

capstone design

rc car controlling through internet

made by :- engineer mohamed abd el mohsen saleh

fresh graduate electronics and communication engineer at Alexandria university ,Egypt

for more information you can contact me at these websites:-

mohamed20163858@gmail.com

<https://www.facebook.com/mohamed.abdelmohsen.9655>

<https://twitter.com/mohamed20163858>

introduction:-

Nowadays, internet has a great impact to our life and ensure communications between all countries in the world as it is rarely for us to see a country that doesn't establish an internet connection yet. due to this fact this inspire me to design this project that have many applications as i will illustrate :-

behavior of the system:-

my project is simply try to control and drive an rc car that have already a camera to do vision applications(like take photos ,face detection,etc...) and router or modem to connect rc car to internet and control it using any usual and daily device u want to use (smart phone , tablet ,laptop,..etc).

(very important note :-the techniques used here in the system can be applied to any complex mechanical devices rather than that rc car example ,quad copter , helicopter,...etc and the only reason to choose rc car is it has a very simple mechanism that make me to illustrate the system idea that i implement and test it without even buying expensive mechanical devices although choosing quad copter will be more fit to

this device but if you understand the idea you will realize you can implement it with any mechanical system you want)

benefits of this project:-

1- you can use it to explore the world between us ,say you live in France and want to see your family and your home and land it will become easy for you to control say a quad copter and take a pictures for family ,house ,sky ,farm,...etc in any position you want., you can use it to explore danger places that very danger for human to explore like bottom of the sea (using ROV in this case),exploring mines(using both rc car and quad copter),.....etc

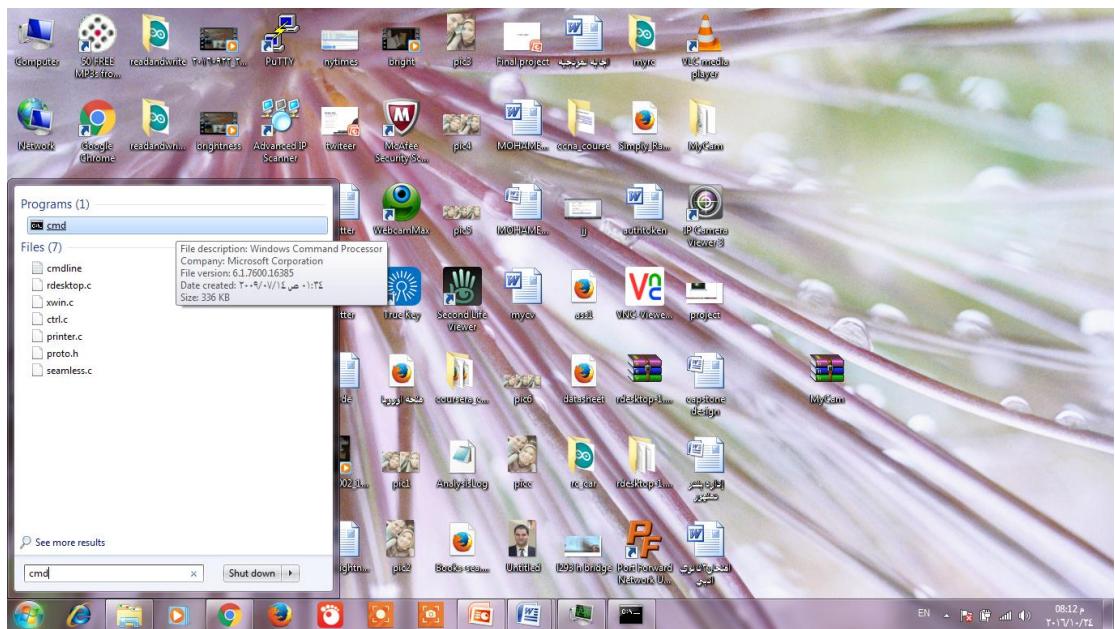
2- it will be actually an open source project as you see there are tons of applications that we can apply this project to it ,so it will be very high space to add components that target to a specified application.

user perspective:-

first you should do some setup configuration to make device work properly .these steps are:-

1-do port forward process for the modem or router connected to your device. this step is very important that will enable you to access your device through internet (i will assume you use windows in your pc or laptop and the device you connected is raspberry pi).

a-connect your router or modem to your PC or laptop then open your command windows (assuming using windows OS)



b- write this command **ipconfig** to get your gateway ip

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 6.1.7600]
Copyright <c> 2009 Microsoft Corporation. All rights reserved.

C:\Users\ALHAMD COMPANY>ipconfig

Windows IP Configuration

Wireless LAN adapter Wireless Network Connection 3:

Connection-specific DNS Suffix . . . .
IPv6 Address . . . . . : fd24:69a5:de0d:d600:4126:1761:d18e:b41b
Temporary IPv6 Address . . . . . : fd24:69a5:de0d:d600:e585:69f0:2df:4bed
Link-local IPv6 Address . . . . . : fe80::4126:1761:d18e:b41b%17
IPv4 Address . . . . . : 192.168.1.2
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : 192.168.1.1

Wireless LAN adapter Wireless Network Connection:

Media State . . . . . : Media disconnected
Connection-specific DNS Suffix . . . .

Ethernet adapter Local Area Connection:
```

c-now reconnect your router or modem to your device (assuming it is raspberry pi) then try to get the device ip address(using sudo ifconfig in case of using raspbian OS)

```
pi@raspberrypi ~ $ sudo ifconfig
eth0      Link encap:Ethernet HWaddr b8:27:eb:05:82:24
          inet addr:192.168.1.10 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fd24:69a5:de0d:d600:c830:e9c4:2f8:8936/64 Scope:Global
            inet6 addr: fe80::b2e8:29cd:9b:cf75/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:390 errors:0 dropped:0 overruns:0 frame:0
              TX packets:613 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:54509 (53.2 KiB) TX bytes:335648 (327.7 KiB)

lo       Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:3153 errors:0 dropped:0 overruns:0 frame:0
            TX packets:3153 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:33879791 (32.3 MiB) TX bytes:33879791 (32.3 MiB)

pi@raspberrypi ~ $
```

very important note:- in case of using portable router (that can handle many devices at same time) you can download ip scanner and you can easily get your device ip address .

d- open your browser and type the gateway ip address and write your user name and password (you can get it from your router user guide)

e- now you easily access your router and get the device ip address

now you should first get a static ip address for your device ,second do port forward process.

to able to do both configuration you should get your router or modem model name then use Google to do these processes .

for getting static ip address for your device write this phrase in google:-
(your router name) Address reservation .

in my case my router name is HG532e so after doing some google research i do that

1- get my device mac address from device itself or router(in case of using Raspberry pi)

```

pi@raspberrypi ~ $ sudo ifconfig
eth0      Link encap:Ethernet HWaddr b8:27:eb:05:82:24
          inet addr:192.168.1.10 Bcast:192.168.1.255 Mask:255.255.255.0
          inet6 addr: fd24:69a5:de0d:d600:c830:e9c4:2f8:8936/64 Scope:Global
            inet6 addr: fe80::b2e8:29cd:9b:cf75/64 Scope:Link
              UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
              RX packets:390 errors:0 dropped:0 overruns:0 frame:0
              TX packets:613 errors:0 dropped:0 overruns:0 carrier:0
              collisions:0 txqueuelen:1000
              RX bytes:54509 (53.2 KiB) TX bytes:335648 (327.7 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
            UP LOOPBACK RUNNING MTU:65536 Metric:1
            RX packets:3153 errors:0 dropped:0 overruns:0 frame:0
            TX packets:3153 errors:0 dropped:0 overruns:0 carrier:0
            collisions:0 txqueuelen:0
            RX bytes:33879791 (32.3 MiB) TX bytes:33879791 (32.3 MiB)

pi@raspberrypi ~ $

```

Etisalat ADSL Router

Device Type	IP Address	MAC Address	Remaining Lease Time
Computer	192.168.1.2	EC:08:6B:11:37:D7	21 hours 6 minutes 14 seconds
Computer	192.168.1.3	84:55:A5:96:3F:D3	18 hours 30 minutes 21 seconds
Computer	192.168.1.4	BC:75:74:59:24:B8	18 hours 39 minutes 28 seconds
Computer	192.168.1.10	b8:27:eb:05:82:24	22 hours 57 minutes 13 seconds

Waiting for 192.168.1.1...

Copyright © Huawei Technologies Co., Ltd. 2012-2013. All rights reserved.

2- do address reservation for my device

DHCP

Index	MAC address(AA:BB:CC:DD:EE:FF)	IP address(XXX.XX)
—	B8:27:E6:05:82:24	192.168.1.10

DHCP Option Pool

Device type:	STB
DHCP Option pool:	<input checked="" type="checkbox"/>
Start IP address:	0.0.0.0
End IP address:	0.0.0.0
DHCP relay:	<input type="checkbox"/> Enable

as you here copy and paste mac address device in mac address field then type the static ip address you want to make your device have always have it. you should now reboot to make your changes take effect(for raspberry pi raspian OS use this command: sudo re

for port forward process go to

<https://portforward.com/router.htm>

and choose your router type and name then follow the instructions for mine

Port Mapping

Mapping Name	Interface	Protocol	Remote Host	External Start Port	External End Port	Internal Port	Internal Host	Enable	Remove
rasp-connection	INTERNET_TR069_R_0_35	TCP/UDP		3389	3389	3389	192.168.1.10	<input checked="" type="checkbox"/>	<input type="checkbox"/>

Settings

Type:	<input checked="" type="radio"/> Customization	<input type="radio"/> Application
Interface:	INTERNET_TR069_R_0_35	
Protocol:	TCP/UDP	
Remote host:		
External start port:	3389	
External end port:	3389	
Internal host:	192.168.1.10	
Internal port:	3389	
Mapping name:	rasp-connection	

as you see here i type the static ip that i enable for my raspberry pi

then i choose port 3389 to access this device from internet as this port is designed for remote desktop(the program that we will use) communications.

https://en.wikipedia.org/wiki/List_of_TCP_and_UDP_port_numbers

2- now we will connect to our raspberry pi through internet

first of all you should create a new gmail account this account will use it to receive your current public(external) router ip address or future ip changes as we cannot get a static public ip address so we need to notify if there is any change in public ip (public ip address changes every time you log in your router (power it up) so suppose your router isnot stable and lose its connection to internet it will send you the new public ip address when it reconnect to internet again)

write sudo nano (script name).py

it should be save at its default location /home/pi

write this script and press ctrl+x then y then Enter

```
import urllib2
from urllib2 import urlopen
import time
def internet_on():
    try:
        urllib2.urlopen('http://216.58.192.142', timeout=600)
        return True
    except urllib2.URLError as err:
        return False
def send_email(sender, pwd, receiver, message):
    import smtplib
    try:
        server = smtplib.SMTP("smtp.gmail.com", 587)
        server.ehlo()
        server.starttls()
        server.login(sender,pwd)
        server.sendmail(sender,receiver ,message )
        server.close()
        print 'successfully sent the mail'
    except:
        print "failed to send mail"
#sender=input("enter your email:")
#pwd=input("enter your password:")
#receiver=input("enter the email of the person you want to send him a message:")
#sender="mohamedmohsen211994@gmail.com"
#pwd="*****"
receiver="the same"
```

```

if receiver == "the same":
    receiver=sender
while True:
    old_ip= urlopen('http://ip.42.pl/raw',timeout=float('inf')).read()
    message="\nthis is your currently public ip:\n"+old_ip
    send_email(sender,pwd,receiver,message)
while internet_on()==True:
    time.sleep(1)
    print(True)
while internet_on()==False:
    time.sleep(1)
    print(False)

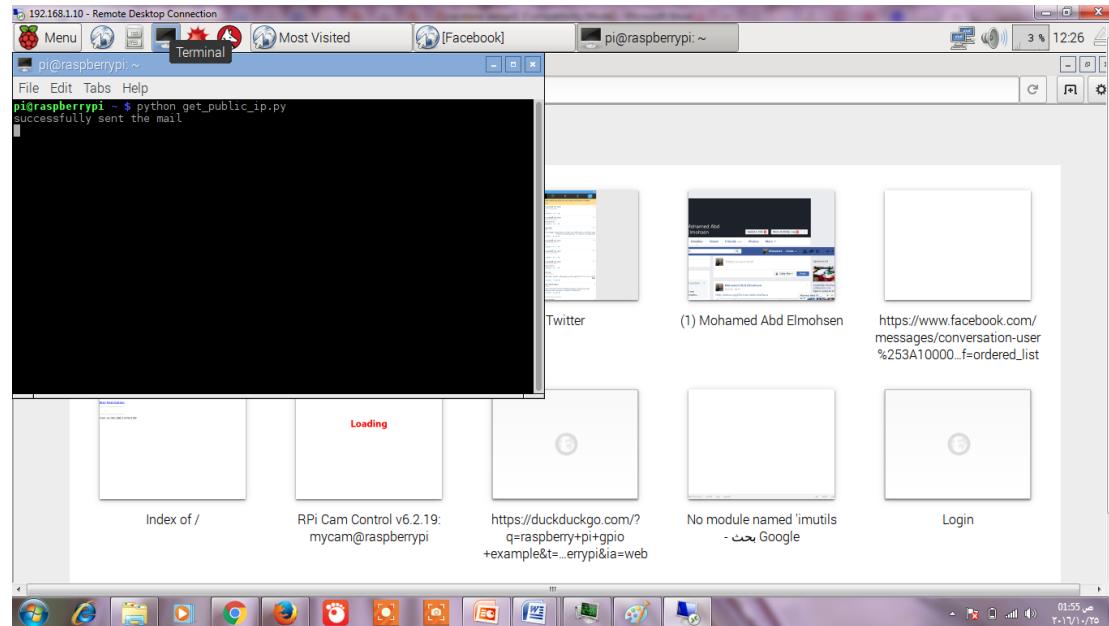
```

all this script do is to email you your email you your public ip address after you enter your email and password then if the public ip changes it will automatic send you your new ip so it is highly recommended

to run this script when you connect raspberry pi to your local home router not its router all you should to do is

open terminal in your raspberry pi then type this command

python (script name).py(donot close this terminal ever) you should open another terminal and let that execution terminal



then remove the connection of your pi to local home router and then connect the raspberry pi portable router or modem to it.

it will automatic send you the new public ip address .

3-now all we are to do is to install some necessary packages using this commands

