

# Weekly Report Four

**Date:** Nov 9, 2019

**Name:** Shuwen Li (V00024025) & Zijian Chen (V00867494)

**Course:** CSC461 Course Project

## Title:

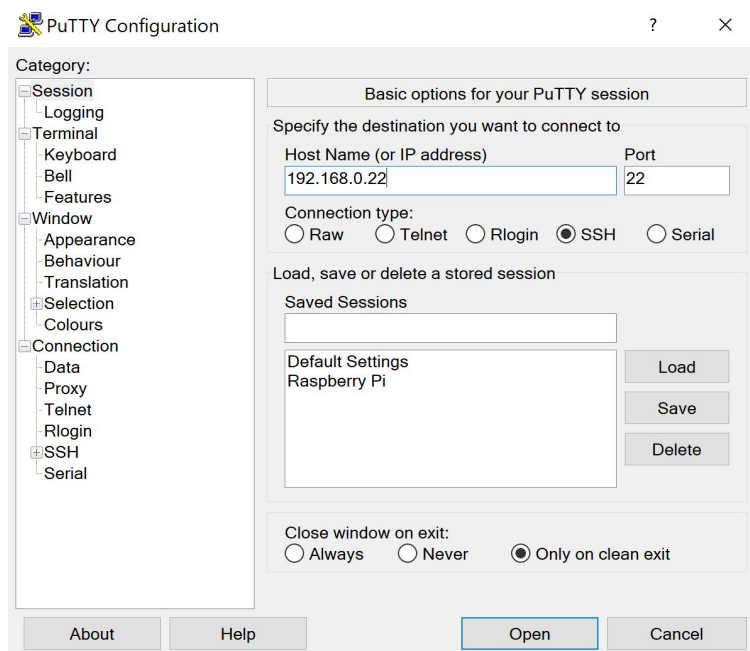
Test out the ultrasonic sensor and use it to control the Picar keep in a good distance between the object and itself.

## Description:

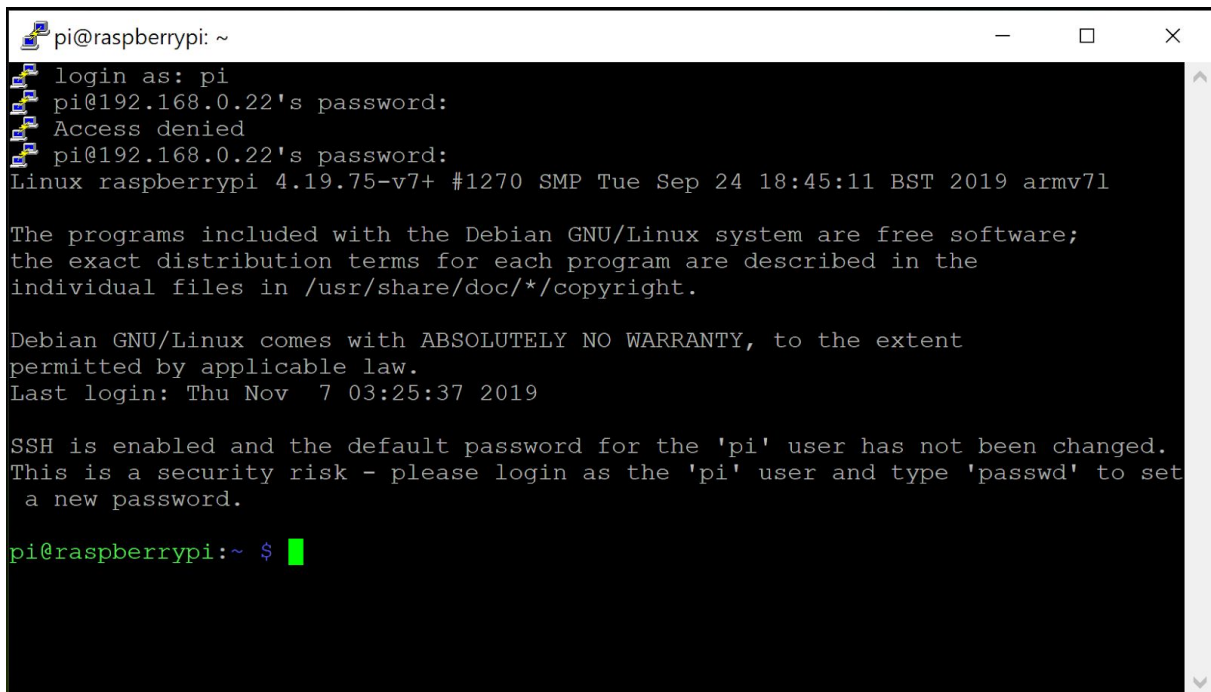
First, we need to connect the Picar by using ssh, and we need to run the server code on the Picar. After, we need to run the client code on my local desktop. After that, we need to run the function and place an object on the front of the Picar, and the Picar will automatically use the ultrasonic sensor to detecting the front and calculate the distance and keep itself in a good distance.

