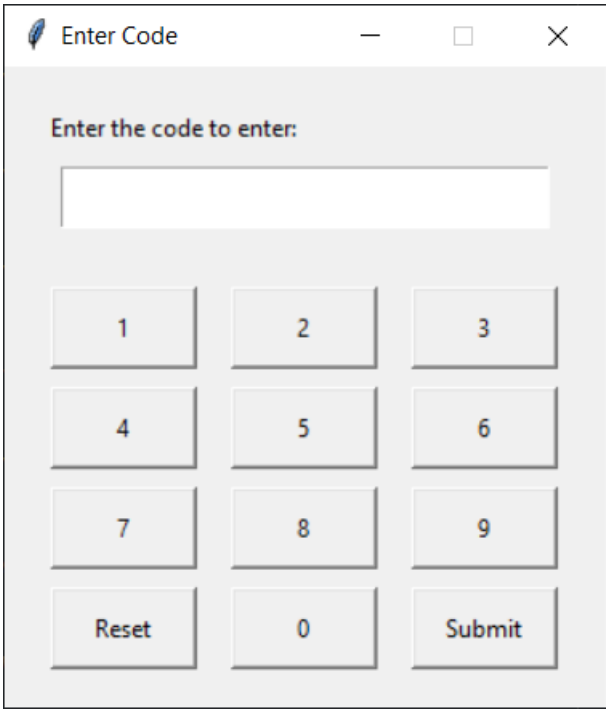
```
                result_window.mainloop()

                #changing state for 4th time and resetting state
                securitySystem.run('sysOff1')
                print(securitySystem.current_state)
            else:

                #changing state 3rd time
                securitySystem.run('codeNotMatched')
                print(securitySystem.current_state)

                #using the class Tk() for crating new window
                result_window = Tk()

                result_window.geometry("300x400")
                result_window.resizable(False,False)
                result_window.title("Home Security System")

                #creating labels to place in resullt winodw
                image1 = Image.open("D:/Users/mohak/Downloads/Home Security System/CAMERA.png")
                camera = ImageTk.PhotoImage(image1)

                label1 = Label(result_window, image=camera)
                label1.image = camera
                label1.place(x=50, y=8)

                result = Label(result_window, text = "Access Denied", fg = "white", bg = 'red')
                result.configure(font=("Bookman Old Style", 20))
                result.place(x = 50, y = 275)

                #closing keypad window and opening another window
                keypad.destroy()
                result_window.mainloop()

                #if above condition fails then this condition will run and state will change here 4th time aand resetting state here
                securitySystem.run('sysOff2')
                print(securitySystem.current_state)


   #creation and placing of all the labels, buttons and entry widgets
        label2 = Label(keypad, text = "Enter the code to enter:")
        label2.place(x = 20, y = 20)

        code_text = Entry(keypad, show = "*", font = ('Times New Roman',18))
        code_text.place(x = 28, y = 50)
        code_text.insert(tk.END, "")

        button1 = Button(keypad, text = "1", height = 2, width = 9, command = button1_action)
        button1.place(x = 23, y = 110)

        button2 = Button(keypad, text = "2", height = 2, width = 9, command = button2_action)
        button2.place(x = 113, y = 110)

```

```
277.       button3 = Button(keypad, text = "3", height = 2, width = 9, command = button3
    _action)
278.       button3.place(x = 203, y = 110)
279.
280.       button4 = Button(keypad, text = "4", height = 2, width = 9, command = button4
    _action)
281.       button4.place(x = 23, y = 160)
282.
283.       button5 = Button(keypad, text = "5", height = 2, width = 9, command = button5
    _action)
284.       button5.place(x = 113, y = 160)
285.
286.       button6 = Button(keypad, text = "6", height = 2, width = 9, command = button6
    _action)
287.       button6.place(x = 203, y = 160)
288.
289.       button7 = Button(keypad, text = "7", height = 2, width = 9, command = button7
    _action)
290.       button7.place(x = 23, y = 210)
291.
292.       button8 = Button(keypad, text = "8", height = 2, width = 9, command = button8
    _action)
293.       button8.place(x = 113, y = 210)
294.
295.       button9 = Button(keypad, text = "9", height = 2, width = 9, command = button9
    _action)
296.       button9.place(x = 203, y = 210)
297.
298.       reset = Button(keypad, text = "Reset", height = 2, width = 9, command = reset
    _action)
299.       reset.place(x = 23, y = 260)
300.
301.       button0 = Button(keypad, text = "0", height = 2, width = 9, command = button0
    _action)
302.       button0.place(x = 113, y = 260)
303.
304.       submit = Button(keypad, text = "Submit", height = 2, width = 9, command = sub
    mit_action)
305.       submit.place(x = 203, y = 260)
306.
307.   #closing main window and opening next window
308.       main_window.destroy()
309.       keypad.mainloop()
310.
311.   press_button = Button(main_window, text = "Press", command = press_action)
312.   press_button.configure(font=("Bookman Old Style", 20))
313.   press_button.place(x = 100, y = 275)
314.
315.   #printing the current state at initial state
316.   print(securitySystem.current_state)
317.
318.   mainloop()
```

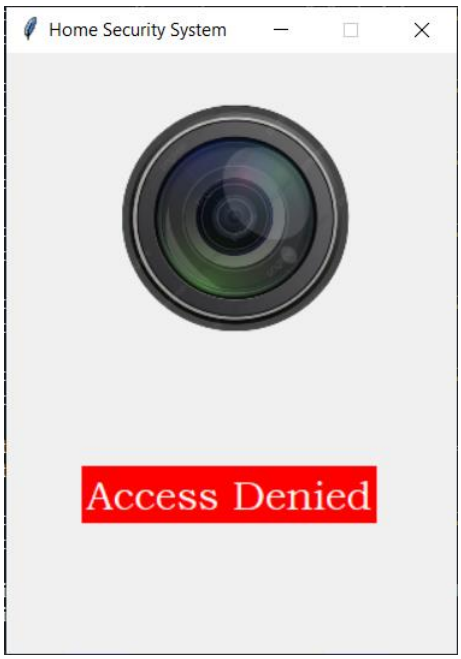
RESULTS AND DISCUSSION

First we will approach the main window

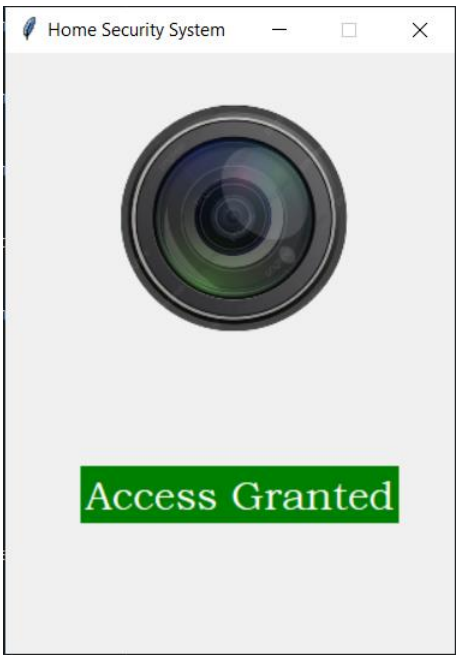
Now after pressing the “Press” button



If we enter the wrong code the door will remain locked and message will be displayed “ACCESS DENIED”



If right code is entered then the door will be unlocked and a message will be displayed ”ACCESS GRANTED”



REFERENCES

https://www.geeksforgeeks.org/

https://stackoverflow.com/

https://www.w3schools.com/

https://github.com/

https://pythonbasics.org/