```
sudo apt-get install xrdp
```

```
sudo apt-get install arduino
```

```
sudo apt-get install
```

```
git clone  
https://github.com/silvanmelchior/RPi_Cam_Web_Interface.git  
cd RPi_Cam_Web_Interface  
chmod u+x *.sh  
.install.sh
```

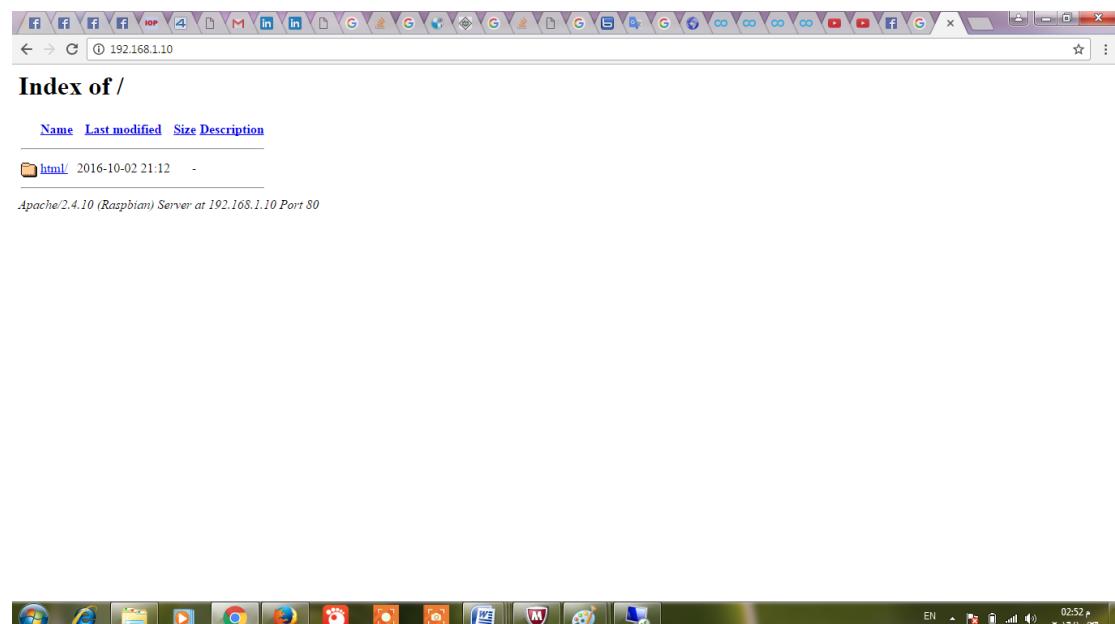
and the last command is for install necessary package to enable raspberry pi camera .

now if you open the browser and type this address

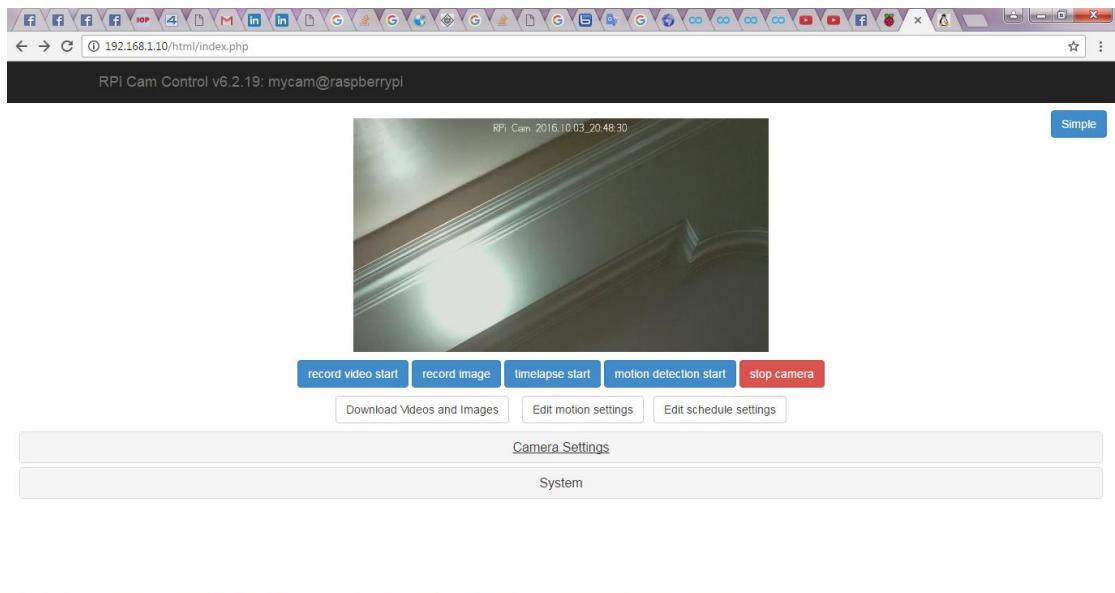
(raspberry pi address) then Enter

which in my case is static and the address is:-

192.168.1.10 i will get this result:-



then if i open html folder i will find this result:-



you can check more data about this in this website

<http://elinux.org/RPi-Cam-Web-Interface>

for ordinary usb web camera use these commands in terminal:-

```
sudo apt-get install fswebcam -y
sudo apt-get install ffmpeg -y
```

note:- if you use raspian jessie ffmpeg will not be installed so you should follow these instruction on this link:-

<https://www.assetbank.co.uk/support/documentation/install/ffmpeg-debian-squeeze/ffmpeg-debian-jessie/>

`sudo apt-get install mplayer -y`

`sudo apt-get install motion -y`

`sudo apt-get install scrot -y`

after that we should modify motion files a little bit

first open motion text file through this command

`sudo nano /etc/default/motion`

```
GNU nano 2.2.6          File: /etc/default/motion

# set to 'yes' to enable the motion daemon
start_motion_daemon=no
```

change start motion daemon from no=>yes.

then open motion config.text file through this command

`sudo nano /etc/motion/motion.conf`

now we will change four things in this file

```
# Daemon
#####
# Start in daemon (background) mode and release terminal (default: off)
daemon on
```

change daemon off =>on

```
# Image width (pixels). Valid range: Camera dependent, default: 352
width 640

# Image height (pixels). Valid range: Camera dependent, default: 288
height 480
```

change width to 640 and height to 480

(note width and height means the width and height of the frame
image change them to any values that you are comfortable to
them)