## Process:

First, use PuTTY to connect to our Picar and it is basic on ssh. Before connecting to Picar we need to know what is IP address on the Picar.

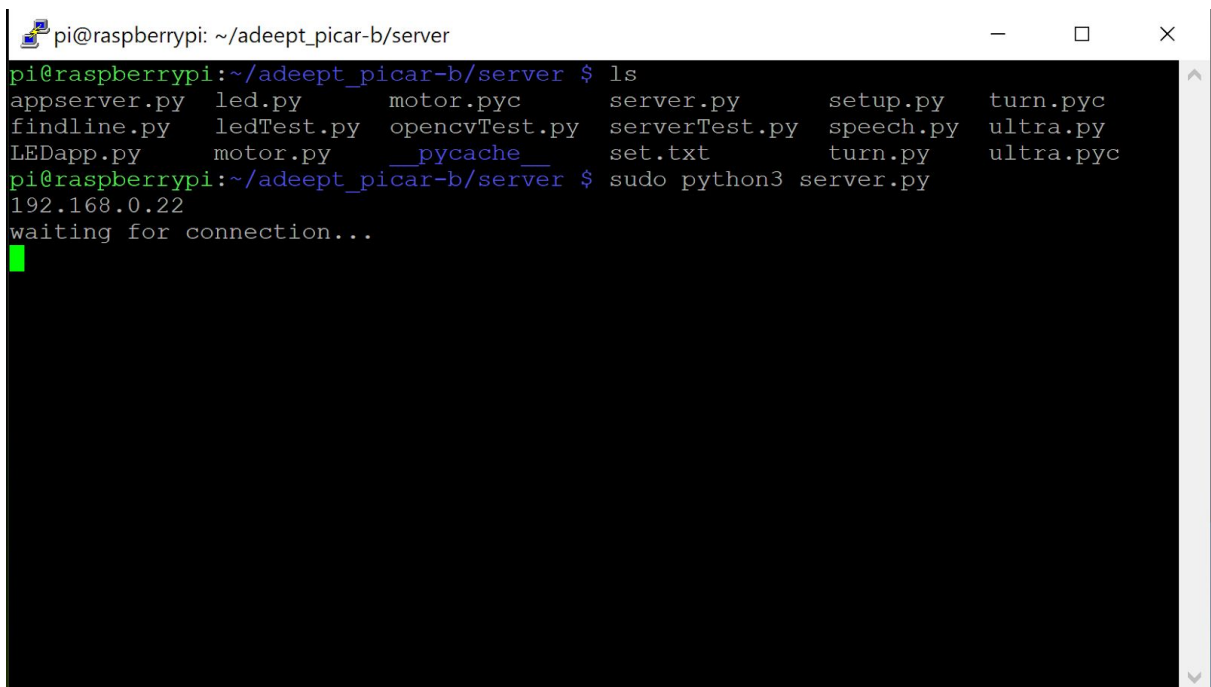


In my case, the IP address is 192.168.0.22. Once we connect to the Picar and we need to login to the system.



```
pi@raspberrypi: ~  
login as: pi  
pi@192.168.0.22's password:  
Access denied  
pi@192.168.0.22's password:  
Linux raspberrypi 4.19.75-v7+ #1270 SMP Tue Sep 24 18:45:11 BST 2019 armv7l  
  
The programs included with the Debian GNU/Linux system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent  
permitted by applicable law.  
Last login: Thu Nov 7 03:25:37 2019  
  
SSH is enabled and the default password for the 'pi' user has not been changed.  
This is a security risk - please login as the 'pi' user and type 'passwd' to set  
a new password.  
  
pi@raspberrypi:~ $
```

After that, we need to run the python server code in the Picar system.