```
# Restrict stream connections to localhost only (default: on)
stream_localhost off
# Limits the number of images per connection (default: 0 = unlimited)
# Number can be defined by multiplying actual stream rate by desired number of seconds
# Actual stream rate is the smallest of the numbers framerate and stream_maxrate
stream_limit 0
```

change stream _localhost from on=>off

note:- to easy change these values press ctrl +w then type the
above 4 words to change easily these values.

now you should press ctrl+x then write y and press Enter to save
changes in motion.conf file.

now to make these changes takes effect write this command in
the Terminal.

sudo reboot.

now if we open the browser and enter the static ip address of raspberry
pi then this port 8081 in this formula

(static raspberry pi address:8081) and in my case:-

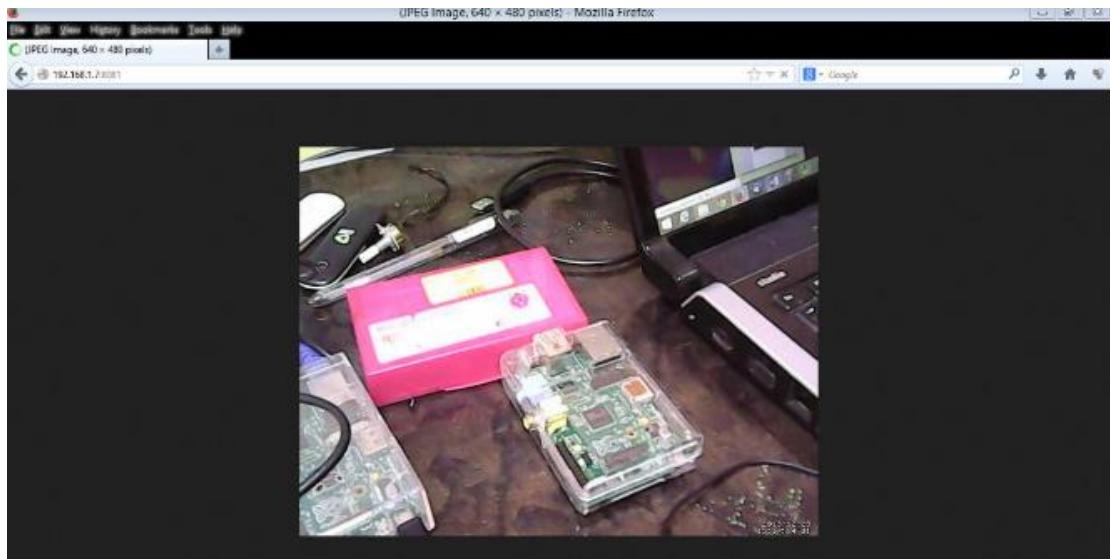
192.168.1.10:8081

if not work open your terminal and type:-

sudo service motion start

then reload your page

this is the picture we will get:-



now:- after we download all the packages we want lets get started
first of all:-

hardware components:-

1-raspberry pi model b

2- arduinio uno

3- DC and micro servo motors

4- body for the rc car

5- portable internet connection(modem or router)

6- portable USB charger

setup:-

1- you should open your raspberry pi using one of the three methods :-

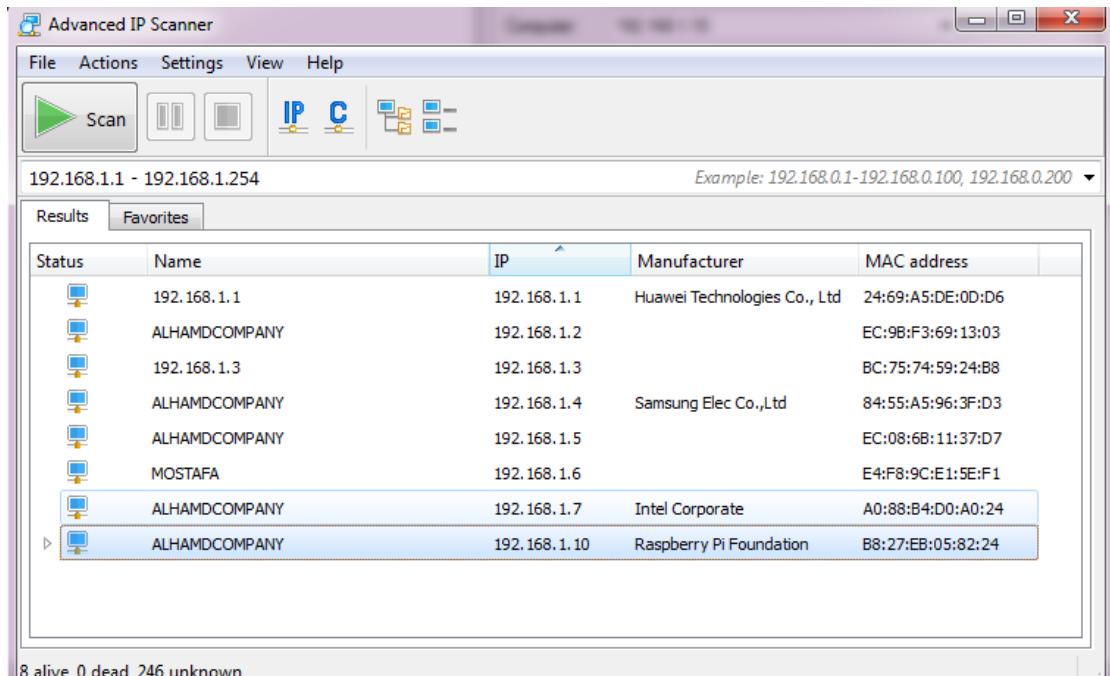
a) using keyboard , mouse and lcd monitor that have HDMI connection
that is ok

b) if you have a local router in your home that is ok

all you need to do is to know the ip address of raspberry pi you can
easily know ip by using this program named advanced ip scanner you
can download it from this link:-

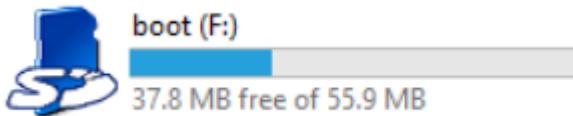
http://www.advanced-ip-scanner.com/index2.php?utm_expid=62919999-51.rwd68Dh0SDGksqgLfQE8g.2&utm_referrer=https%3A%2F%2Fwww.google.com.eg%2F

then open it and click scan you will get the ip address:-



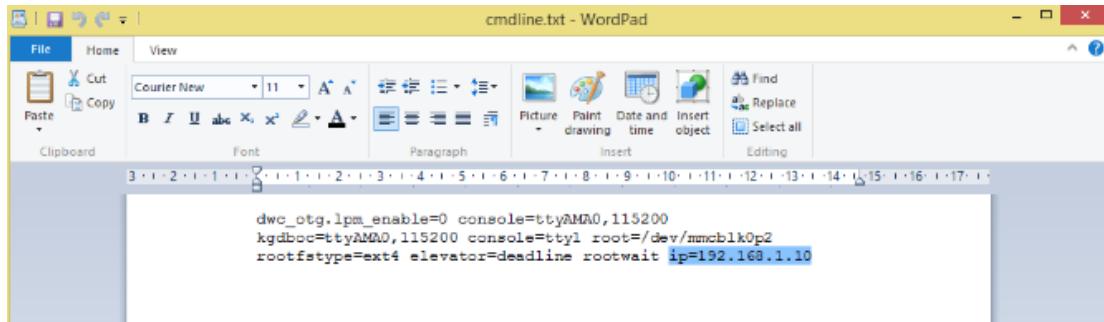
c) i will assume you have only 2 3G modems that only interface with one device so after you do port forward and address reversing steps you should do the following (assuming you are a windows user)

first take of your SD card from raspberry pi and plug in your SD reader and plug them in your PC or laptop .you should see an icon like this

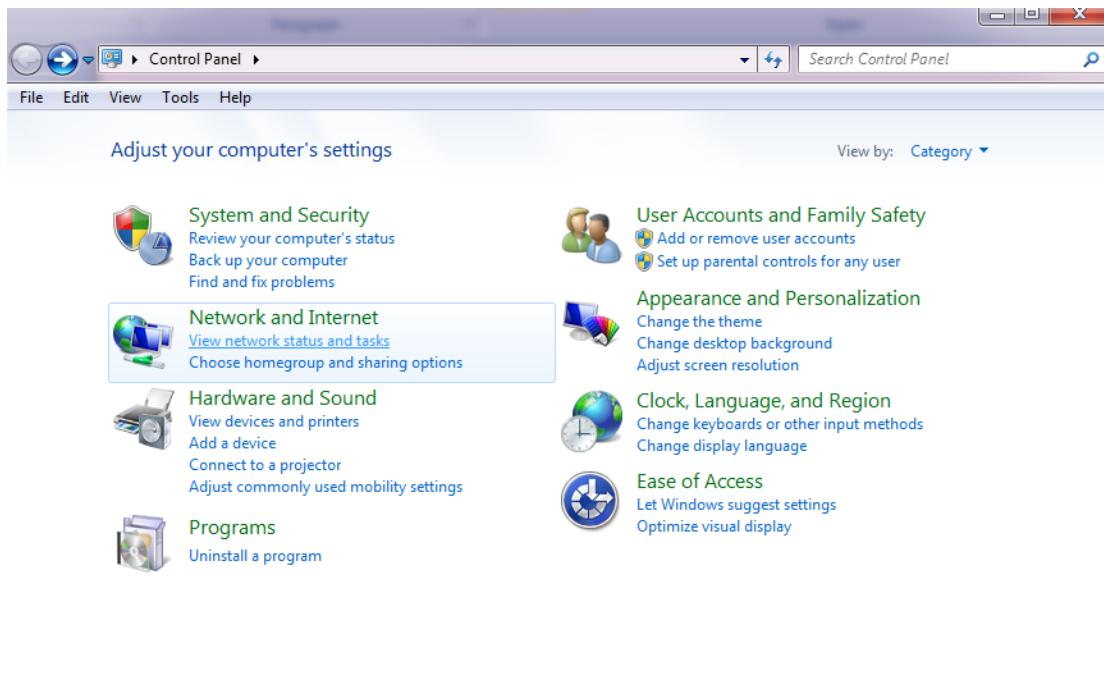


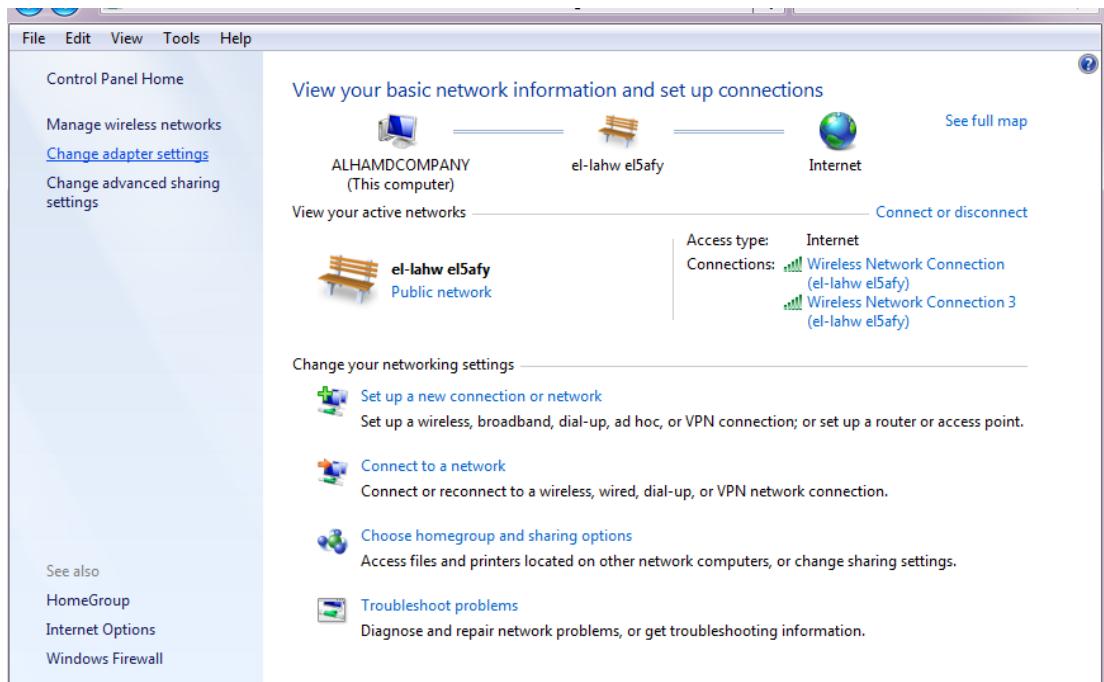
open it (boot icon) and you will see a folder named cmdline.txt open it and add this line to it ip=(your static raspberry pi ip that you want to reserve) in my case i choose ip=192.168.1.10 then save

then plug off SD card and return it to raspberry pi then power on raspberry pi

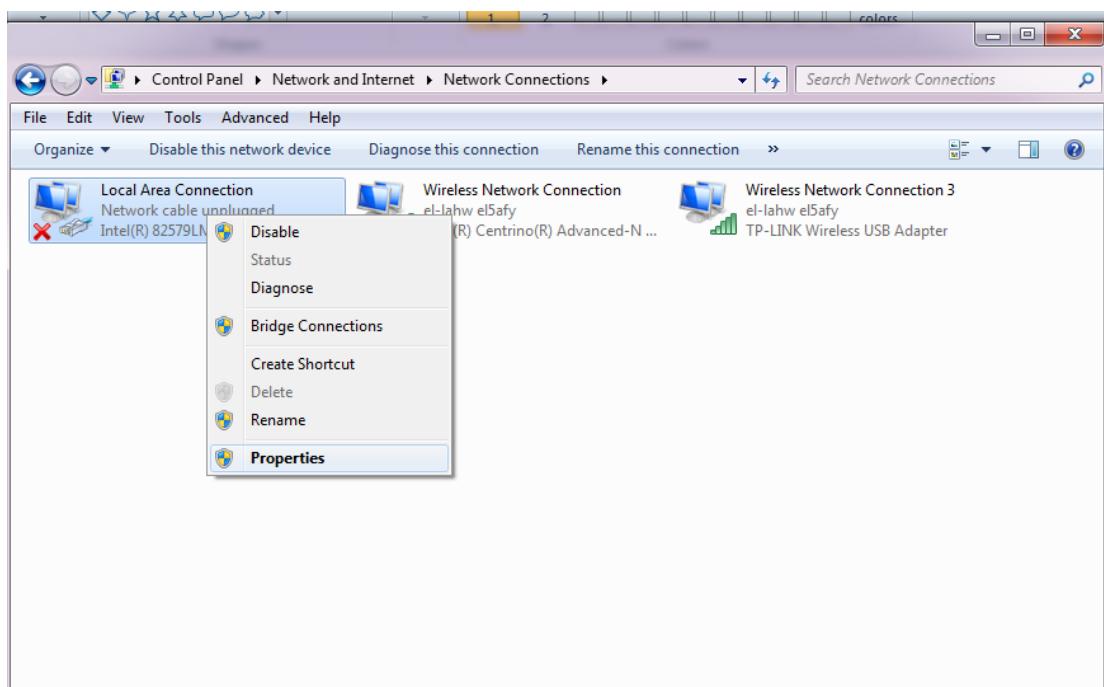


open your control panel => Network and internet=>view internet and Tasks=>change adaptor settings.

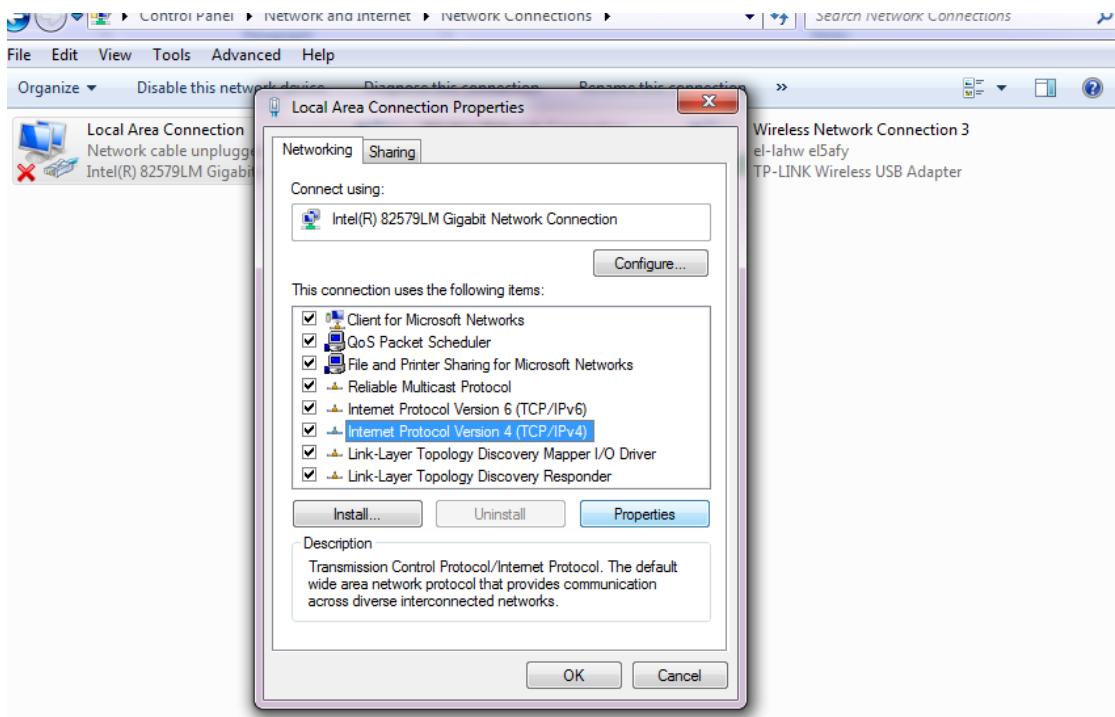




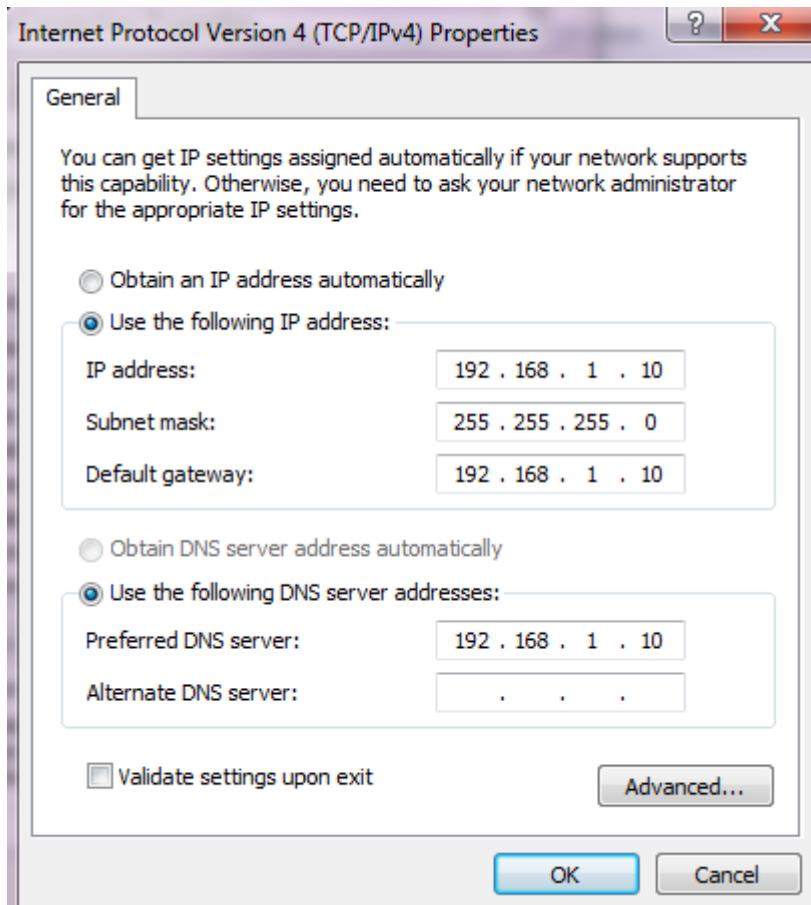
now click right in the local area connection then choose properties



now highlight the choose internet protocol Version 4 (TCP/ IP v4) then click left on properties.



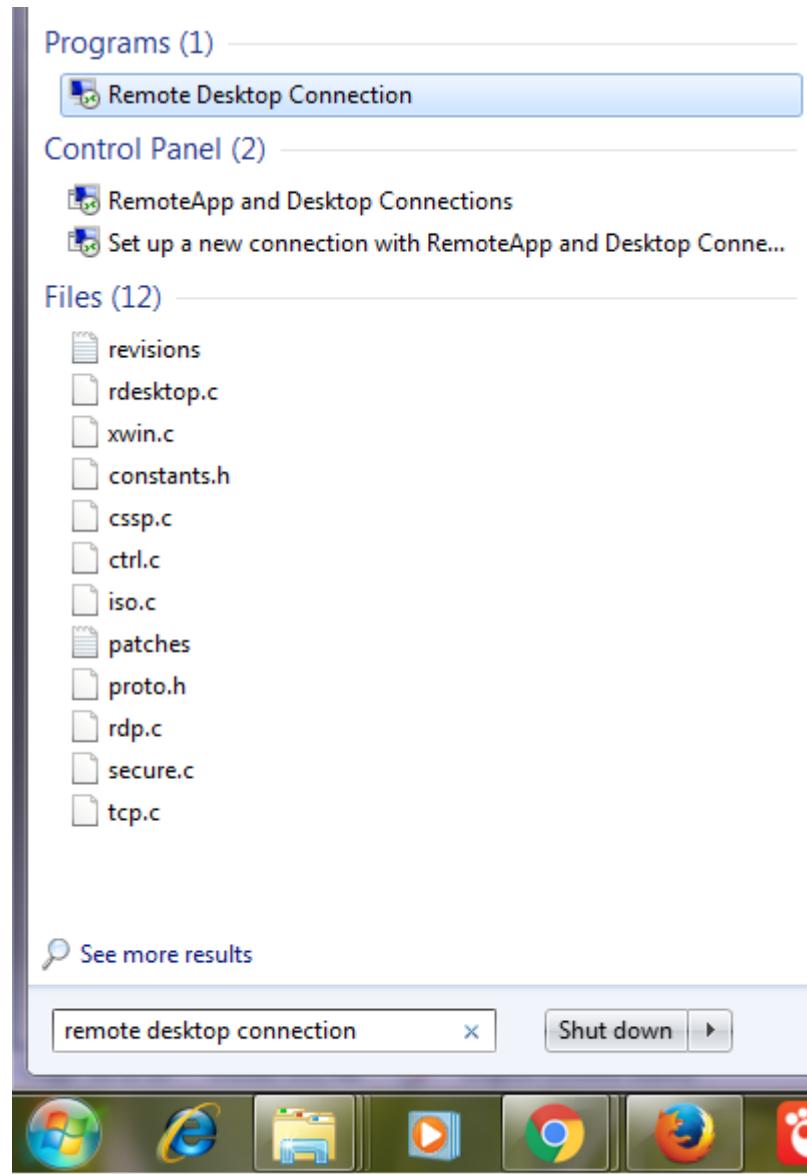
now fill the required fields :-



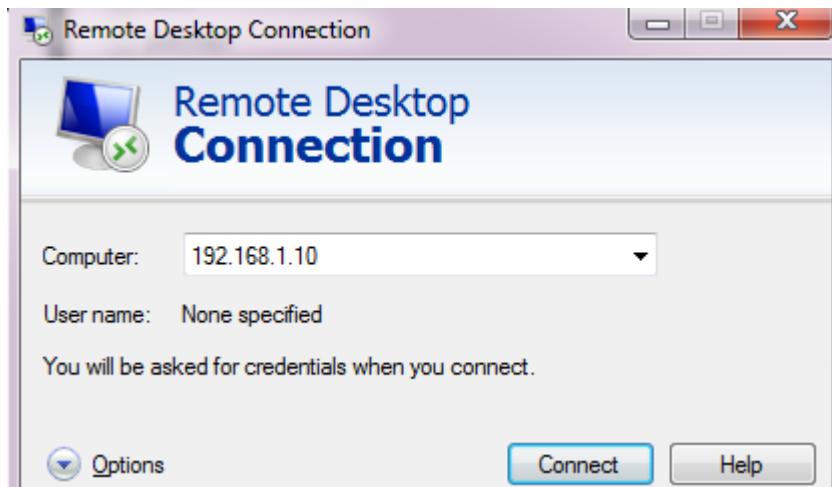
note that all the configuration above done only for case c users

now for cases b,c the following setup process are applied to them:-

i will assume you are a windows user so it is by default you have remote desktop connection program just press start then search it by name



open it and write the ip address for your raspberry pi then click connect



if you use any thing neither than windows don't worry remote desktop connection have already applications for Linux and android just search for a program name remote desktop for xrdp .

now after opening it you should run 1 program by using this command

```
python get_public_ip2.py
```

very important note: after running this command there is a window will appear donot close this windows for ever as it will remain with us

then exit remote desktop connection. and now power on your raspberry pi router or connect your modem to raspberry pi of course you will do that after ending connection between your pi and your laptop through wifi or Ethernet cable.

i assume you doing all setup process while you plug on your pi in portable charger

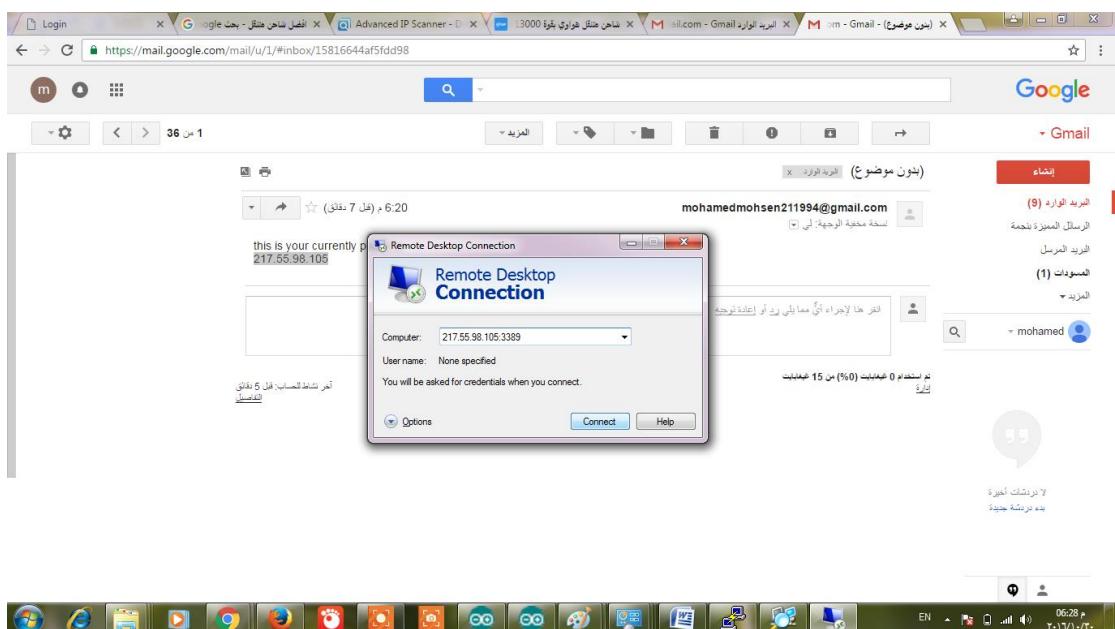
now return your pi and router or modem to their rc car body positions

and open your gmail you will find your public ip address



copy it and paste to your remote desktop connection

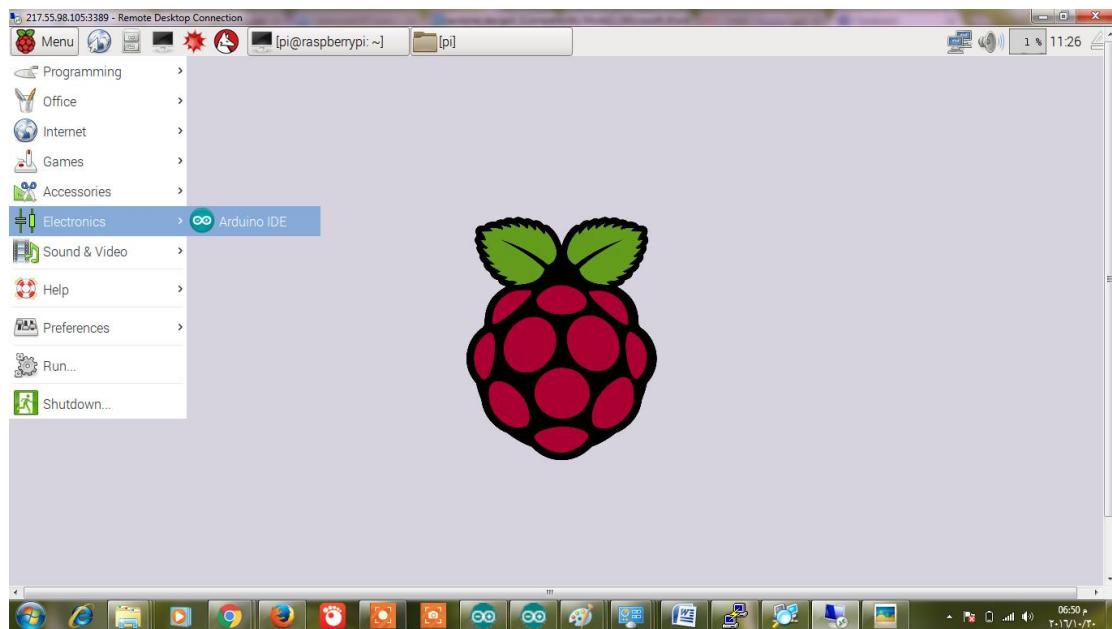
followed by :3389 then connect



now if there are any problem case to disconnect internet to your rc car for a while your pi should send you the new public ip address to your gmail after the return of internet connection to your rc_car so

you should do the same above process again.

now open arduino software as follows:-



copy and paste this code in the sketch:-

```
//pins

int DC_forward=3;

int DC_backward=11;

int servo=9;

//speed and angle

int DC_speed=255;

int state=0;

#include<Servo.h>

Servo myservo;

void setup()

{

    myservo.attach(servo);

    pinMode(DC_forward,OUTPUT);

    pinMode(DC_backward,OUTPUT);

    Serial.begin(9600);
```

```
}

void loop()

{

if(Serial.available() > 0){

state = Serial.read();

}

if(state=='0')

{

DC_speed=0;

}

else if(state=='1')

{

DC_speed=32;

}

else if(state=='2')

{

DC_speed=64;

}

else if(state=='3')

{

DC_speed=96;

} else if(state=='4')

{

DC_speed=128;

}

else if(state=='5')

{
```

```
    DC_speed=160;

} else if(state=='6')

{

    DC_speed=192;

} else if(state=='7')

{

    DC_speed=224;

}

else if(state=='8')

{

    DC_speed=255;

}

if (state=='w')

{

//forward

    analogWrite(DC_backward,0);

    analogWrite(DC_forward,DC_speed);

    myservo.write(90);

}

else if (state=='e')

{

//forward_wright

    analogWrite(DC_forward,DC_speed);

    analogWrite(DC_backward,0);

    myservo.write(135);

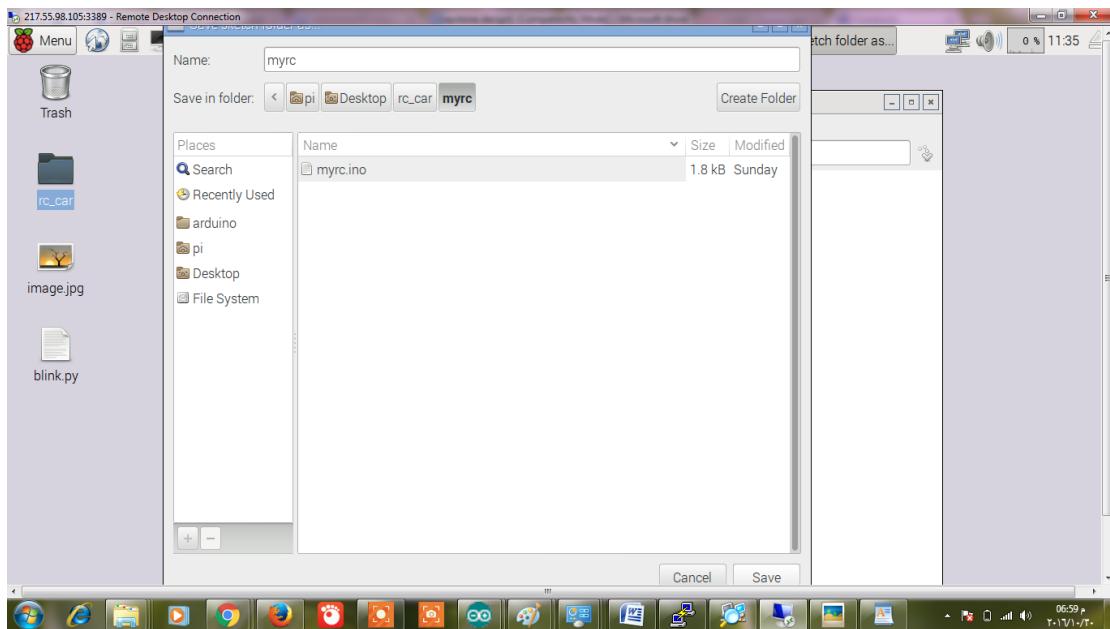
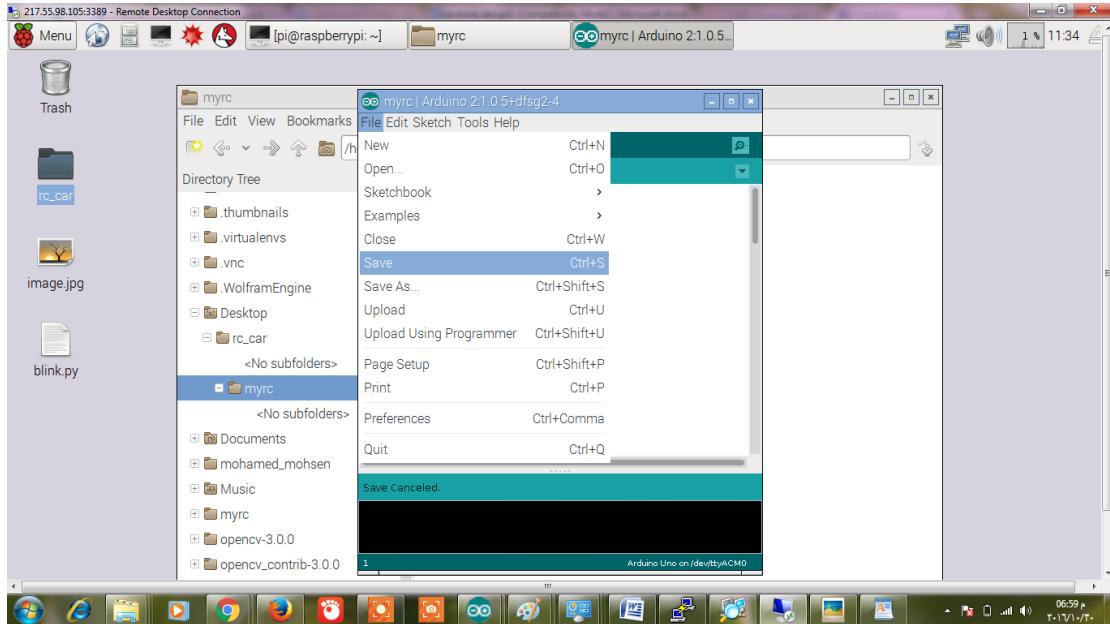
}

else if (state=='q')
```

```
{  
    //forward left  
  
    analogWrite(DC_forward,DC_speed);  
  
    analogWrite(DC_backward,0);  
  
    myservo.write(45);  
  
}  
  
else if (state=='s')  
  
{  
    //backward  
  
    analogWrite(DC_forward,0);  
  
    analogWrite(DC_backward,DC_speed);  
  
    myservo.write(90);  
  
}  
  
else if (state=='d')  
  
{  
    //backward wright  
  
    analogWrite(DC_forward,0);  
  
    analogWrite(DC_backward,DC_speed);  
  
    myservo.write(135);  
  
}  
  
else if (state=='a')  
  
{  
    //backward left  
  
    analogWrite(DC_forward,0);  
  
    analogWrite(DC_backward,DC_speed);  
  
    myservo.write(45);  
  