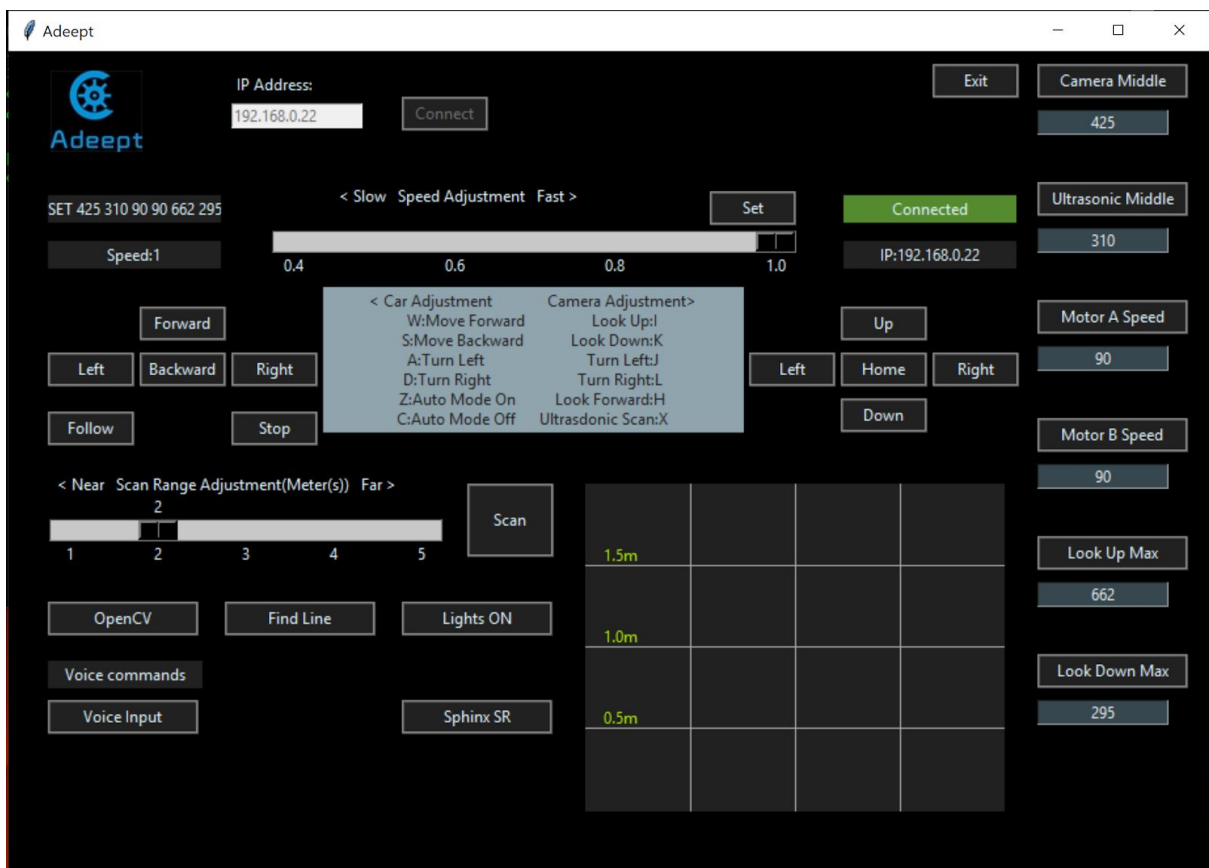
```
pi@raspberrypi: ~/adeept_picar-b/server  
pi@raspberrypi:~/adeept_picar-b/server $ ls  
appserver.py  led.py  motor.pyc  server.py  setup.py  turn.pyc  
findline.py  ledTest.py  opencvTest.py  serverTest.py  speech.py  ultra.py  
LEDapp.py  motor.py  __pycache__  set.txt  turn.py  ultra.pyc  
pi@raspberrypi:~/adeept_picar-b/server $ sudo python3 server.py  
192.168.0.22  
waiting for connection...  
█
```

Now the Picar is waiting for a connection, so we need to go back to our local desktop and run the client code.



```
Command Prompt - python client.py
IP.txt client.py logo.png
c:\PiCar-B\code\client>python client.py
```

Once, we run the client code and we are able to connect to the Picar and control py use python GUI.



Now we can place a box on the front of the Picar and run the auto mode and the Picar will start detecting the object by using the ultrasonic sensor. In the program, we set up a good distance as 0.4 meters, so the Picar will keep this distance between the object at the front of the Picar.