}
```

}

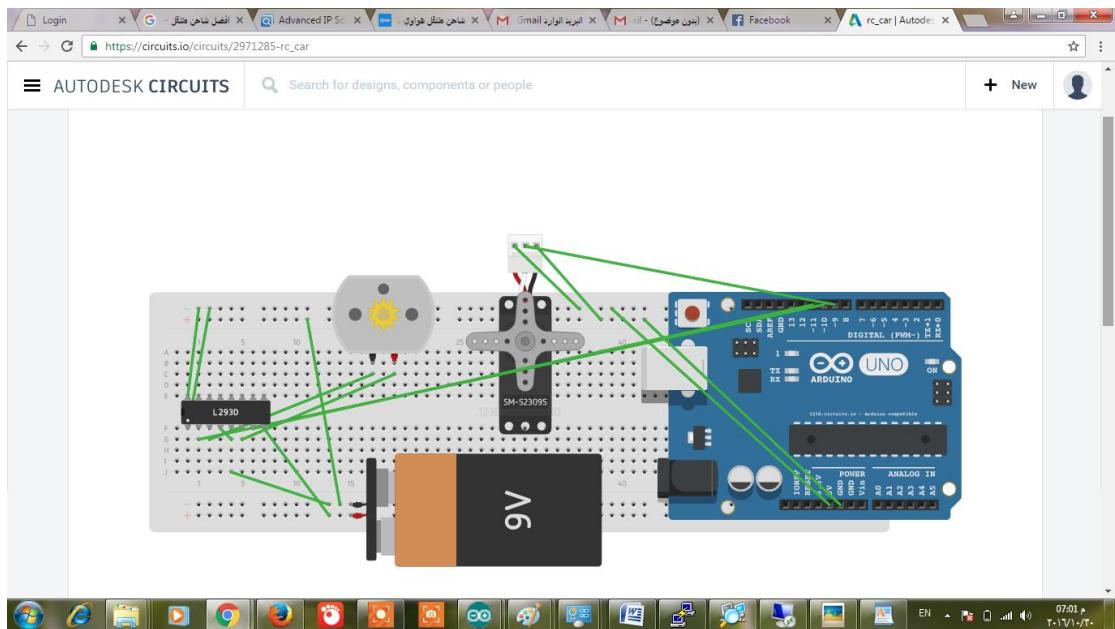
save this code with any name you want i save it with this name myrc and then upload it



i saved it in myrc folder that i created in my_rc folder

now upload it

hardware connection:-

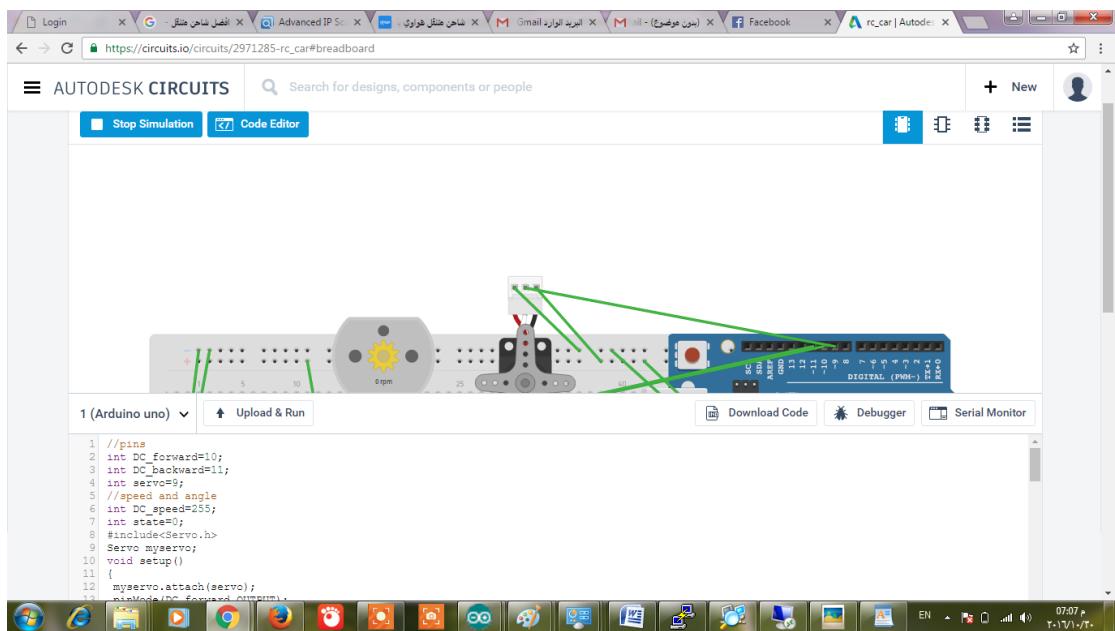


to check it you can open it by open this link to examine it carefully and simulate it:-

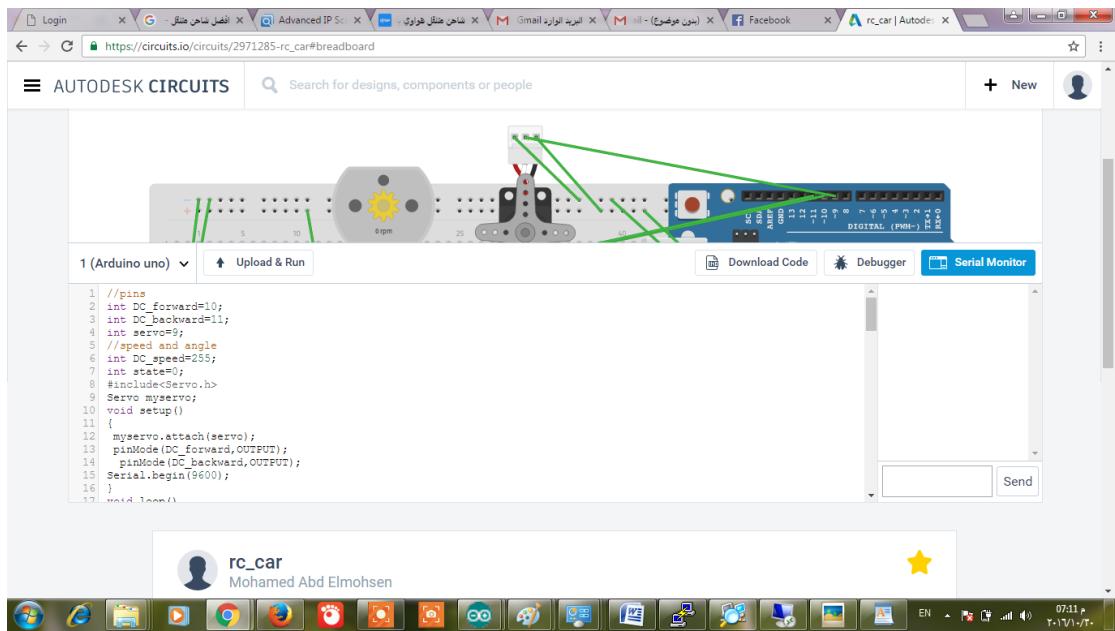
https://circuits.io/circuits/2971285-rc_car

for simulation: follow these instruction:-

press start simulation button and code editor:-



press serial monitor



instruction:-

for forward :-send w

for forward right:-send e

for forward left:-send q

for backward :-send s

for backward right:-send d

for backward left:-send a

for changing DC motor speed send number from 0 up to 8

0 means speed =0 8 means speed= full power speed .

the rc car using steering mechanism:-

for more information please watch these videos:-

https://www.youtube.com/watch?v=wzYQ_9hRqkY

<https://www.youtube.com/watch?v=NL5-FV28uRA>

now to control this car more easier i make a python gui to control the car and here is the code i make :-

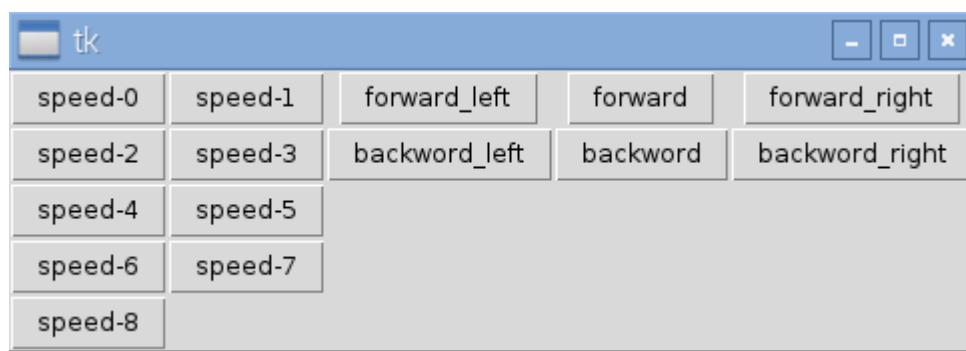
```
import serial
from Tkinter import *
class send_command:
    def __init__(self,n):
        port=serial.Serial("/dev/ttyACM0",baudrate=9600,timeout=3.0)
        port.write(n)
        print(n)
def q():
    send_command('q')
def w():
    send_command('w')
def e():
    send_command('e')
def a():
    send_command('a')
def s():
    send_command('s')
def d():
    send_command('d')
def number0():
    send_command('0')
def number1():
    send_command('1')
def number2():
    send_command('2')
def number3():
    send_command('3')
def number4():
    send_command('4')
def number5():
    send_command('5')
def number6():
    send_command('6')
def number7():
    send_command('7')
def number8():
    send_command('8')
root=Tk()
frame3=Frame(root)
frame3.pack(side=LEFT)
frame1=Frame(root)
frame1.pack(side=TOP)
frame2=Frame(root)
frame2.pack(side=BOTTOM)
rc_button1=Button(frame1,text="forward_left",command=q)
rc_button1.grid(row=0,column=0)
rc_button1.grid(row=0,column=0)
rc_button2=Button(frame1,text="forward",command=w)
rc_button2.grid(row=0,column=1)
rc_button3=Button(frame1,text="forward_right",command=e)
rc_button3.grid(row=0,column=2)
rc_button4=Button(frame1,text="backword_left",command=a)
rc_button4.grid(row=1,column=0)
rc_button5=Button(frame1,text="backword",command=s)
rc_button5.grid(row=1,column=1)
rc_button6=Button(frame1,text="backword_right",command=d)
rc_button6.grid(row=1,column=2)
```

```

speed_button1=Button(frame3,text="speed-0",command=number0)
speed_button1.grid(row=0,column=0)
speed_button2=Button(frame3,text="speed-1",command=number1)
speed_button2.grid(row=0,column=1)
speed_button3=Button(frame3,text="speed-2",command=number2)
speed_button3.grid(row=1,column=0)
speed_button4=Button(frame3,text="speed-3",command=number3)
speed_button4.grid(row=1,column=1)
speed_button5=Button(frame3,text="speed-4",command=number4)
speed_button5.grid(row=2,column=0)
speed_button6=Button(frame3,text="speed-5",command=number5)
speed_button6.grid(row=2,column=1)
speed_button7=Button(frame3,text="speed-6",command=number6)
speed_button7.grid(row=3,column=0)
speed_button8=Button(frame3,text="speed-7",command=number7)
speed_button8.grid(row=3,column=1)
speed_button9=Button(frame3,text="speed-8",command=number8)
speed_button9.grid(row=4,column=0)
root.mainloop()

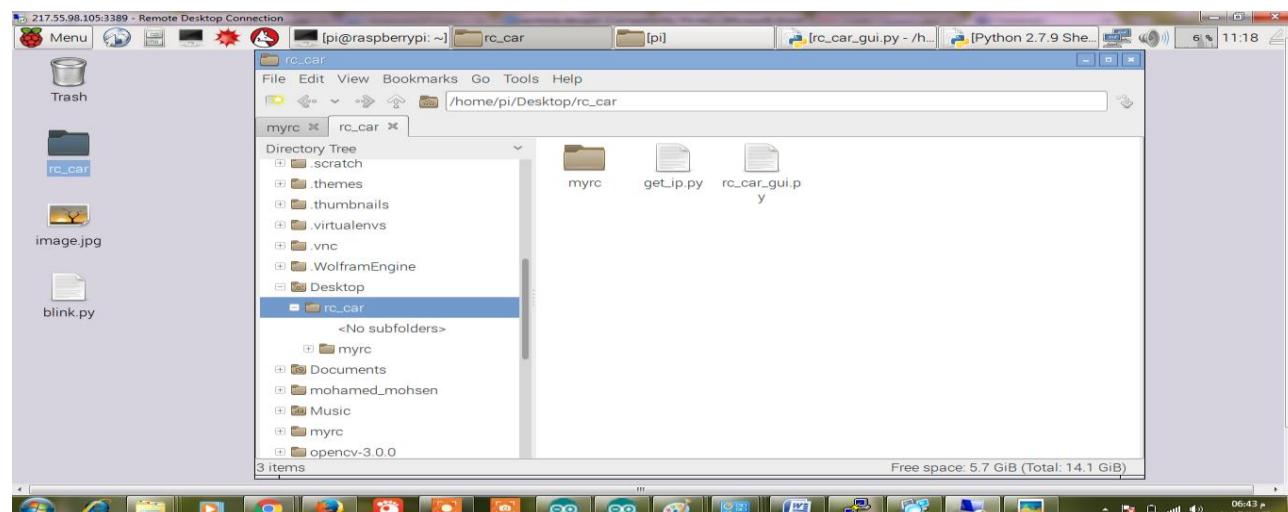
```

and here is how it look when i run it



for make things easy i create a folder called rc_car in the desktop

and i put the required codes for this project in it



how to control rc through internet:-

after preparing and install our tools and writes our codes here is how to put all things together:-

1- first connect to your raspberry pi through internet and this done in setup process above

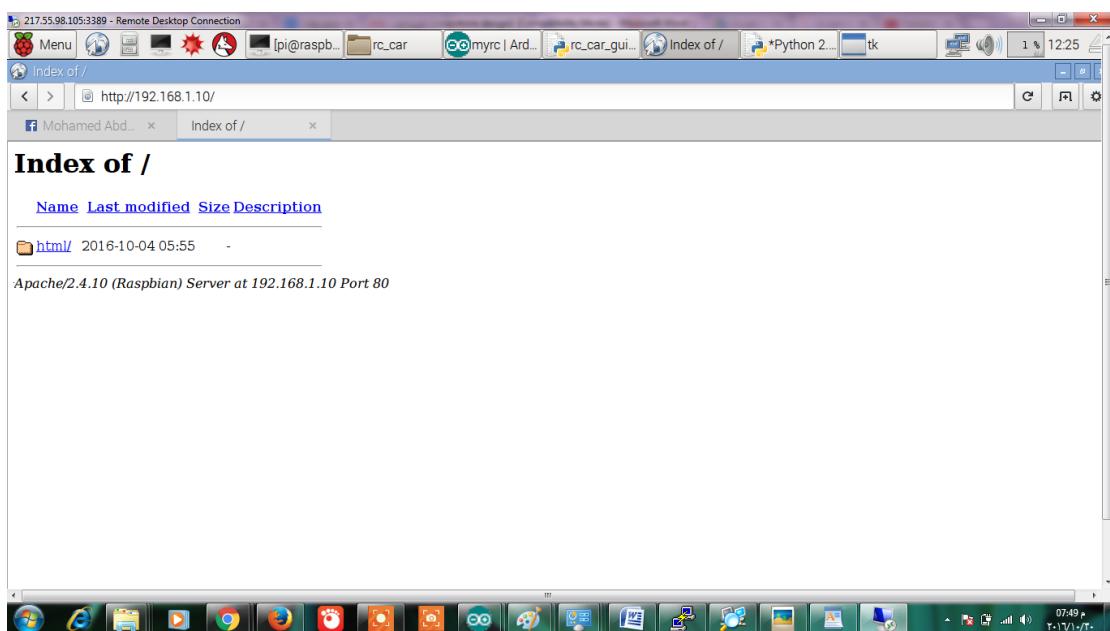
2- running all codes we build (in setup process we already run and upload 2 programs we don't need to do this again get_ip.py,myrc.io)

note:- you can run all python programs by double click on them ,by default they will opened by idle2 then click F5

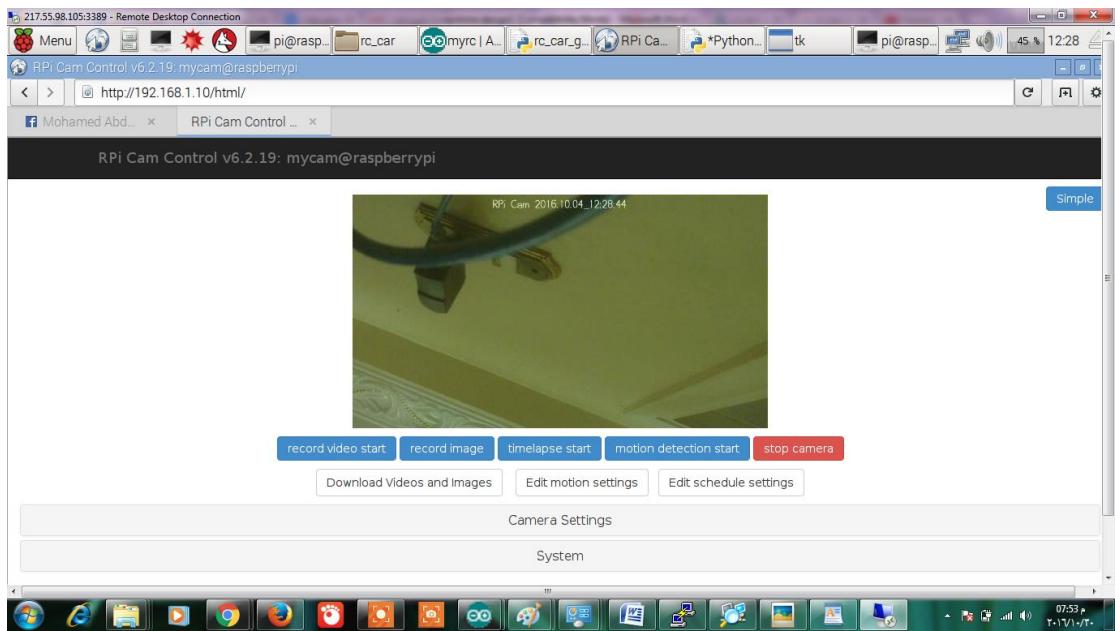
2- after that open your internet browser then write this ip address

if you using raspberry pi board camera

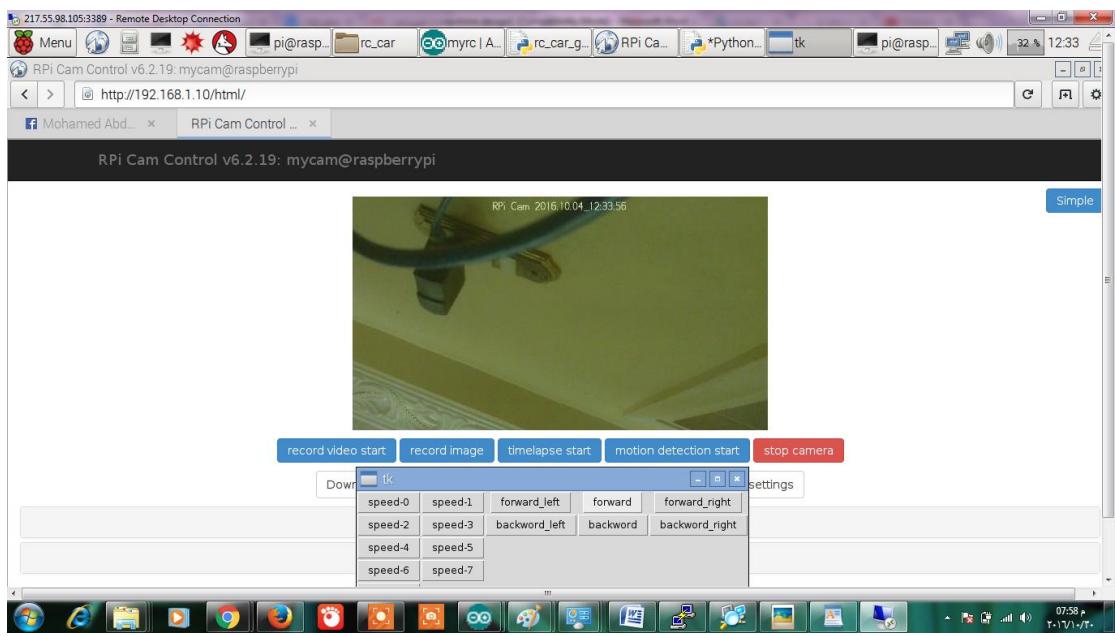
192.168.1.10 (this is my static pi address for your case write your own pi address)



click on html this windows should appear



then if you press on tk application a gui will appear together with video streaming now you can easily control rc_car



have fun:-

if you use usb camera then enter this ip address on the browser
(your pi ip static address):8081

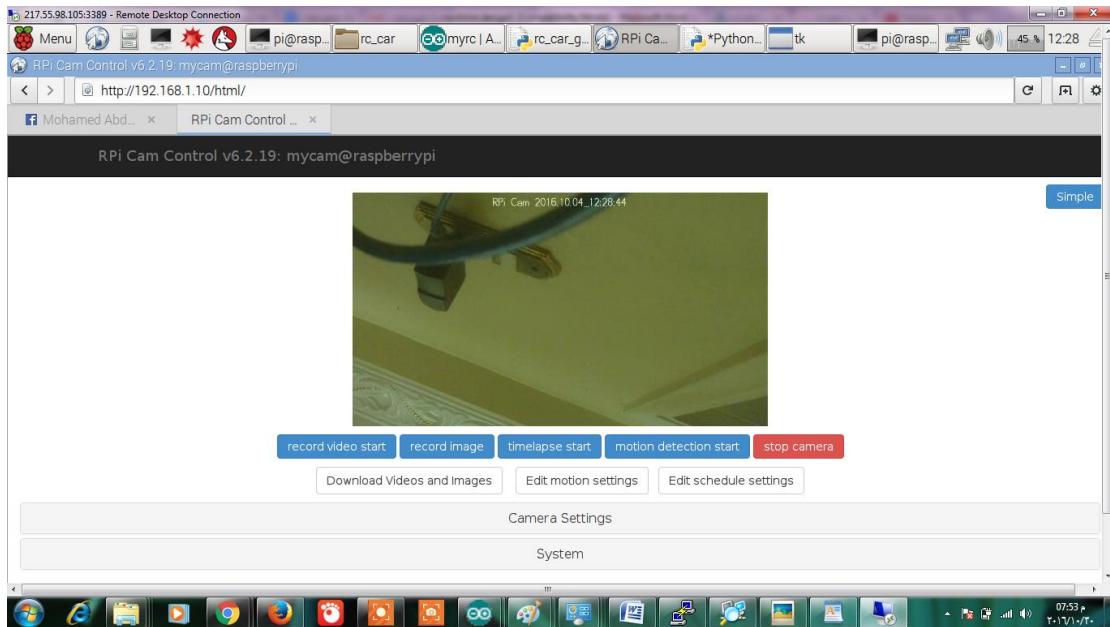
for mine:-

192.168.1.10:8081

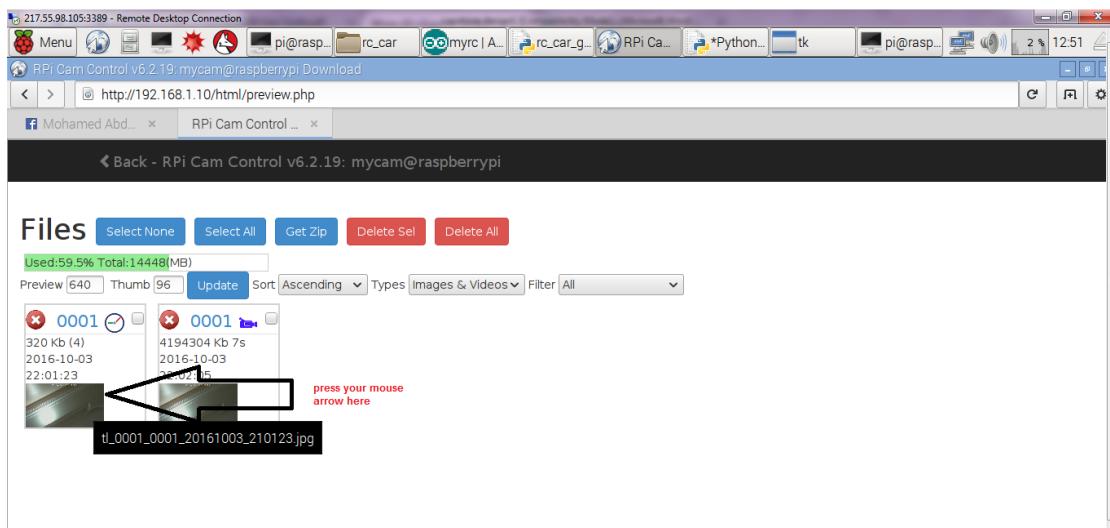


and if you want to take pictures and record videos you can do that easily for raspberry pi board camera using this html page by clicking on record video start,record imageetc that i get it from web camera interface project

<http://elinux.org/RPi-Cam-Web-Interface>



and if you want to download any thing you should press on download videos and images button this page will appear



and by pressing in the photo or video you want this page will appear

you can download it in any place of raspberry pi you want . for more information please refer to <http://elinux.org/RPi-Cam-Web-Interface>

and if you use USB cam you cannot do this and if you want to record or take a picture you cannot .

note:-(honestly i tried to add two buttons one for take a picture and the other for record the video although there is no bugs in the code i made, i donot have a usb camera to test the code and i tried to get intuition at

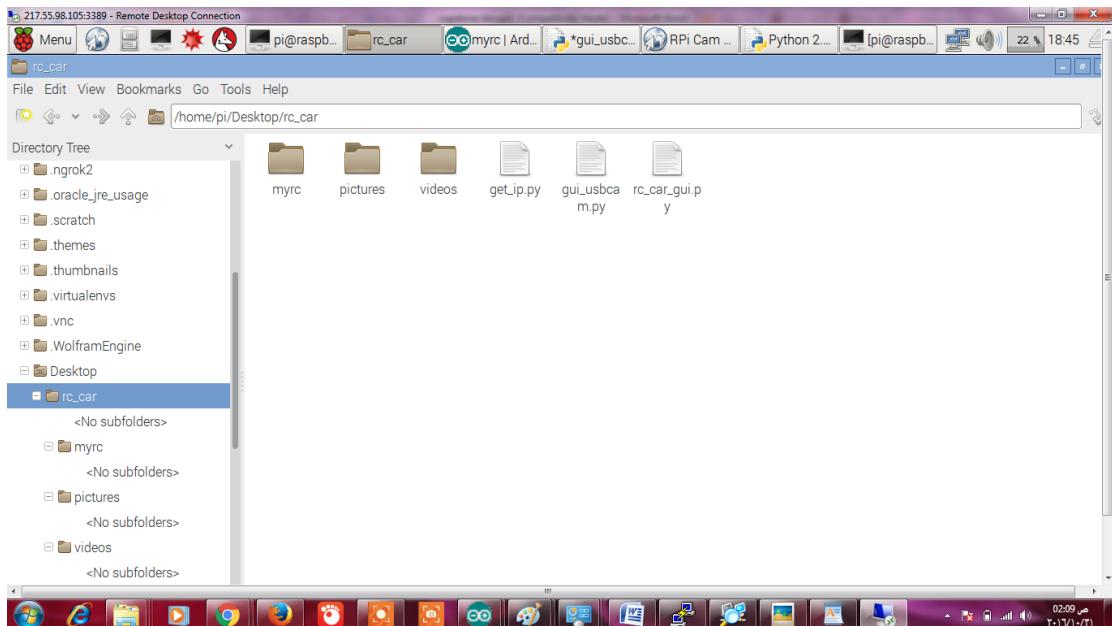
least by make these buttons interface with pi board cam through terminal commands to be similar to USB cam case as there is no standard python library to usb cam but i realize that RPi-Cam-Web-Interface has play on some config to enable picam can do streaming so i cannot use command raspistill or raspivid to record a picture or a video

```
mimal: Failed to run camera app. Please check for firmware updates
pi@raspberrypi ~ $ raspistill -o 1.jpg
mimal: mimal_vc_component_enable: failed to enable component: ENOSPC
mimal: camera component couldn't be enabled
mimal: main: Failed to create camera component
mimal: Failed to run camera app. Please check for firmware updates
pi@raspberrypi ~ $ 
```

)

any way for USB cam users try to make the following it may work for you and if anyone want to configure or fix my following work if it doesnot work feel free to do it .

first make 2 directories named pictures and videos (all letters are small)



then open your idle2 program and copy and paste this program then save it with any name you want then press F5 to execute:-

```
import serial
import subprocess
from Tkinter import *
```

```
x=1
y=2

class send_command:

    def __init__(self,n):
        port=serial.Serial("/dev/ttyACM0",baudrate=9600,timeout=3.0)
        port.write(n)
        print(n)

    def q():
        send_command('q')

    def w():
        send_command('w')

    def e():
        send_command('e')

    def a():
        send_command('a')

    def s():
        send_command('s')

    def d():
        send_command('d')

def number0():
    send_command('0')

def number1():
    send_command('1')

def number2():
    send_command('2')

def number3():
    send_command('3')
```

```
def number4():
    send_command('4')

def number5():
    send_command('5')

def number6():
    send_command('6')

def number7():
    send_command('7')

def number8():
    send_command('8')

def take_picture():
    global x
    print("fswebcam -r 1280x720 --no-banner
/home/pi/Desktop/rc_car/pictures/"+str(x)+".jpg")
    subprocess.call("fswebcam -r 1280x720 --no-banner
/home/pi/Desktop/rc_car/pictures/"+str(x)+".jpg",shell=True)
    x=x+1

def take_video():
    global y
    print("ffmpeg -f video4linux2 -r 25 -s 640x480 -i /dev/video0"+str(y)+".av")
    subprocess.call("ffmpeg -f video4linux2 -r 25 -s 640x480 -i
/dev/video0"+str(y)+".av",shell=True)
    y=y+1

root=Tk()

frame3=Frame(root)

frame3.pack(side=LEFT)

frame1=Frame(root)

frame1.pack(side=TOP)

frame2=Frame(root)
```

```
frame2.pack(side=BOTTOM)

rc_button1=Button(frame1,text="forward_left",command=q)
rc_button1.grid(row=0,column=0)
rc_button1.grid(row=0,column=0)

rc_button2=Button(frame1,text="forward",command=w)
rc_button2.grid(row=0,column=1)

rc_button3=Button(frame1,text="forward_right",command=e)
rc_button3.grid(row=0,column=2)

rc_button4=Button(frame1,text="backword_left",command=a)
rc_button4.grid(row=1,column=0)

rc_button5=Button(frame1,text="backword",command=s)
rc_button5.grid(row=1,column=1)

rc_button6=Button(frame1,text="backword_right",command=d)
rc_button6.grid(row=1,column=2)

picture=Button(frame2,text="take a picture",command=take_picture)
picture.grid(row=0,column=0)

video=Button(frame2,text="record a video",command=take_video)
video.grid(row=0,column=1)

speed_button1=Button(frame3,text="speed-0",command=number0)
speed_button1.grid(row=0,column=0)

speed_button2=Button(frame3,text="speed-1",command=number1)
speed_button2.grid(row=0,column=1)

speed_button3=Button(frame3,text="speed-2",command=number2)
speed_button3.grid(row=1,column=0)

speed_button4=Button(frame3,text="speed-3",command=number3)
speed_button4.grid(row=1,column=1)

speed_button5=Button(frame3,text="speed-4",command=number4)
```

```

speed_button5.grid(row=2,column=0)

speed_button6=Button(frame3,text="speed-5",command=number5)

speed_button6.grid(row=2,column=1)

speed_button7=Button(frame3,text="speed-6",command=number6)

speed_button7.grid(row=3,column=0)

speed_button8=Button(frame3,text="speed-7",command=number7)

speed_button8.grid(row=3,column=1)