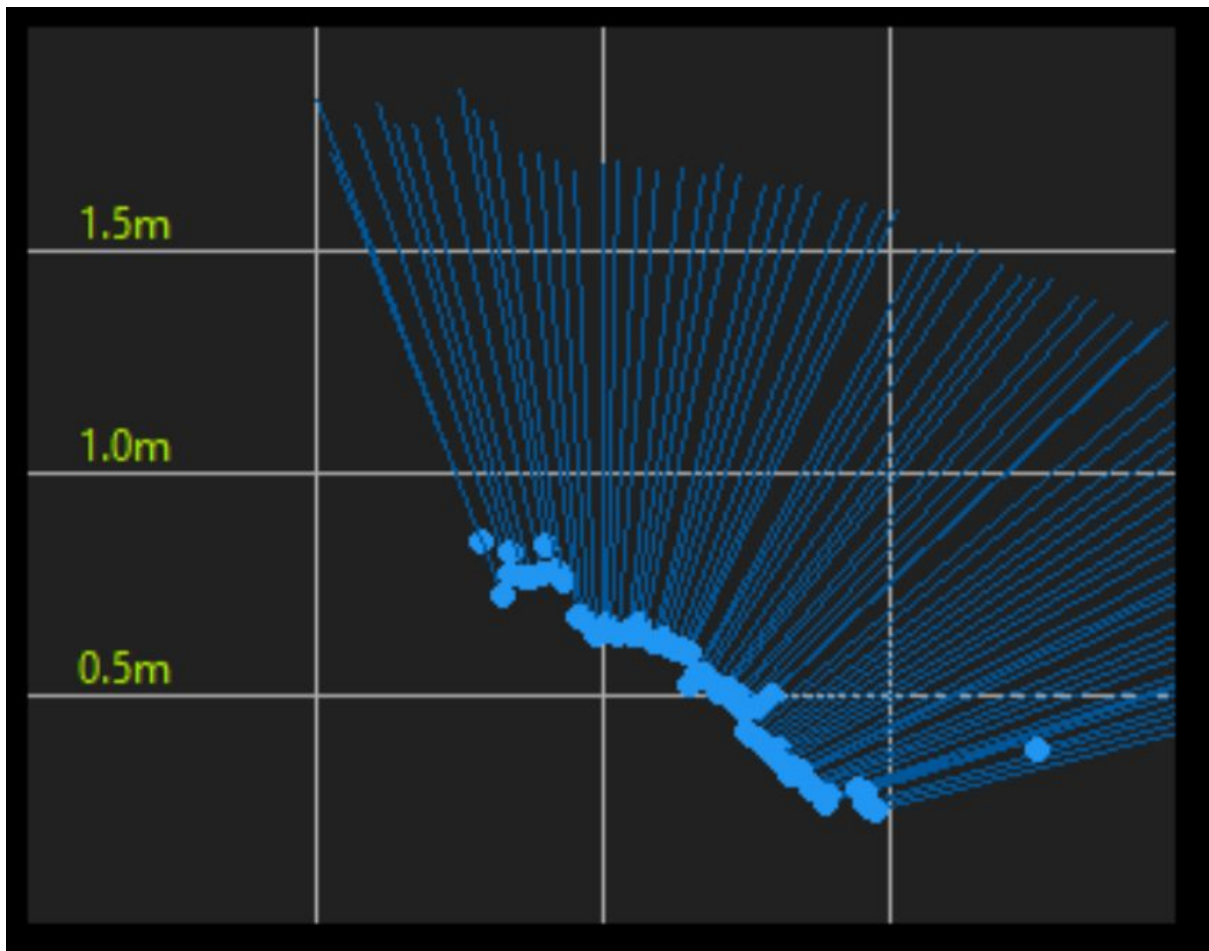


Now the Picar is at a good distance.

We try to move the box a little bit farther and the Picar will move forward until it is in a good distance range. In the picture maybe it is difficult to show the movement of Picar, but we will record it during our video presentation.



Finally, the ultrasonic sensor will be detecting the front of the Picar, and it will draw out a graph to show the distance between the object and itself.



Basic on this graph, as we can see we set up the farthest detecting distance is 2 meters, and we can see the left of the car is no object, and the middle to right is an object, so basic on this graph it enough for use to control the car and not to hit the object at the front of the car.

## Conclusion:

This week, we know how to use the ultrasonic sensor to control the car to keep a good distance between an object and itself. Also, we know how to use the graph to control and move the car with our visual image or camera to present the real environment.