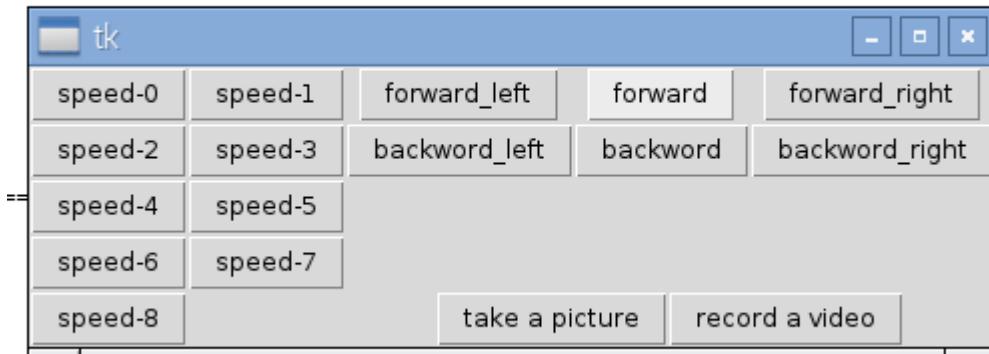
speed_button9=Button(frame3,text="speed-8",command=number8)

speed_button9.grid(row=4,column=0)

root.mainloop()

```

and again this code you should test yourself as i didnot have usb cam to test and you must see a picture like this:-



click in take a picture or record a video and then check your pictures and videos folders i hope this code work with you guys best of wishes.

any way for recording videos or taking pictures and save it in pi itself i donot like it as what ever we do raspberry pi has very limited memory storage so instead of recording videos or capture a photo using pi itself

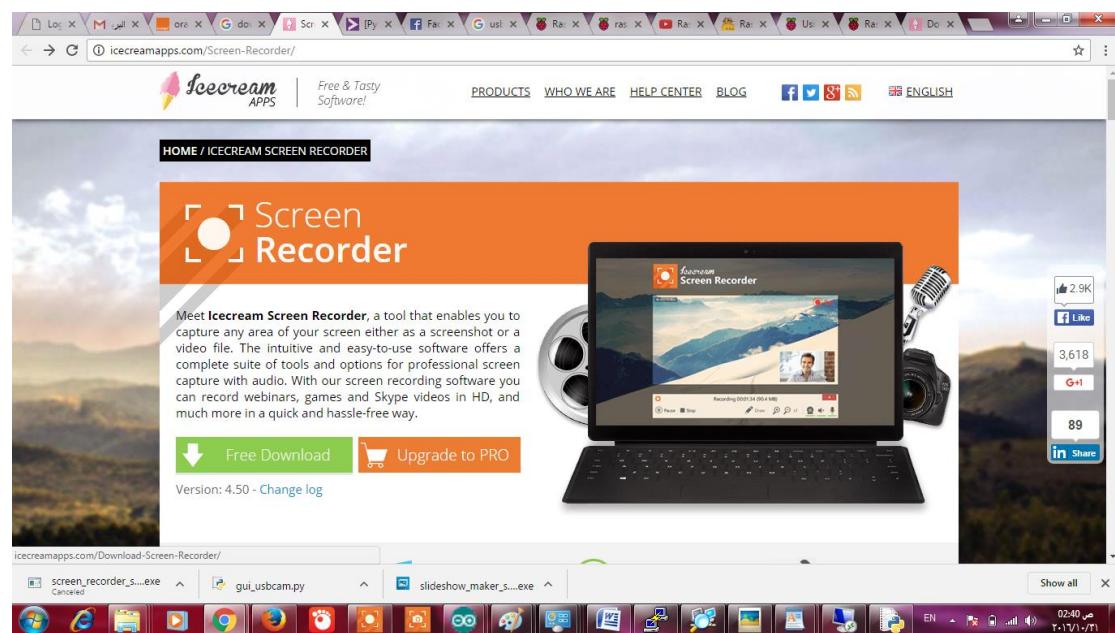
we can make this mission with our pc or laptop that have a huge storage memory compared to raspberry pi and here is a method for that:-

i will assume you are a windows user

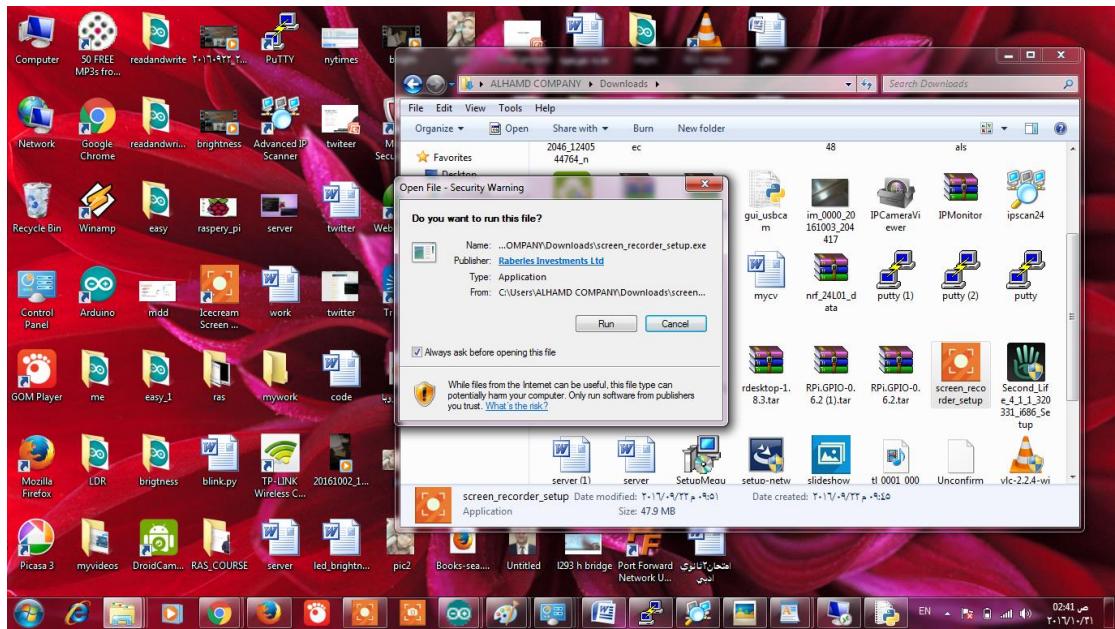
1- install any desktop screen recorder program and i recommend this free program it is called icecream screen recorder

[/http://icecreamapps.com/Screen-Recorder](http://icecreamapps.com/Screen-Recorder)

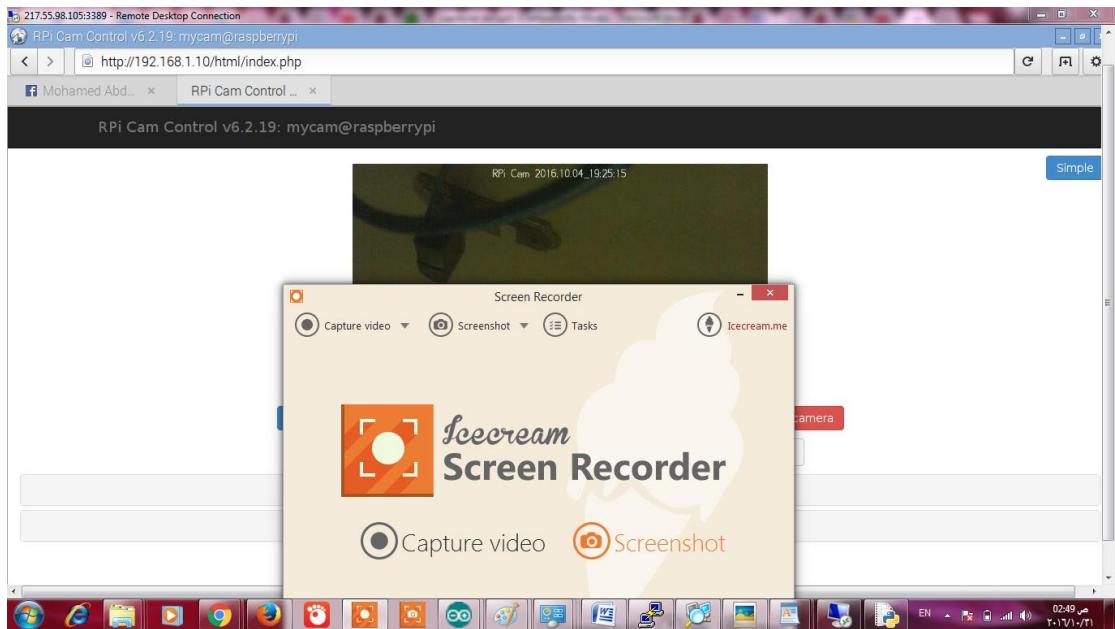
choose free download and it will downloaded to your desktop automatically.



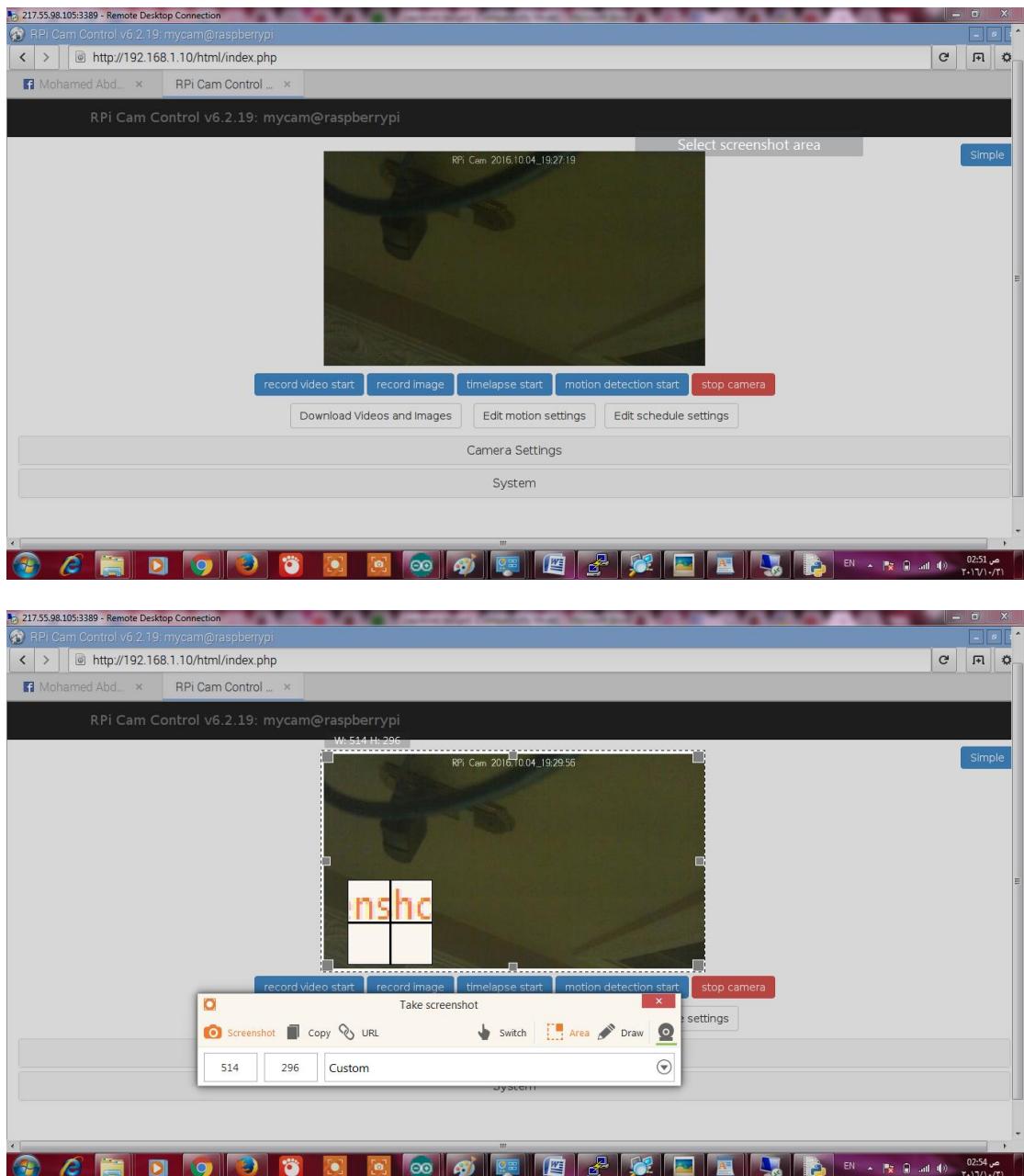
open screen recorder setup file and then press run



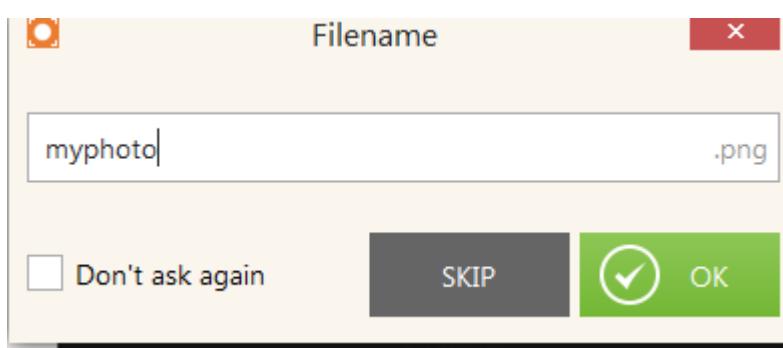
the setup is pretty easy to do so i willnot include it any way after finishing installation open your screen recorder from the icon appeared in the desktop then tap to your desktop remote program



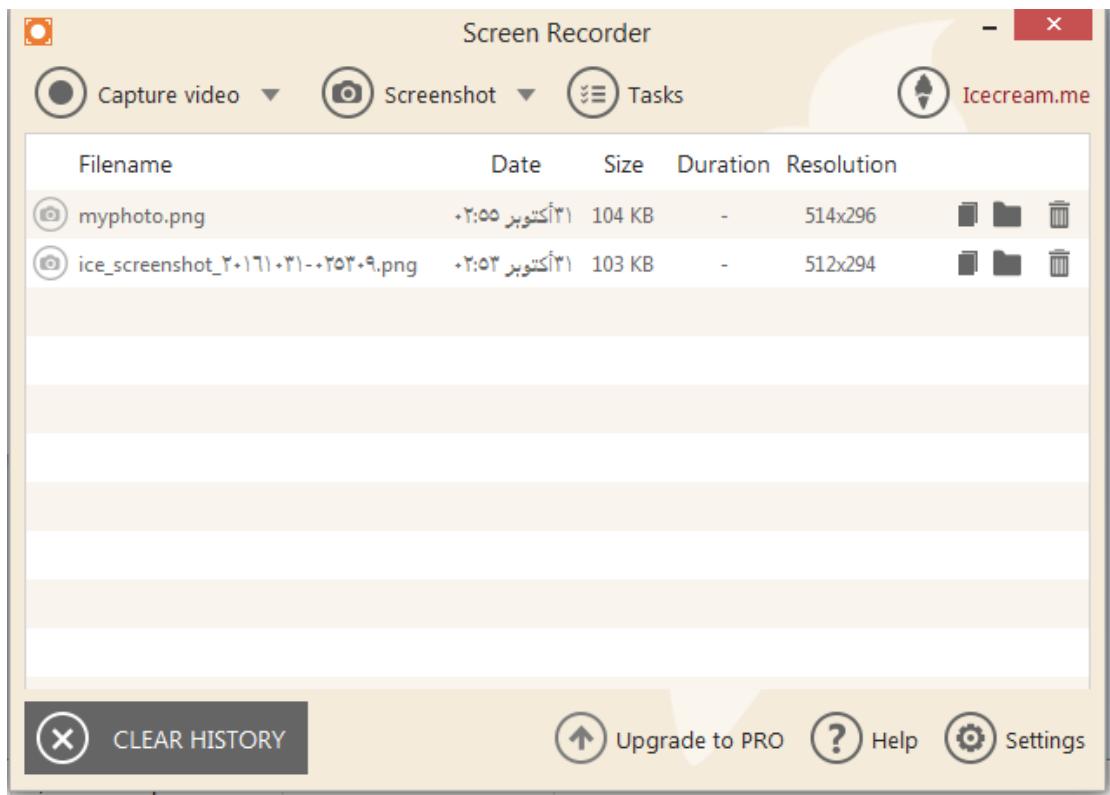
for screenshot click on screen shoot option and then draw a rectangle that contain the pi image



after determine the area click in screenshot

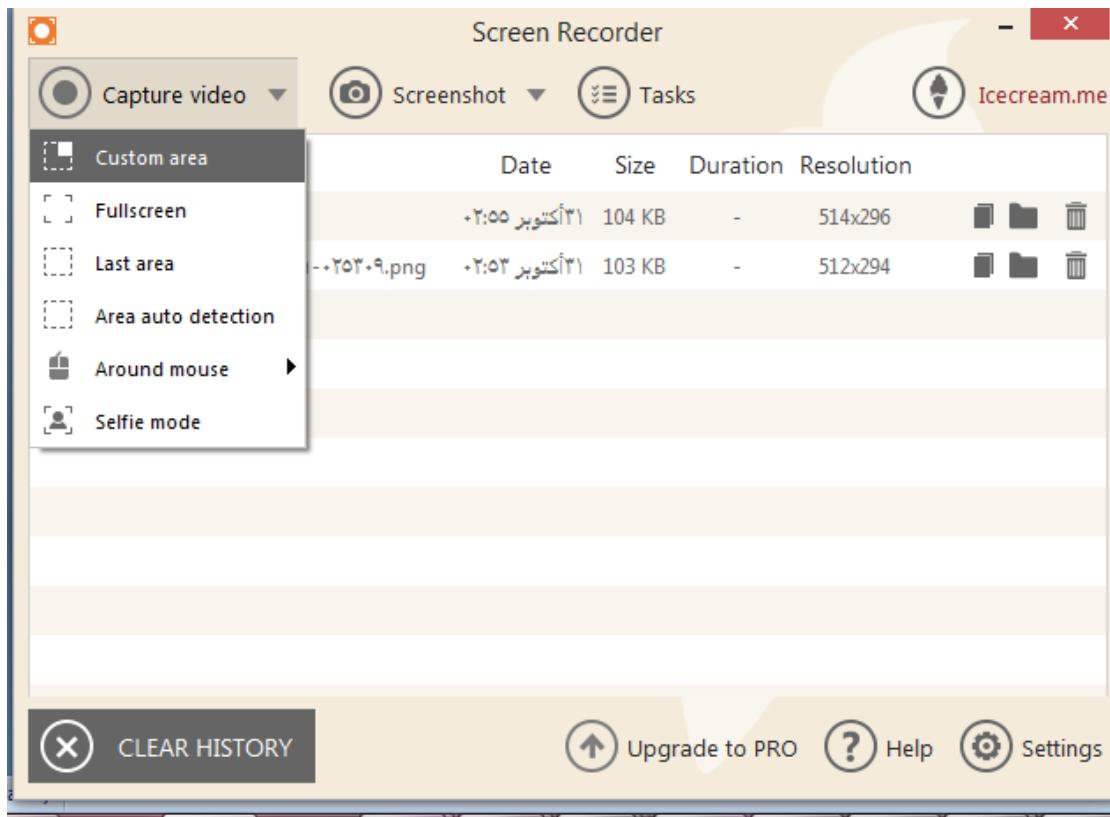


write the name of your photo and click ok this screen will appear



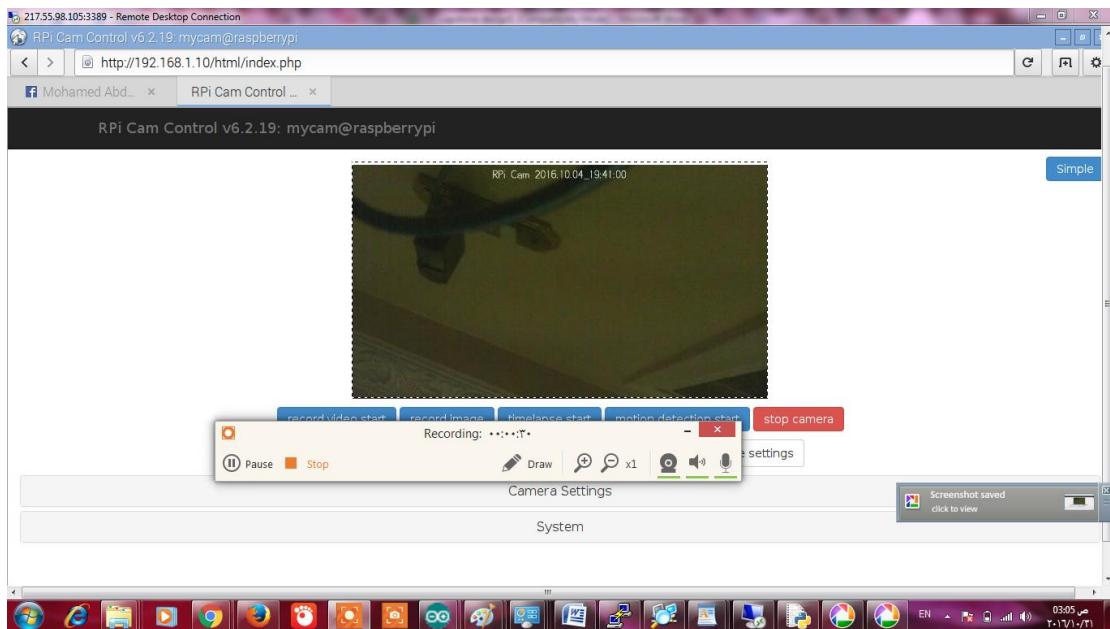
as you see you can copy image open its folder or even delete it from history.

for recording video press on capture video custom area



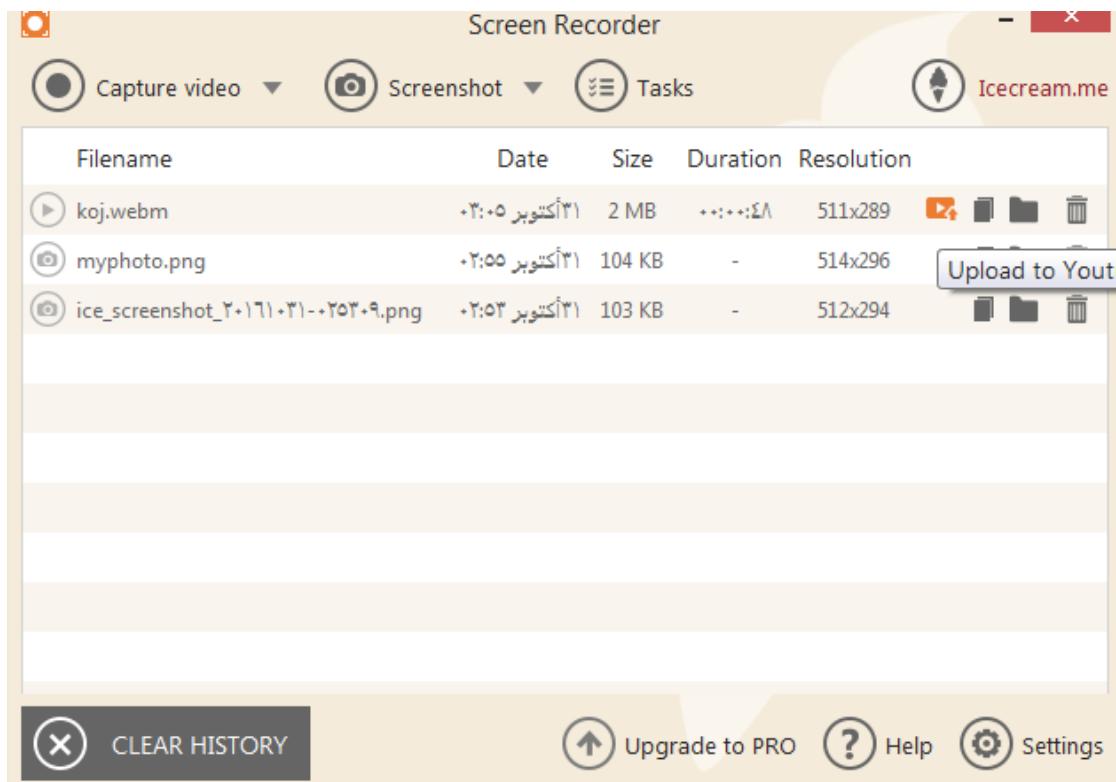
it has almost the same process you will determine the area

after that a toolbox will appear choose rec to start recording then choose stop to stop recording after that

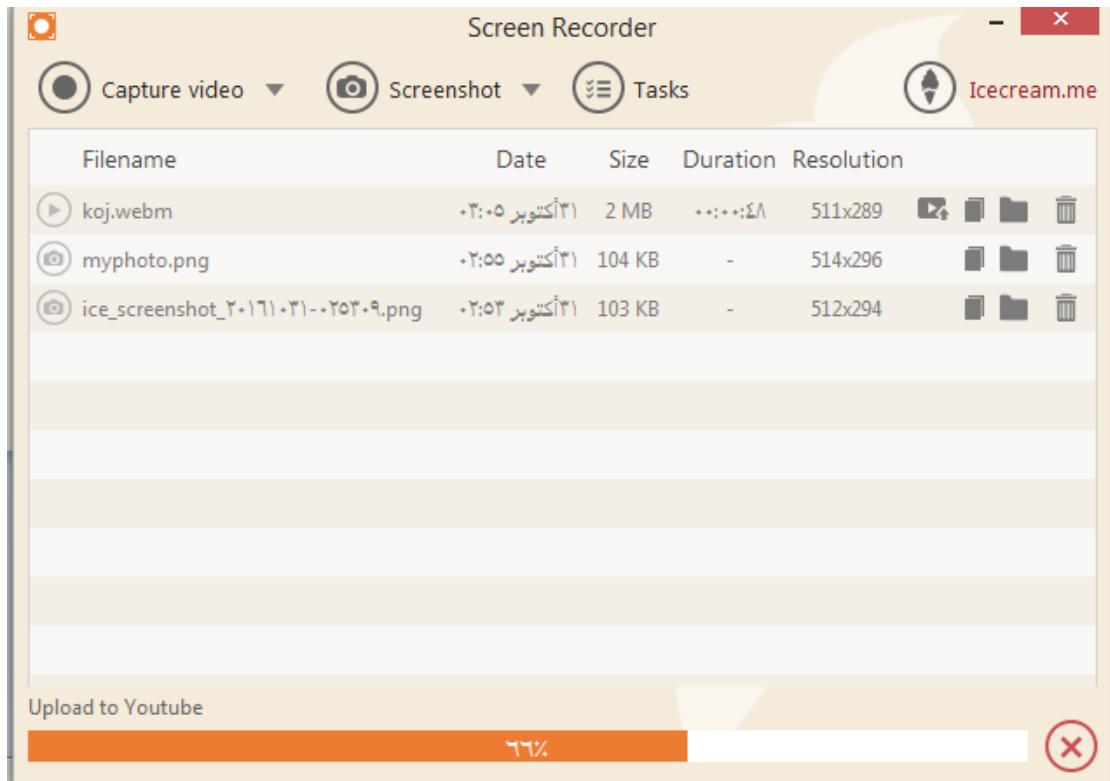


as you can see you record your voice also if you want to add some comments on video you get from raspberry pi

then it will ask you the name of this file to save



then this screen will appear it has the same features as the screenshot before except adding upload to youtube button. it will ask you your gmail account and password to access it to upload your video then this screen will appear



after finishing it will send you a congratulation message and a link to the uploaded video in your account in youtube click open and enjoy!

that is all !!!!!

i hope you enjoy this project guys and as you see i left a huge pins that isnot connected for you guys to make you add the components you want for your specific application and i left all analog to digital converters pins(ADC) for your sensors application that is only built in arduino.

best of luck.